NRL Report 7228

CROSEC, A Fortran IV–APT Program to Give Orthographic, Section, and Definable Perspective Views of a Planar-Curved Surface

KINGSLEY P. THOMPSON

Information Systems Group

April 28, 1971

NAVAL RESEARCH LABORATORY
Washington, D.C.

PREVIOUS REPORTS IN THIS SERIES

"Cross-Sectional Plots of Plane Intersections," K. P. Thompson, NRL Report 7025, Jan. 27, 1969

"CROSEC Technical Manual I–(NRL APT System): Computational Equations for Intersections Between Planes & Cylinders (Infinite & Bounded) Planes & Spheres," K. P. Thompson, NRL Report 7202, Jan. 26, 1970

UNCLASSIFIED

CONTENTS

ABSTRACT

A Fortran IV program, CROSEC (MOD 2.0) has been written to operate in the environment of NRL's APT system on the CDC 3800 Computer. This program enables the part programmer for numerically controlled tooling systems to obtain plots of the intersections between a specially designated plane (called the HOPE plane) and certain defined forms in his APT part program. The forms with which intersections can be obtained are lines, planes, circles, cylinders, and spheres. When the HOPE plane corresponds to a coordinate plane (XY, XZ, and YZ), an orthographic projection pattern is obtained corresponding to engineering drawing's top, front, and end views, respectively. When the HOPE plane cuts through the volume of the part surface and the plotting limits are the part's minimum and maximum dimensions, a true view of the cross section is obtained. When the HOPE plane lies outside the volume of the part and is skewed with respect to the coordinate planes, an abstract pattern is obtained because the planes and cylinders are not bounded. Such an abstract pattern, when composed only of plane intersections, through an unprogrammed process, can be interpreted as a perspective view of the part. When the plotting limits extend beyond the dimensions of the part and the HOPE plane cuts its volume, a combination of true view and interpretable perspective view is obtained.

Preliminary considerations suggest that a hidden-line-detection addition to the program would not be difficult because the abstract pattern indicates depth in an easily identifiable manner, and that the abstract intersection pattern forms a suitable framework for describing tool motions in an interactive environment.

Up to ten viewing (i.e., HOPE) planes can be designated in one Part Program. The report includes programmer instructions, a complete description of CROSEC (MOD 2.0) with flowcharts and listings and, as an example, the computer output resulting from a run on a production piece. CROSEC (MOD 2.0) is the implementation of the computational forms developed in NRL Reports 7025 and 7202.

PROBLEM STATUS

This is an interim report on a continuing problem.

AUTHORIZATION

NRL Problem Z00.01

Manuscript submitted December 21, 1970.

CROSEC, A FORTRAN IV — APT PROGRAM TO GIVE ORTHOGRAPHIC, SECTION, AND DEFINABLE PERSPECTIVE VIEWS OF A PLANAR-CURVED SURFACE

INTRODUCTION

This report is both a program and a programming manual for CROSEC (MOD 2.0). CROSEC (MOD 2.0) is a series of Fortran IV programs that were written into the framework of NRL's APT system to provide a capability for plotting intersections between a specially designated plane and other defined planes, cylinders, and spheres.* (In APT, a line is a special case of a plane—an intersection of the XY plane with a vertical plane—and a circle is a special case of a cylinder parallel to the Z axis; therefore, intersections with these special cases, lines and circles, are also obtained in CROSEC (MOD 2.0).) The output can be obtained on either a CALCOMP plotter or on a flatbed Gerber plotter.

This is the fourth of a series of CROSEC-related publications. The first, NRL Report 7025 (Ref. 1), described an earlier version of the program, CROSEC (MOD 1.0), that obtained plane intersections only. The second, NRL Report 7202 (Ref. 2), presented the computational forms for the intersections with cylinders and spheres. Finally, a paper on CROSEC (MOD 2.0) was delivered at the 1970 Fall APT Technical Conference in Chicago, Illinois, and was published in the proceedings of that meeting (3).

CROSEC (MOD 2.0) is the computer implementation of the computational forms of Ref. 2, in addition, of course, to the original capability of CROSEC (MOD 1.0). In this version up to ten intersection plots can be obtained in one Part Program. The programmer can inhibit a particular plane, line, or circle from producing an intersection by using a prefix X in its name. Minimum and maximum XYZ limits can also be established to define the bounds within which intersections are desired, and to a limited extent plotting scale can be controlled.

CROSEC gives the part programmer the ability to view the structural framework within which he plans to move his tool. Generally speaking, intersection patterns are identical with line endpoint projection patterns in the three principal planes (TOP-XY, SIDE-YZ, and FRONT-XZ). An intersection pattern becomes a cross-sectional pattern when the cutting plane on which the pattern is plotted is actually intersecting through the volume of the part defined by the part program. (Examples of CROSEC intersection patterns are presented in Refs. 2 and 3 and in Appendix B of this report.)

ORGANIZATION

This report consists of some programmer instructions and the computer program descriptions. The programmer instructions discuss the Part Program additions that are needed to obtain CROSEC patterns as well as interpretation principles, and the descriptions are of the two forms of programming elements constituting CROSEC (MOD 2.0). The first of these are the minor modifications to three standard APT subroutines, FINI, PRCNTL, and LBSRCH, and the second, the 27 especially written programs and subprograms organized into three segments, that perform the CROSEC actions. The appendixes contain the program listings and a sample run.

---

*APT references are available through the Engineering Services Division, Naval Research Laboratory, Washington, D.C. 20390, and from the IIT Research Institute, 10 West 35th Street, Chicago, Ill., 60616.

PROGRAMMER INSTRUCTIONS

Earlier instructions given on p. 60 of NRL Report 7025 still apply in general (including the use of HOXYMIN and HOXYMAX cards); however, some improvements have been made in this version and they are described below.

1. More than one HOPE plane (up to a maximum of ten) can be specified in the Part Program by using suffix numbers 1 through 9, and terminating the series with a HOPE or HOPEN name for the last plane; viz.,

HOPE1 = PLANE/ . . .
HOPE2 = PLANE/ . . .
.
.
.
HOPE9 = PLANE/ . . .
HOPE = PLANE/ . . .

For efficiency it is recommended that the HOPE card series be in order, but the only programming restriction is that the HOPE or HOPEN card be *below* the HOPE cards with suffix numbers.

2. Often a line or plane is used for a purpose other than part description, such as positioning a tool, and its appearance in an intersection pattern can cause confusion. Such lines or planes can be voided from CROSEC patterns by preceding their names with an X (e.g., XPL1 = PLANE/ . . .). This X prefix can also be used to suppress circle, cylinder, and sphere intersections.

3. The overlay form of operation is less expensive and preferred over the use of the Load and Go Edit procedure described in Ref. 1. The order of cards for an overlay run* is

a. Job card
b. Demand card (use 53730B)
c. Equip card† for
   CALCOMP plotter, 10=PL
   Gerber tape output, 17=TW
d. Equip card for overlay tape
   49 = . . .
e. LOAD MAIN card
f. Part Program cards
g. End of File card

Appendix B contains a reproduction of a Part Program and the output related to it. This includes the Gerber plots and the line printer messages.

The scale of the CROSEC plot (CALCOMP) can be controlled by the proper choice of variables on the HOXYMIN and HOXYMAX cards. Generally speaking, these cards define the minimum and maximum points on the part as shown in Fig. 1. This does not provide

*NRL CDC 3800 Drum Scope Environment.
†Two separate overlay tapes are needed, only one of which is used at any one time. One of these is used when running CALCOMP plots, and the other when running to obtain Gerber tape output.

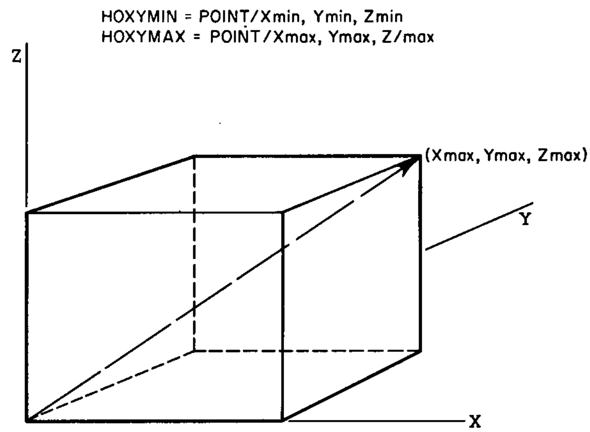


Fig. 1 - Minimum and maximum points on part can be used to define HOXYMIN and HOXYMAX cards

an easily definable scale, however, such as 1:1, and so if circumstances permit, a certain compromise can be obtained. In order to work this compromise, a brief description of what CROSEC does with the HOXYMIN and HOXYMAX values is needed.

CROSEC makes two vectors from the HOXYMIN and HOXYMAX points and uses their magnitudes in a comparison to get the largest. The magnitudes are defined as

$$\left|V_{min}\right| = \sqrt{X_{min}^2 + Y_{min}^2 + Z_{min}^2} . \tag{1}$$

$$\left|V_{max}\right| = \sqrt{X_{max}^2 + Y_{max}^2 + Z_{max}^2} . \tag{2}$$

The *largest* of the two magnitudes, called V, is used in the computation of SCALE;

$$\text{SCALE} = \frac{V}{5} . \tag{3}$$

where the figure 5 has the plot significance shown in Fig. 2. The programmer can obtain a 1:1 plot scale by forcing V to be 5 if he uses

HOXYMIN = POINT/0, 0, 0

HOXYMAX = POINT/2.88, 2.88, 2.88,

since

$$5 = \sqrt{25} \approx \sqrt{2.88^2 + 2.88^2 + 2.88^2} .$$

This principle can be extended to other scale ratios by using different multiples of 2.88 as shown in Table 1.

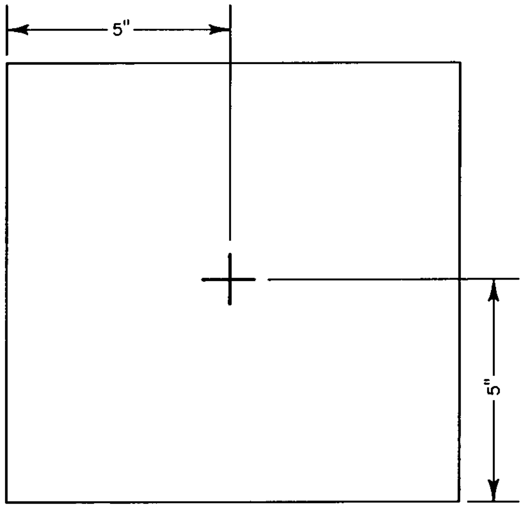


Fig. 2 - The figure 5 used in the computation of SCALE corresponds to the translation in X and Y from the lower left-hand corner of the 10-in.-square plot to the center of the square for representation of HOPE origin

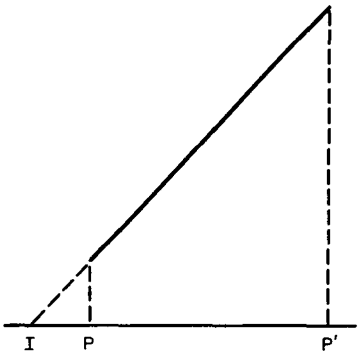
Table 1
Scale Table

Scale Ratio Desired	V for Eq. (3)	Common Linear Limit for X,Y,Z on HOXYMAX Card
⋮	⋮	⋮
$\frac{1}{2}:1$	10	$5.77 \approx \frac{10}{\sqrt{3}}$
1 : 1	5	$2.88 \approx \frac{5}{\sqrt{3}}$
2 : 1	2.5	$1.44 \approx \frac{2.5}{\sqrt{3}}$
⋮	⋮	⋮

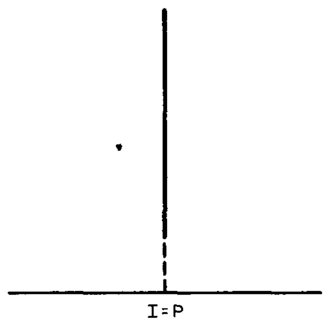
NOTE — If this method of selecting HOXYMIN (all zeros here) and HOXYMAX limits violates the bounds shown in Fig. 1, then some portions of the CROSEC intersections might not be obtained, depending on the HOPE plane involved.

CROSEC INTERPRETATION

CROSEC produces *intersection* patterns. A pattern can be interpreted as a projection pattern, a cross-section pattern, or as a framework for defining a perspective view, depending on the location of the HOPE plane and the HOXYMIN and HOXYMAX bounds. An individual pattern can be totally one type or a combination of types.



(a) Separate I and P



(b) Identical I and P

Fig. 3 - Distinguishing between intersection points and projection points

CROSEC produces intersection patterns because the planes and cylinders* are not bounded by the part and can intersect the HOPE plane in three-dimensional (3-D) space beyond the piece being conceived by the programmer. It is important to understand the difference between an intersection point and a projection point. Figure 3 shows a two-dimensional, two-line sketch to illustrate this difference. Figure 3a shows two line segments inclined to each other at an angle less than 90°. If the lower segment is considered as a base line, the upper line when extended intersects the base line at I, but has projections of its endpoints at P and P'. When the segments are mutually perpendicular, however, as shown in Fig. 3b, the intersection and projection points are identical. A part programmer generally defines his planes as perpendicular or parallel to the XY plane. Consequently, CROSEC intersections in the principal planes (when HOPE = XY or XZ or YZ) are interpretable for parallel and perpendicular planes as projections and compare favorably with engineering drawings for ease of understanding (see Appendix B).

To illustrate the principles of CROSEC interpretation, let us consider some patterns resulting from CROSEC and a cube whose edges are one unit in length and is defined by the six planes $x = 0$, $y = 0$, $z = 0$, $x = 1$, $y = 1$, and $z = 1$.

Initially let the HOPE plane be defined by $x + y + z = 0$. It will pass through the origin and be equally inclined at 54.8° to each principal axis. It will also be tangent to a vertex of the cube. Figure 4a shows a perspective view of the cube and the HOPE plane with the pattern resulting from the intersections of the six planes when allowed to extend beyond the cube. Figure 4b is a true view of the pattern. Point a is projected from the cube vertex (1, 0, 0) and is shown in both views 4a and 4b. Figure 5a identifies the line of intersection with six Roman numerals, viz., I for $x = 0$, II for $y = 0$, III for $z = 0$, IV for $x = 1$, V for $y = 1$, and VI for $z = 1$. The 0 value planes form the three inner intersecting lines, and the 1 value planes form the outside triangle. Such a pattern has been dubbed "the triangular world." There are seven equilateral triangles in this pattern, the outer one and two different size sets of three. These are enumerated in Table 2. The center point can be considered as a degenerate triangle of the three inner lines.

*CROSEC (MOD 2.0) does produce truncated ellipses when the HOPE plane passes through the ends of a cylinder TL units long, but is allowed to produce full ellipses when the intersection is fully within or fully without the TL bounds. HOXYMIN and HOXYMAX limits apply only to plane intersections in CROSEC (MOD 2.0).

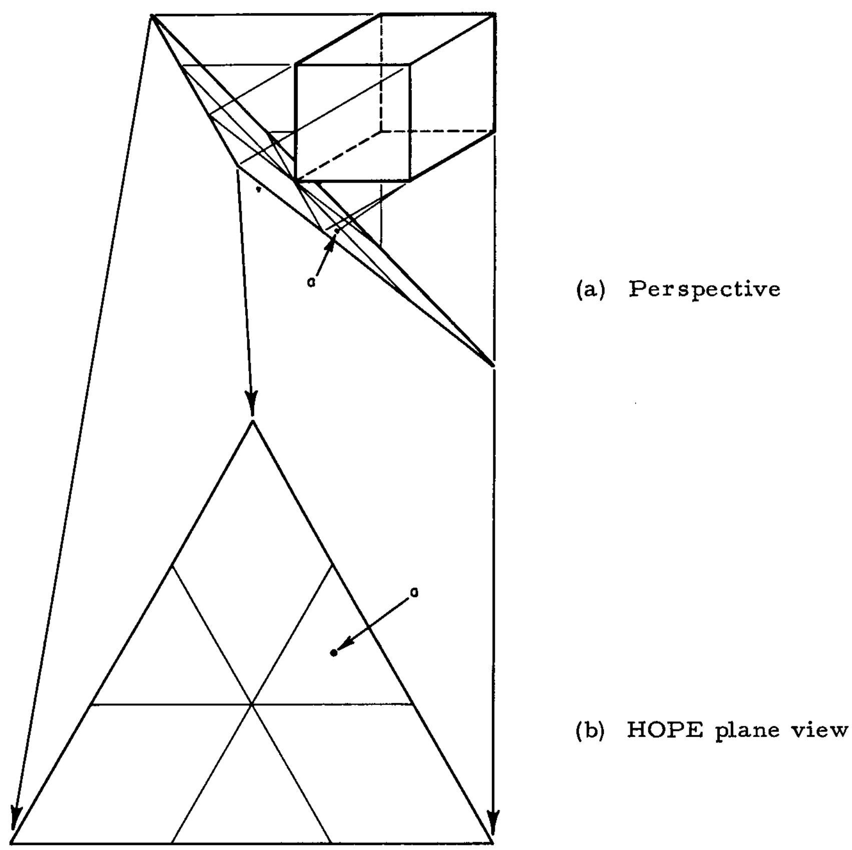


Fig. 4 - CROSEC intersection pattern for a cube on the equiangular plane, X + Y + Z = 0

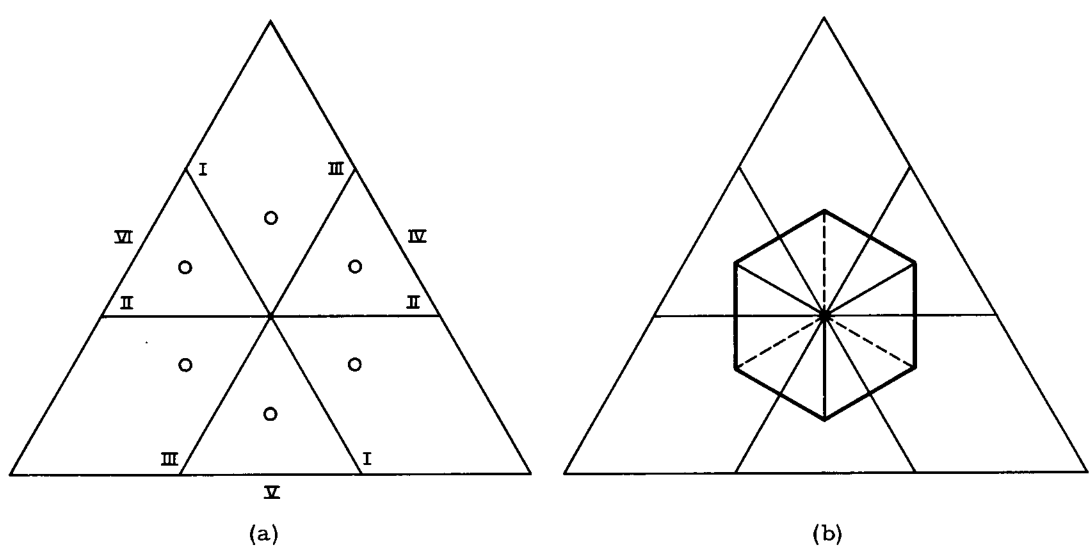


Fig. 5 - The centers of seven equilateral triangles (shown in (a)) are joined to form the projected image of the cube (b)

Table 2
Triangles in the Pattern

Triangle	Component Sides
Outer, largest	{VI, IV, V}
Intermediate	{VI, IV, II}, {III, IV, V}, {VI, I, V}
Smallest	{VI, I, II}, {III, IV, II}, {III, I, V}
Reduced to point	{I, II, III}

The three sides of an equilateral triangle in the pattern correspond to three planes intersecting to form a vertex in the cube. For example, point a is the centroid* of the triangle formed from the intersection lines of planes $x = 1$, $y = 0$, and $z = 0$. Figure 5a highlights the centroid of each of the seven equilateral triangles with a small circle, each a projection of a vertex of the cube. In Fig. 5b these projection points have been joined to outline the latent, perspective image.

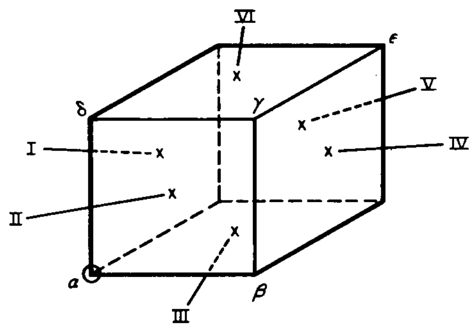
The creation of the latent, perspective image has emphasized the transformation or mapping undergone by the vertexes of the cube into equilateral triangles. Another transformation, perhaps not as immediately obvious, is that an edge of the three-dimensional cube becomes, in the triangular world, the common vertex of two unequal, equilateral triangles that share two common sides in part. Thus, for example, in Fig. 6b the triangle ⓪ (at center) - ④ - ⑥ and the triangle ③ - ④ - ⑦ have a portion of sides II and IV in common. It could be said then that the edge of intersection between planes II and IV has been mapped into vertex ④. It is this very feature of having two planes (lines of intersection) in common that justifies drawing a line of the latent, perspective image between two "center" points, from β to γ in this case in Fig. 6b.

A face of a cube becomes transformed in the triangular world into four such vertexes. This statement, however, must be qualified by an understanding of the special circumstances taking place at the center of the pattern. The HOPE plane is tangent to the cube at a vertex (see Fig. 4a). The three lines of intersection for planes I, II, and III all pass through the center of Fig. 6b. The triangle for this vertex transformation has been reduced to a point.

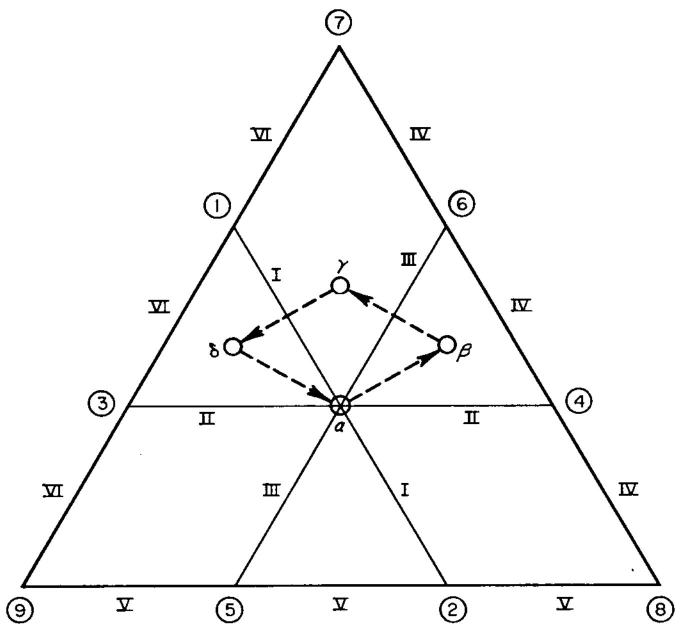
The procedure or algorithm for tracing out the face $\alpha\beta\gamma\delta$ in the perspective view is depicted in the sequence below Fig. 6a. The first line contains four groups of three planes each, one for each of the cube's vertexes α, β, γ, and δ. The first and second groups have planes II and III in common, the second and third groups have planes II and IV in common, and so forth, until the last group has planes I and II in common with the first group, completing the cycle.

Finally the last of the transformations from the 3-D world into the triangular world is that undergone by the depth dimension. (In the triangular world the detection of triangles is basic since the center of the triangle establishes the location of the projected point for the perspective image and the size (area) of the triangle is the measure of the distance (depth) of the source point from the HOPE plane.) The extreme variation in depth corresponds to the tangent point (planes I, II, and III intersecting) to the largest triangle (planes IV, V, and VI) or points ⑦, ⑧, and ⑨, which correspond to the cube vertex point diagonally opposite point α. There are two intermediate depths represented

*The terms *centroid* and *center* are used synonymously.



(a) Perspective



(b) HOPE plane view

I, II, III ⟶ II, III, IV ⟶ II, IV, VI ⟶ I, II, VI
II, III ⟶ II, III
II, IV ⟶ II, IV
II, VI ⟶ II, VI
I, II   I, II

(c) Face-tracing algorithm

Fig. 6 - An edge of the 3-D cube in the triangular world becomes a vertex of two unequal, equilateral triangles that share portions of two lines of intersection in common. A face of the perspective view of the cube can be logically traced out utilizing this edge-transformation rule.

by triangles ④-⓪-⑥ and ③-④-⑦. There are three of each of these two sizes. The projection of point ϵ onto the HOPE plane is *masked* by the tangent point α. Any perspective edge leading into this masked point can be wholly or partially hidden. For example, triangles ③-⑦-④ and ⑦-⑧-⑨ have planes IV and VI in common and the perspective edge γ - α (Fig. 5b) is a hidden line. These transformations from the 3-D world into the triangular world are summarized below:

1. A 3-D point $\Rightarrow$ center of equilateral triangle (with variations)
2. An edge $\Rightarrow$ two common sides (common vertex) of two unequal equilateral triangles
3. A face $\Rightarrow$ closed path of points connecting centers with two common sides
4. Depth $\Rightarrow$ different size triangles (with variations).

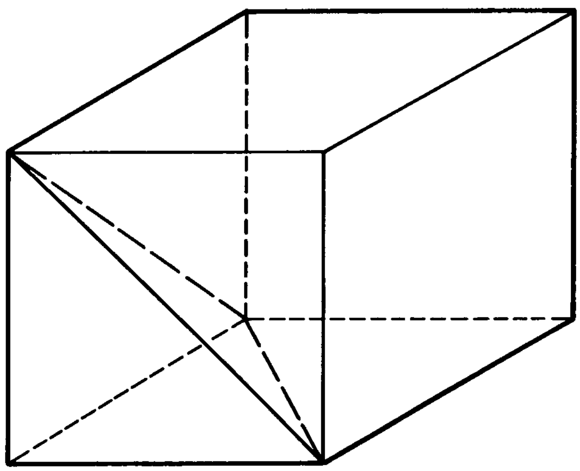
Now for some cross-section examples. If the HOPE plane passes through the vertexes (1, 0, 0), (0, 1, 0), and (0, 0, 1) as in Fig. 7a, a combination true view* of the cross-section and intersection pattern is formed (Fig. 7b). If a portion of the cube is considered discarded (Fig. 7c), the cross section is seen to fit into the projected image which has been drawn in Fig. 7b with wavy and dashed lines. The vertexes of the cross section should be considered as zero-area triangles of the form , , and . Each is composed of three intersecting lines, corresponding to three planes of the cube, forming a part of the latent image. In Fig. 8a the HOPE plane again cuts through the unit cube. The normal distance from the origin to the HOPE plane is one unit. The intersection is five-sided, and is shown in true view in the CROSEC pattern in Fig. 8b. If part of the cube is assumed to be cut away (Fig. 8c) to make the cross section visible, then part of the latent image of the cube is not drawn. The remaining portion of the latent image is shown in wavy and dashed lines in Fig. 8c. This is an instance when the vertexes of the latent image and the vertexes of the cross section are distinct and separate.

In Fig. 9 the HOPE plane is tangent to that edge of the cube that lies along the Z axis. A perspective sketch is shown in Fig. 9a, and a true view of the CROSEC pattern in Fig. 9b. The lines of intersection are identified by their plane number in the table and on the lines of intersection in Fig. 9c. Planes I and II have coincident lines of intersection which is the cube edge tangent to the HOPE plane. In Fig. 9d the latent image points are shown by circles and these are joined by wavy lines and a dashed line to form the latent image. The vertexes (a, b, c, d, e, f, g, h) of Fig. 9a have corresponding image points (a′, b′, c′, d′, e′, f′, g′, h′) in Fig. 9d. There are no triangles. A U degenerate case is present in this situation. The three planes corresponding to an image point are of the shape └───┘ or ┌───┐. Half the length of the base of the U is the distance measure. The image point is located at this halfway position. For example, the image point e′ for planes V, VI, and IV is halfway along intersection line VI, and distance $\overline{ie'}$ is its depth measure. Similarly image point g′ has depth measure $\overline{ig'}$ and image point b′, has depth measure $\overline{a'b'}$. Since the intersection lines for planes I and II overlap their image point with III, c′ is the cross-section image point where I, II, and III intersect. Image line $\overline{e'f'}$ is dashed because it is hidden by face a′b′d′c′. The depth measures for face a′b′d′c′ are shown below.

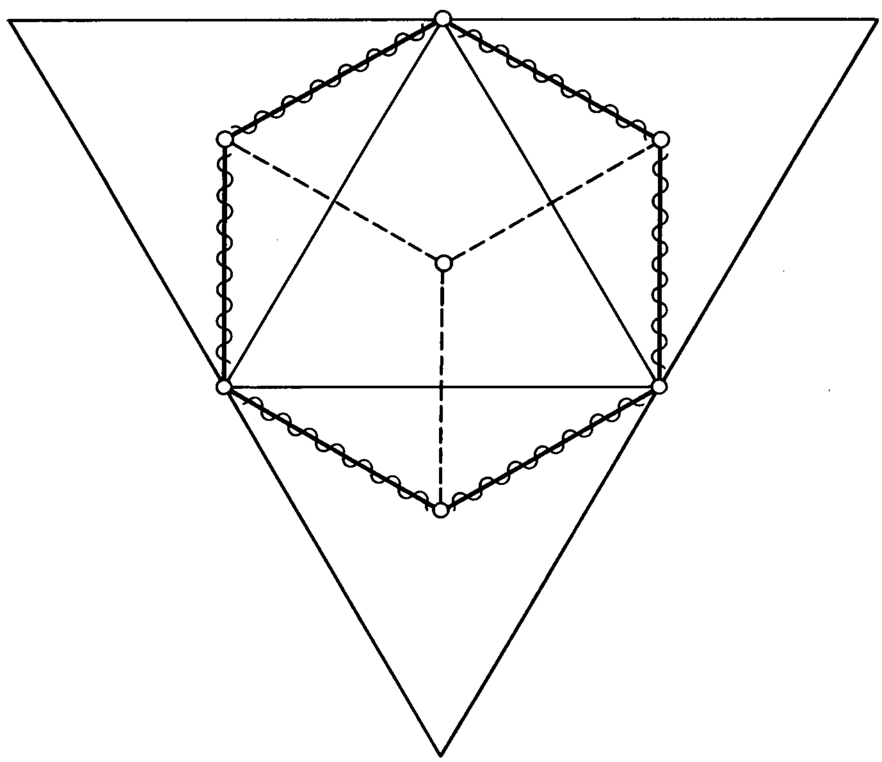
Point #	Depth Measure
a′	0, tangent
c′	0, tangent
b′	line a′b′
d′	line c′d′

All of these are less than $\overline{ie'}$, the depth measure for line e′f′, and this fact forms the justification for concluding that e′f′ is hidden.

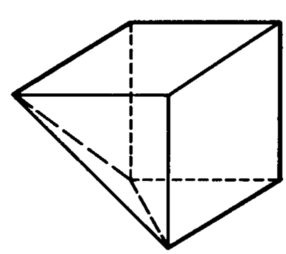
*A true view is a perpendicular projection that permits scaling of exact lengths, a correct, undistorted picture.



(a) Cube with cutting HOPE plane

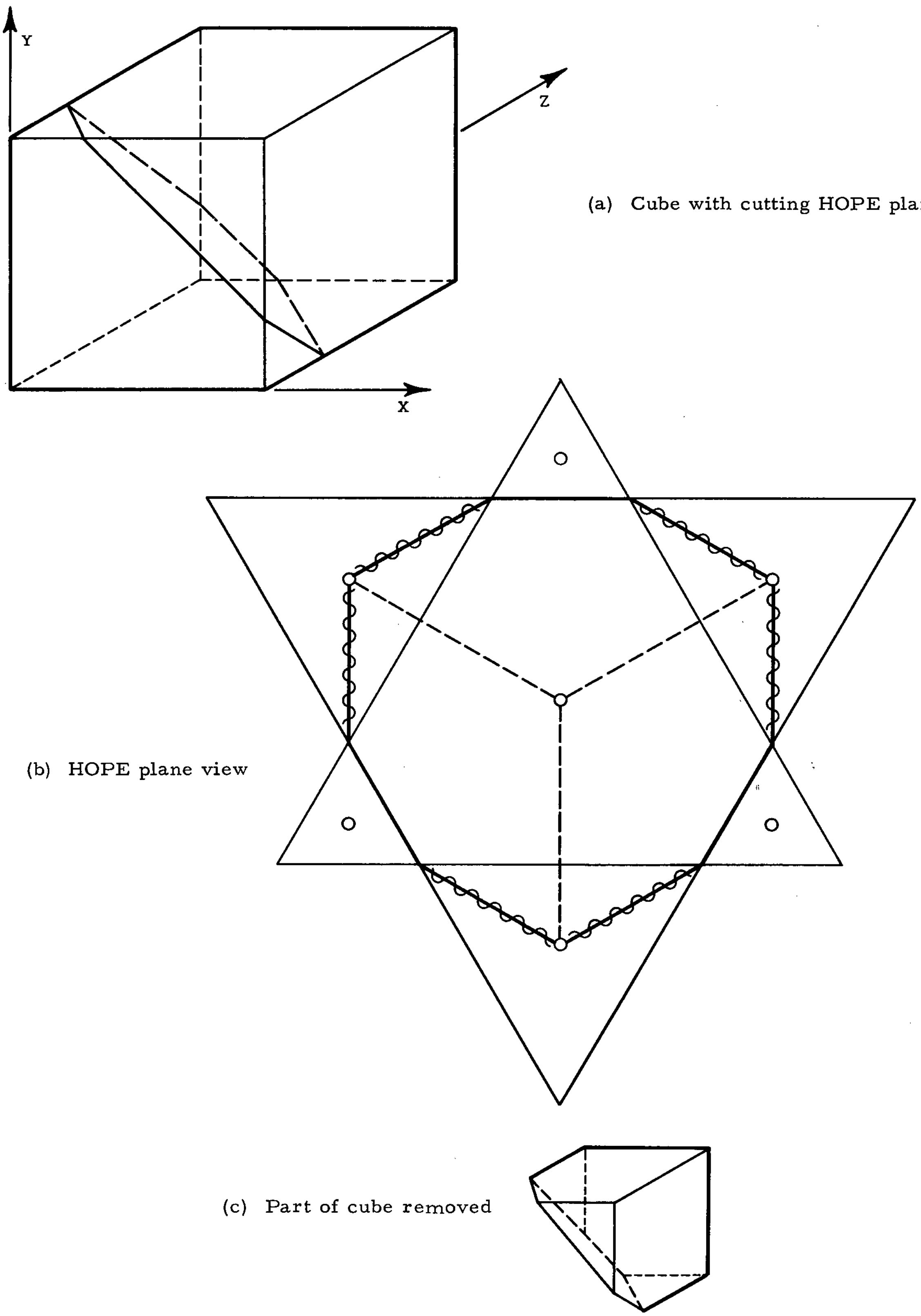


(b) HOPE plane view



(c) Part of cube removed

Fig. 7 - The HOPE plane intersects three vertexes of the cube

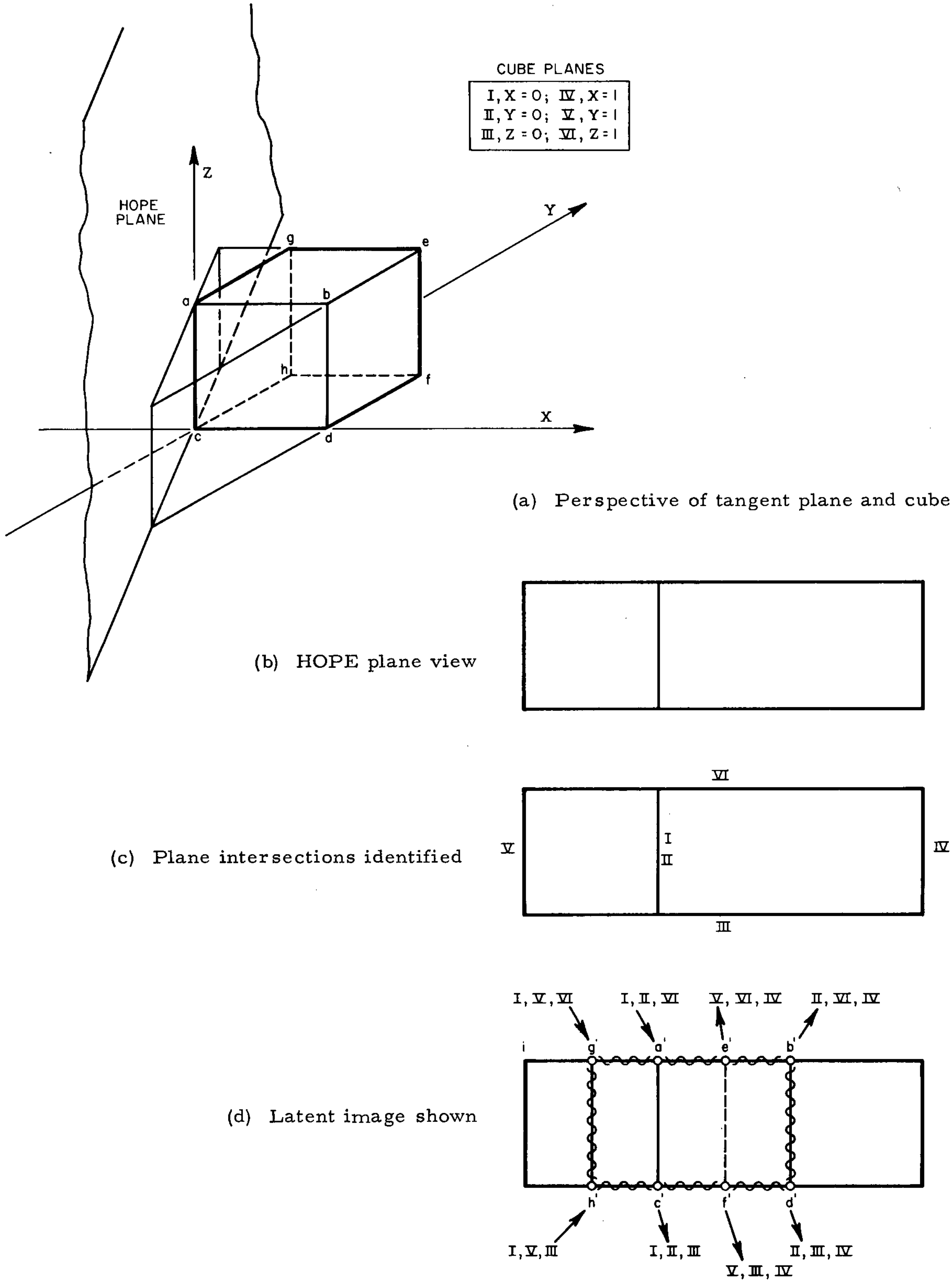


(a) Cube with cutting HOPE plane

(b) HOPE plane view

(c) Part of cube removed

Fig. 8 - The HOPE plane is equiangular with a unit normal distance. The cross-sectional pattern is five sided.



(a) Perspective of tangent plane and cube

(b) HOPE plane view

(c) Plane intersections identified

(d) Latent image shown

Fig. 9 - HOPE plane tangent to an edge of the cube

In Fig. 10 the origin is centered in the z = 0 face, the cube itself is rotated 30° about the Z axis, and the equiangular plane passes through the origin. A perspective sketch of this organization is shown in Fig. 10a. The cube faces are identified with Roman numerals I through VI; the HOPE plane is depicted in an arbitrary manner outside the cube, except that its cross section with the cube attempts to be accurate. The corresponding CROSEC pattern is in Fig. 10b, with the intersection lines identified, and the latent image with the cross section is shown in Fig. 10c. The triangles are not equilateral, but they each have centers, and therefore definable image points. A′ is the image of A, and similarly B′, C′, and D′ are images of B, C, and D.

What about curved surfaces in CROSEC interpretation? Spheres and cylinders are the only nonplanar forms presently handled by CROSEC. The sphere is well behaved, it closes on itself in all three dimensions, and only gives circle intersections—for this discussion, forget tangent conditions—which can probably be interpreted as part of the cross section, since the sphere is usually considered as making a contribution to the volume of a part. If the HOPE plane is outside the part and the sphere is inside the part, unlike part planes the sphere will not have an intersection.

Cylinder intersections inside the part are part of the cross section and present no problems in interpretation. The cylinder, however, exudes in one dimension and can produce circles or ellipses of intersection outside the part dimensions (TL length) in CROSEC (MOD 2.0). Such a situation occurs in Fig. 11a, the extended PARTNO TESTING example of earlier report work (see pp. 7, 8, 10 of NRL Report 7025 and p. 37 of NRL Report 7202). The equiangular plane x+y+z=1 in this situation produces a small triangle of intersection and the latent image shown in Fig. 11b. To be a part of the perspective view, the cylinder should have an "image ellipse" that lies in face abcd. The CROSEC ellipse is a true view of what the cross section would be if the HOPE plane were parallel to the present HOPE plane and intersected the cylinder within the part. Its presence helps identify the tangent lines of intersection but does not directly contribute to the image. The CROSEC ellipse would have to be rotated, translated, and reduced to be a part of the latent image. The magnitudes of the rotation, translation, and reduction can be readily defined from Fig. 11b and are delineated in Fig. 11c. They are

Rotation, θ (60°),

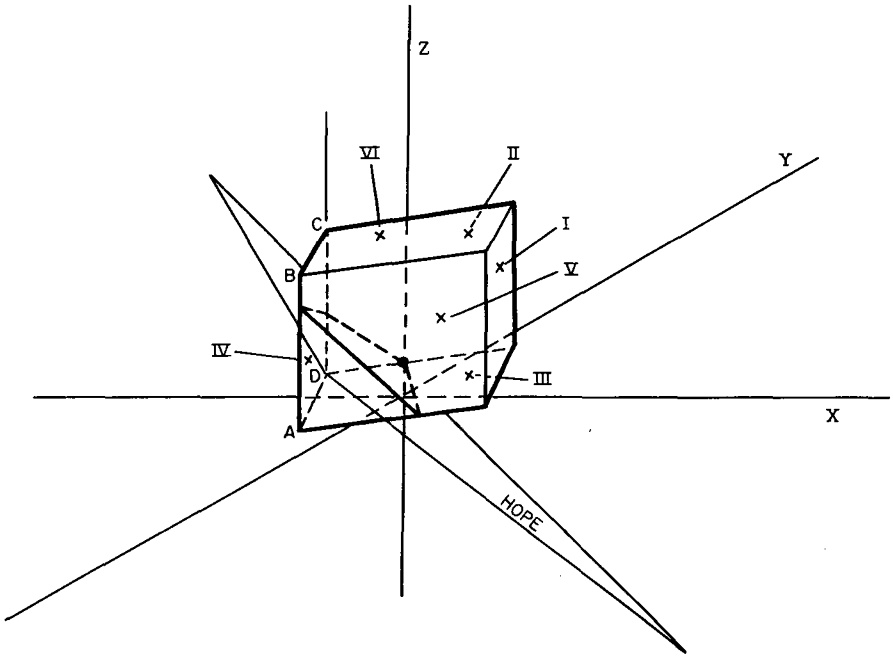
Translation, from h to i, and

Reduction, ef/bg.

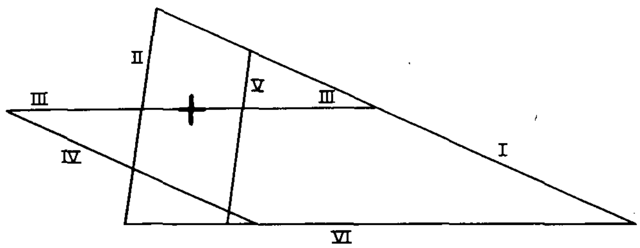
The CROSEC ellipse is tangent to midpoints of the rhombus wxyz formed by *sides* of triangles. The image ellipse is tangent to midpoints of the rhombus abcd formed from parts of *altitudes* of the triangles which correspond to X- and Y-axis directions. The translation is *along* a triangle altitude corresponding to the Z-axis direction. The transformation is not done by CROSEC (MOD 2.0) but could be done in one of two ways; (a) given the proper constants generate a new ellipse, or (b) transform the set of points that define the CROSEC ellipse.

CROSEC pattern interpretation is interesting (for those who like such things), but even for a simple cube the pattern can become complicated, and for a large part can become close to impossible.* The average user of CROSEC (MOD 2.0) will probably stick to (a) the principal coordinate planes for PLAN, SIDE, and END views or HOPE planes parallel to them, or (b) tightly limited cross-sectional views. However, should

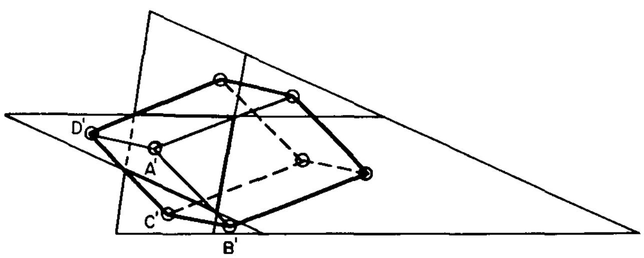
*However, even though a triangular world pattern can be complex, some trends of what the image would be like can be readily surmised; e.g., centers of a group of triangles, with two sides in common, would form an edge. See Appendix B, Fig. B5.



(a) Cube rotated



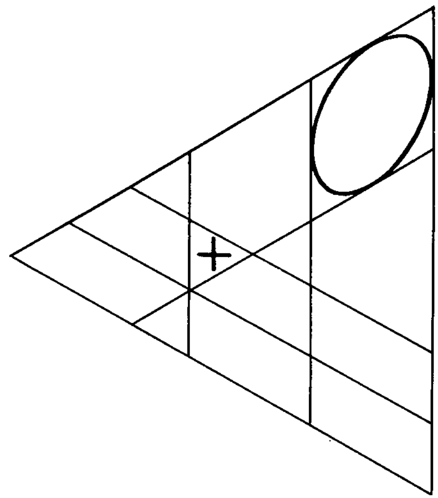
(b) HOPE plane view

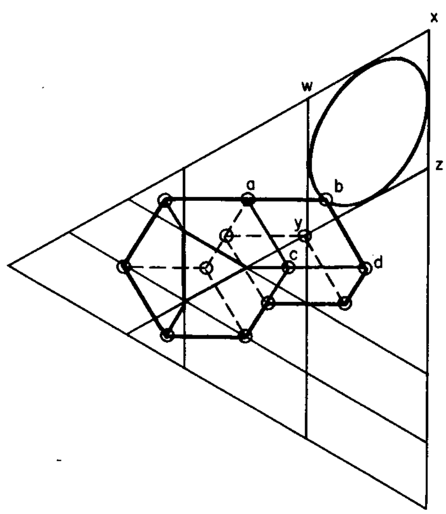


(c) Latent image shown

Fig. 10 - Example of CROSEC triangles that are not equilateral

(a) HOPE plane view of extended PARTNO TEST





(b) Latent image

(c) Elliptical transformation

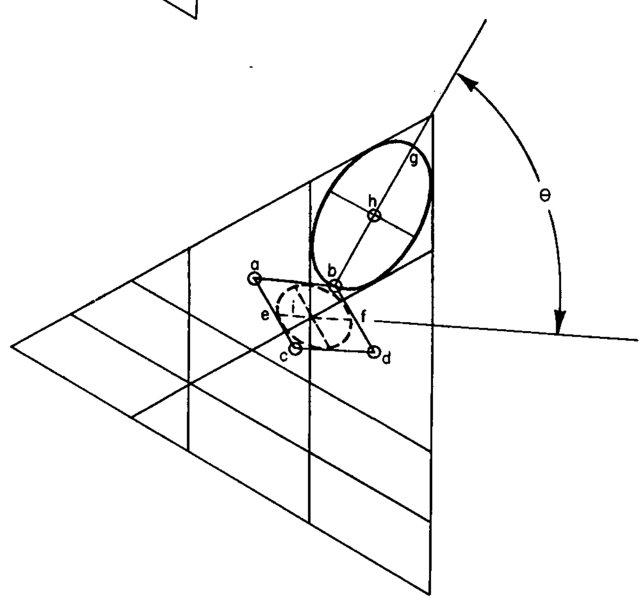


Fig. 11 - Triangular world CROSEC pattern for augmented PARTNO TESTING, the latent image and the transformed ellipse

the user care to visit the triangular world, the computer printout gives sufficient information on angles between planes and the HOPE plane and identification of intersection lines to make CROSEC interpretation possible. The equilateral triangular world for a principle plane-oriented part is entered by use of one of eight possible HOPE planes, viz.,

$$x+y+z=C$$
$$-x+y+z=C$$
$$x-y+z=C$$
$$x+y-z=C$$
$$-x-y+z=C$$
$$-x+y-z=C$$
$$x-y-z=C$$
$$-x-y-z=C,$$

where C is a constant, zero, negative or positive, and represents the normal distance from the origin. Degenerate forms of non-equilateral triangles and differing combinations of three intersections that define an image point can occur unpredictably if the HOPE plane and part planes are randomly oriented. Any time a plane is at an angle other than 90° to the HOPE plane its intersection is no longer a projection and needs interpretation; so if a pattern looks peculiar, the printout should be checked for angle information.

The nine principles of CROSEC interpretation are summarized below.

CROSEC INTERPRETATION PRINCIPLES

1. CROSEC produces intersections.

2. A plane's intersection is a projection when the plane is perpendicular to the HOPE plane.

3. Principle plane views will be projections if the defined planes are parallel to XY, YZ, or XZ (called well defined).

4. CROSEC cuts through the volume of a part, producing true-view cross sectional lines which can be bounded by HOXYMIN and HOXYMAX.

5. CROSEC patterns in equiangular plane views for well-defined planes provide equilateral triangles from which a latent perspective image can be constructed from parts of altitude (median) lines. These patterns provide a convenient framework for outlining tool motions.

6. CROSEC patterns in non-equiangular plane views that are not projections can also provide latent perspective images.

7. Cross sections and latent images can coexist in the same pattern.

8. Cross sections can include intersections with spheres and cylinders inside the volume of a part.

9. CROSEC patterns with cylinders outside a part must be transformed for inclusion in images.

The question might be asked, "Why not compute images directly?" The answer is that CROSEC provides an excellent beginning for doing exactly that. A preliminary consideration of some references on the hidden-line problem solutions (4-10) shows a transition from elaborate volume considerations by Roberts (4) to mostly planar considerations by Lerman (10) (his exception is some normal computations for faces). A triangular world solution does not appear too awkward, and perhaps, just perhaps, might have some advantages with its considerations entirely in the HOPE plane, since it has a depth measure in the area of the triangles and/or degenerate solution line length.

CROSEC provides *not only* a base for image definition with hidden-line detection, but, in addition, principal plane views and true view cross sections. It is our strong contention that CROSEC has been and is the right way to go as a graphical aid to APT. Beyond this is the possible application to defining tool motion.

TOOL MOTION

In the APT language a tool motion statement is defined in terms of three surfaces. (a) The part surface PS which is perpendicular to the end of the tool, (b) the drive surface DS which is parallel to the axis of the tool, and (c) the check surface CS which ends the motion. A CROSEC pattern in the triangular world could be used in an interactive fashion to define tool motion because of the convenient display of all relevant surfaces. Figure 12 depicts the passage of a tool around the cube in four steps, with z=0 as the part surface (see table in middle of figure). Tool rotation about a major axis could be related to rotation of the CROSEC figure about a vertex bisector as shown in Fig. 12e. The axes in Fig. 12e correspond to the latent-image orientation of Fig. 5b.

Criticism of such an approach to tool motion definitions centers around its abstractness or unreality. The alternative is the display of a two-dimensional plan view (XY, XZ, or YZ) with depth keyed in, or via a perspective view which always lacks equal measurement of the three dimensions. The triangular world, as has already been stressed, treats each of the three dimensions in an unbiased and equal fashion. The transformations that relate the triangular world to the real world for the purposes of tool motion definitions are illustrated by the CROSEC pattern for the cube. These are

1. Parallel lines define parallel surfaces.

2. The shortest of such a series of parallel lines would be the front surface in that direction, and the longest line the back surface.

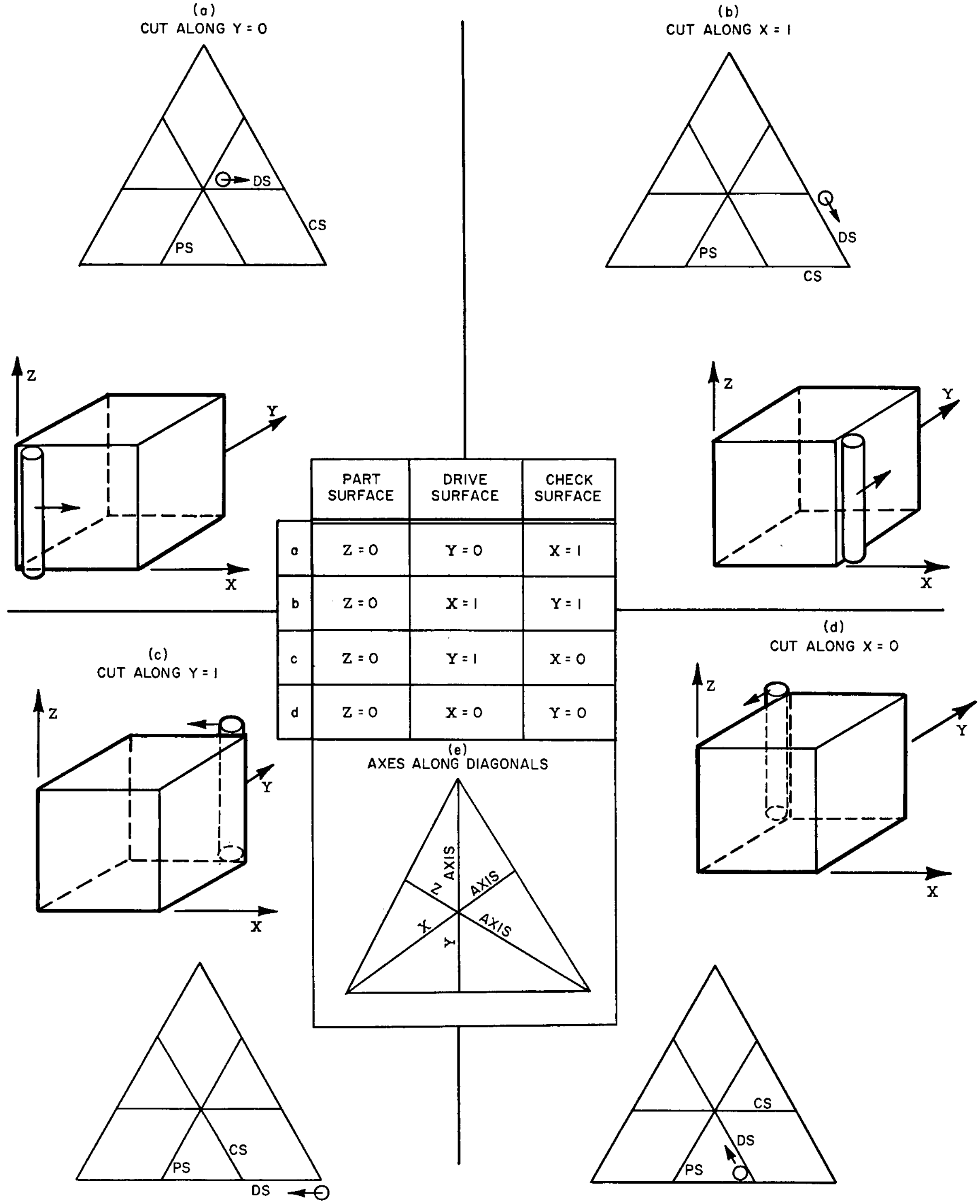
3. A 90° 3-D world angle can be represented by either 60° or 120° in the triangular world, depending on the orientation of front and back surfaces.

4. The traversal of a rhombus does not require a change of position of the tool axis.

5. The traversal of a triangle requires one 90° change in the tool axis (e.g., to successively pass over the x=1, y=1, and z=1 surfaces by passing around the outside triangle).

Assume for the moment that an interactive version of CROSEC existed with a background data structure. Then orientation and identification of surfaces in the triangular world could be accomplished by quizzing the data structure via light pen or cursor hits on the CROSEC pattern or by calling up the latent image simultaneously with the CROSEC pattern or by a combination of these methods. If the pattern were too complex then windowing could be done to simplify the appearance in any given sector.

Such is the potential of the CROSEC pattern in the triangular world created by an equiangular HOPE plane. Next is a detailed description of the elements of CROSEC (MOD 2.0).



	PART SURFACE	DRIVE SURFACE	CHECK SURFACE
a	Z = 0	Y = 0	X = I
b	Z = 0	X = I	Y = I
c	Z = 0	Y = I	X = 0
d	Z = 0	X = 0	Y = 0

Fig. 12 - Tool motion considerations

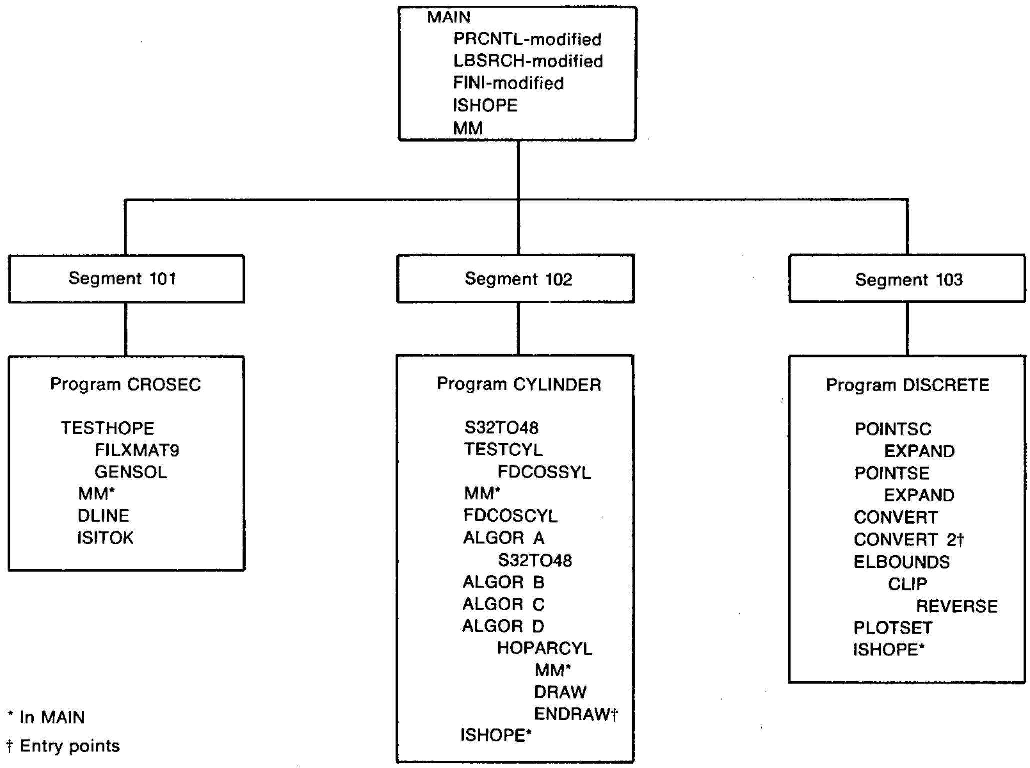


Fig. 13 - Program and subprogram dependence for those elements specially written for CROSEC (MOD 2.0) and described in this report. See listings for additional information on APT and Library subprogram linkages.

CROSEC (MOD 2.0)

Overall Considerations

The detailed description of the programming elements making up CROSEC (MOD 2.0) now commences. It is helpful to get an overall look at the entire CROSEC family by considering Fig. 13, which shows the organization. "MAIN" constitutes that group of program units that stay in core at all times during APT processing. However, only those elements affecting CROSEC are listed in the MAIN box. The first three (PRCNTL, LBSRCH, FINI) are APT subroutines that have been slightly modified to allow the CROSEC tie-in and to control the segment calls. ISHOPE and MM belong to CROSEC and reside in MAIN because they are referred to by more than one segment. CROSEC is subdivided into three segments.* The first of these, 101, entered at Program CROSEC, processes plane/line intersections. The second, 102, entered at Program CYLINDER, obtains cylinder/circle and sphere intersections in coefficient form, and the third, 103, which is entered at Program DISCRETE, produces and plots discrete sets of x′, y′ points for the ellipses and circles of intersection obtained in Segment 102.

Figure 14 is a flowchart of the control for the CROSEC system that allows for the transfer between segments. Conventional APT processing proceeds until the end of

*101 means Segment 1 of Overlay 1, 102 means Segment 2 of Overlay 1, etc. Use of "Segment 1" and "101" interchangeably should not be confusing in this context. Same comment applies for Segments 2 and 3.

Subroutine FINI, where a call is made on ISHOPE. If a HOPE plane is found IFILL7 is coded with "101," otherwise IFILL7 gets a "0." In any event, PRCNTL is called which looks at the contents of IFILL7. If the "101" is present Segment 1 is called, but if "0," control passes to APT Section II and CROSEC is bypassed.

At the conclusion of Segment 1 an unconditional jump is made to Segment 102 by filling IFILL7 with 102 and calling PRCNTL. At the conclusion of Segment 2 if the intersection is not valid, Segment 2 is recycled to look for more cylinders, circles, or spheres. If, however, the intersection is valid, control (via IFILL7 and PRCNTL) goes to Segment 3 for discrete point processing.

When the defined symbol table has been fully searched for cylinders, circles, or spheres, ISHOPE will be reentered if there are more HOPE planes to be processed; otherwise, IFILL7 is filled with 9999 as a key to PRCNTL to continue the APT processing.

The listing of the program elements described below is reproduced in Appendix B in the order in which they are described.

MAIN Program Units

Changes to Subroutine PRCNTL — The modification to PRCNTL that controls the segment calls is shown in Fig. 15a. The CROSEC insertion begins after statement 9300. KFLAG1 is temporarily set to 0 and IFILL7 examined for its five possible values, 0, 101, 102, 103, or 9999. These correspond, respectively, to

No CROSEC processing,

Ready for Segment 1,

Ready for Segment 2,

Ready for Segment 3, and

Done with CROSEC processing.

If there is no CROSEC processing to be done, transfer is to statement 9031. Segments are called via a call to LBSRCH with the parameters 101, 102, and 103. If CROSEC processing is completed the normal PRCNTL flow is entered by transfer to statement 9031. If IFILL7 does not contain any of its allowed values, a trouble message is printed before reentering the normal procedure.

At statement 9031, just prior to returning to normal PRCNTL operation, the following flags are set:

IFILL7 is set to zero. This controls decisions in PRCNTL.

IFILL8 is set to zero. This is equivalent to INCHOPE which is used by ISHOPE to keep track of the current HOPE plane.

IFILL9 is set to zero. This is equivalent to PLOTPLNO and is used by both ISHOPE and Program CROSEC to identify the defined symbol table's location of the current HOPE plane.

KFLAG1 is set to 1 to signify the completion of Section I processing.

After statement 9049, just prior to the call to EXIT, a series of final plot termination calls are given. These are peculiar to NRL's CDC 3800 plot package. First the paper is

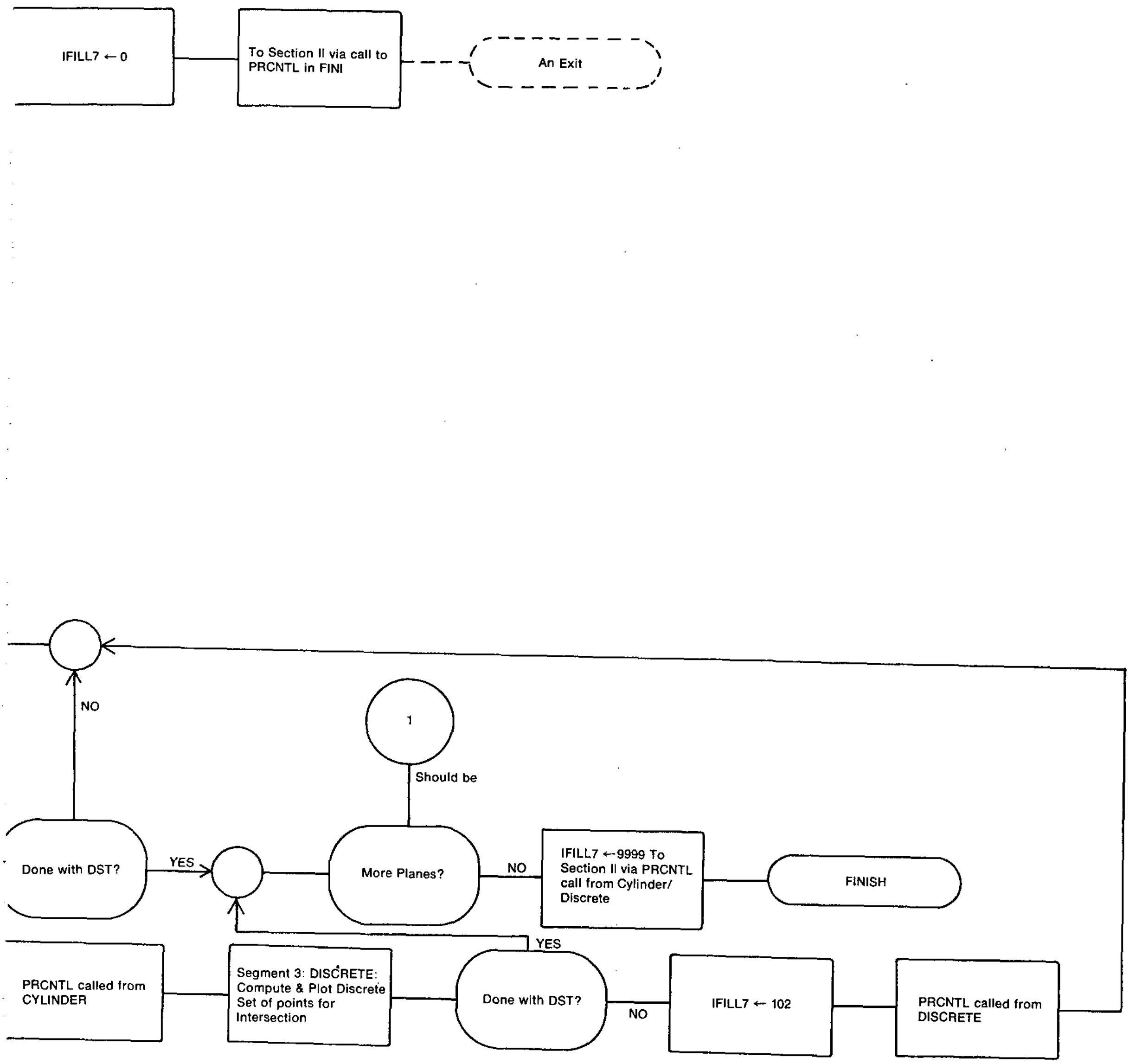
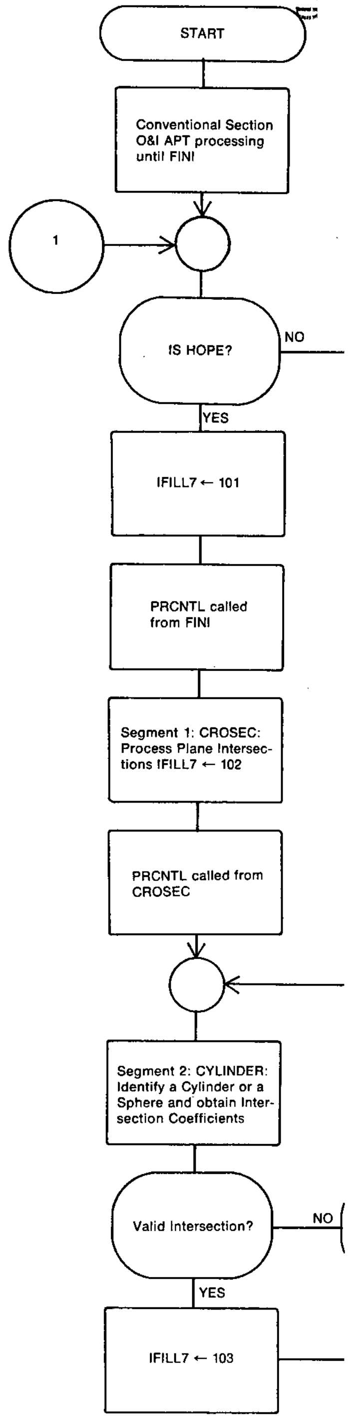
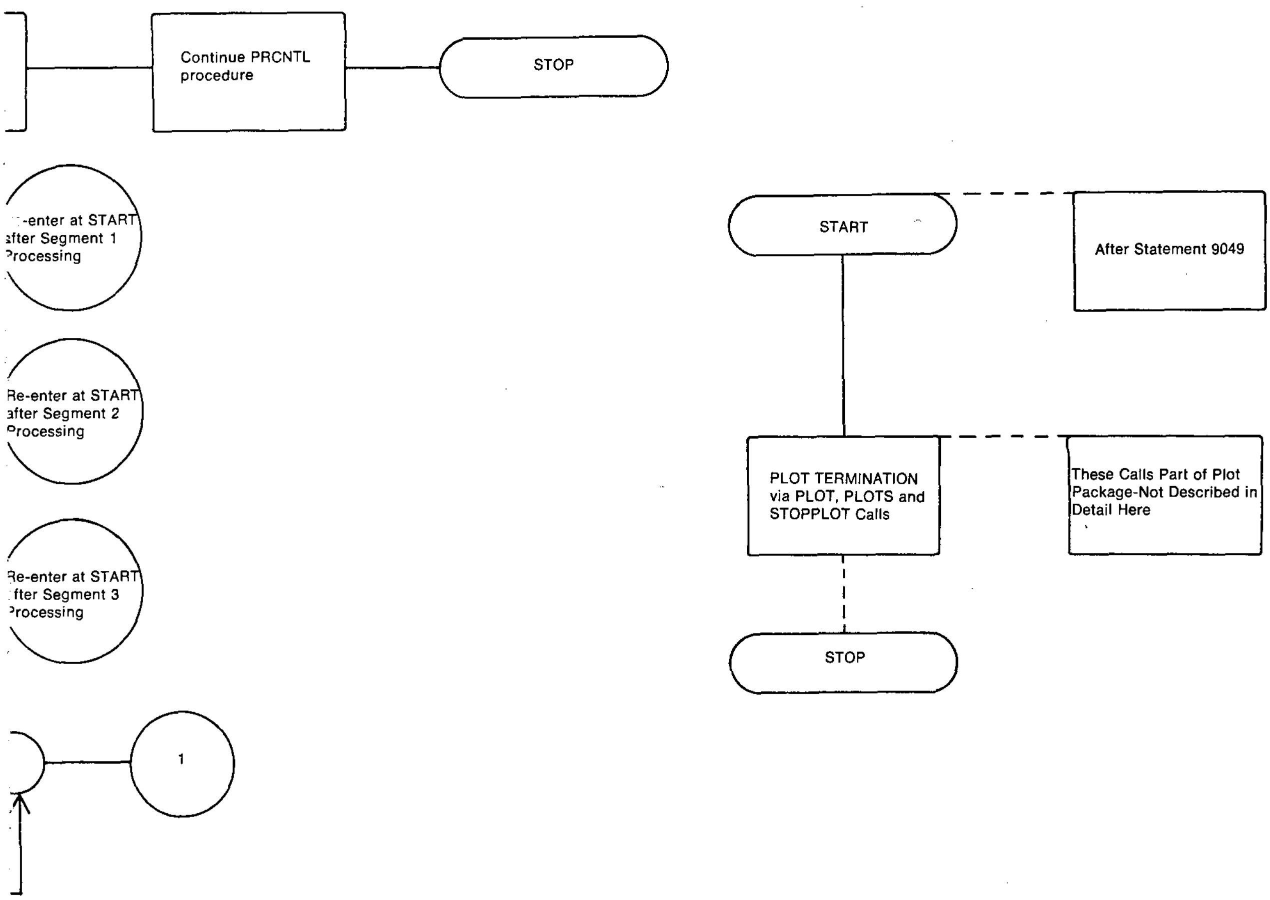


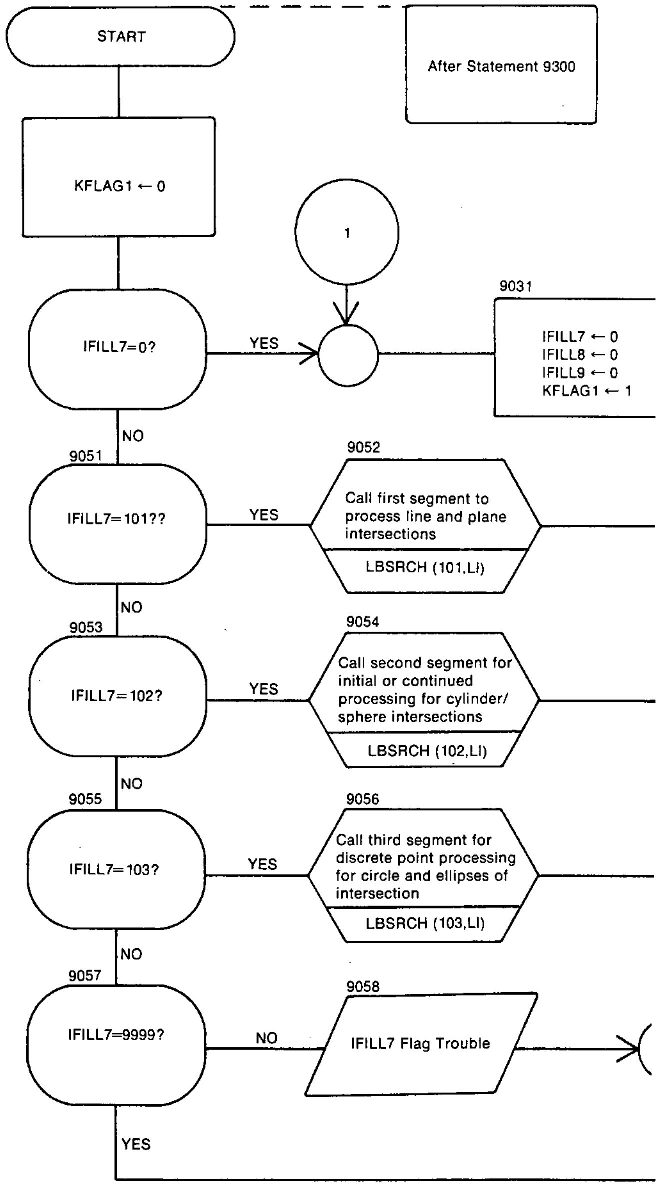
Fig. 14 - CROSEC system control


START
Conventional Section O&I APT processing until FINI
1
IS HOPE?
NO
YES
IFILL7 ← 101
PRCNTL called from FINI
Segment 1: CROSEC: Process Plane Intersections IFILL7 ← 102
PRCNTL called from CROSEC
Segment 2: CYLINDER: Identify a Cylinder or a Sphere and obtain Intersection Coefficients
Valid Intersection?
NO
YES
IFILL7 ← 103



ig. 15 - CROSEC segment control in PRCNTL

UNCLASSIFIED



F

advanced to a new origin, then the pen is raised and moved to this origin, and the final printout call is given in the call to STOPPLOTS which is given only once and results in a termination message to the operator and on the plot itself. Placing these calls in this location allows for a common termination for Section I and Section III plotting and minimizes special paper tape feed and symbol punching peculiar to the Gerber plot package. This plot termination is diagrammed in Fig. 15b.

This completes the description of the PRCNTL modifications.

Changes to Subroutine LBSRCH — The three segments of Overlay 1 are listed in the DATA statement for the NON TP array,

DATA (NON TP = 100, 101, 102, 103, 200, 300, 400, 18 (7777)).

The 101, 102, 103 sequence is inserted between 100 and 200 and the last term changed from 21 (7777) to 18 (7777). The original form of this DATA statement is on cards 00240270 and 00240276. These two cards have been removed and replaced by the two cards numbered 00240271 and 00240272, which contain the revision. This is the only CROSEC change in LBSRCH.

Changes to Subroutine FINI — The set of variables in COMMON CROSSEC are inserted into FINI so that they will become a part of Section I common. This insertion takes place after card #15450680.

Entrance to CROSEC processing is accomplished by removing card #15450970, which is a statement 30 call to PRCNTL, and inserting two cards containing

30 CALL ISHOPE

and

CALL PRCNTL

on cards numbered 15450971 and 15450972, respectively.

The call to ISHOPE will establish a flag in IFILL7 if a HOPE plane is present. This flag is tested in PRCNTL and directs the segment calls.

These are all the FINI changes necessitated by CROSEC.

Subroutine ISHOPE — The purpose of this subroutine is to locate a HOPE plane in the Defined Symbol Table (D.S.T.). Initially it is called from Subroutine FINI, and succeeding calls are from Program CYLINDER or Program DISCRETE.

ISHOPE incorporates the common from Section 0, Section I, and CROSSEC. The following equivalences are established:

INCHOPE = IFILL8

PLOTPLNO = IFILL9.

PLOTPLNO is declared an integer, and the INCAR array is filled with the nine numbered names of possible HOPE planes, HOPE1 through HOPE9.

The action begins with the execution of a top-of-form to separate the CROSEC output to follow from earlier APT output.

When ISHOPE is entered for the first time INCHOPE will be equal to 0, and transfer is made to statement 110; otherwise, the attention shifts to statement 170.

At statement 110 PLOTS is initialized, the length of the Defined Symbol Table (D.S.T.) is placed in IDSEND and the I loop is entered.

The first word of a two-word pair is placed into ITRY, and the D.S.T. positional information is stored in PLOTPLNO.

ITRY is tested. If it contains HOPEN or HOPE, only one HOPE plane is present; otherwise, HOPE1 is tested for at statement 140. If none of these three possibilities is present the loop is recycled.

If only one HOPE plane is present INCHOPE is set equal to "ONCE"; but if *more* than one HOPE plane is present, as indicated by the presence of HOPE1, then INCHOPE is set equal to 1. Following the setting of INCHOPE, IFILL7 is filled with 101 (to be acted upon in PRCNTL) and the return is made.

At statement 170 if ISHOPE has previously been entered, then INCHOPE is incremented and the J loop entered. JTRY picks up the first word of a D.S.T. pair and saves the positional information in PLOTPLNO.

If JTRY is equal to HOPE or HOPEN, then the *last* HOPE plane has been located and transfer is made to statement 220; otherwise, a test is made to see if JTRY is equal to the INCHOPE position of the INCAR array, which, if successful, generates a jump to statement 130. If all three of these tests fail, the loop is recycled.

If the tests in the I loop and the J loop prove unsuccessful, appropriate trouble messages are printed out, followed by a return. (Since IFILL7 will be empty, PRCNTL will discontinue CROSEC processing.)

The ISHOPE flowchart is shown in Fig. 16.

Subroutine MM (XP, YP, ZP, XR, YR, ZR, XMAT9) — This subroutine was described on p. 42 of the first CROSEC report, NRL Report 7025. In CROSEC (MOD 2.0), because of the segmenting involved, it has been placed in MAIN along with ISHOPE. MM is called by programs or subprograms in more than one segment. It is called by Program CROSEC and Subroutine DLINE in Segment 101, as well as by Program CYLINDER and Subroutine HOPARCYL in Segment 102. It performs an important duty in converting a point from the XYZ system to the HOPE plane system.

This subroutine has been passed through TIDY. See Fig. A7 in Ref. 1 for the flowchart.

Segment 101

Changes in Program CROSEC — Program CROSEC was originally described in Ref. 1 (NRL Report 7025) for CROSEC (MOD 1.0). What follows is a description of the changes that have been made in the program to bring it up to its present form in CROSEC (MOD 2.0).

1. Program CROSEC in CROSEC (MOD 2.0) has been resequenced in its card numbering, the statements have been renumbered to be more orderly, and all format statements appear at the end of the listing. (This is done by a utility program called TIDY.)

2. The original listing numbers CRS 2150 through CRS 2500 performed the task of locating the HOPE plane in the defined symbol table. This job is now performed by

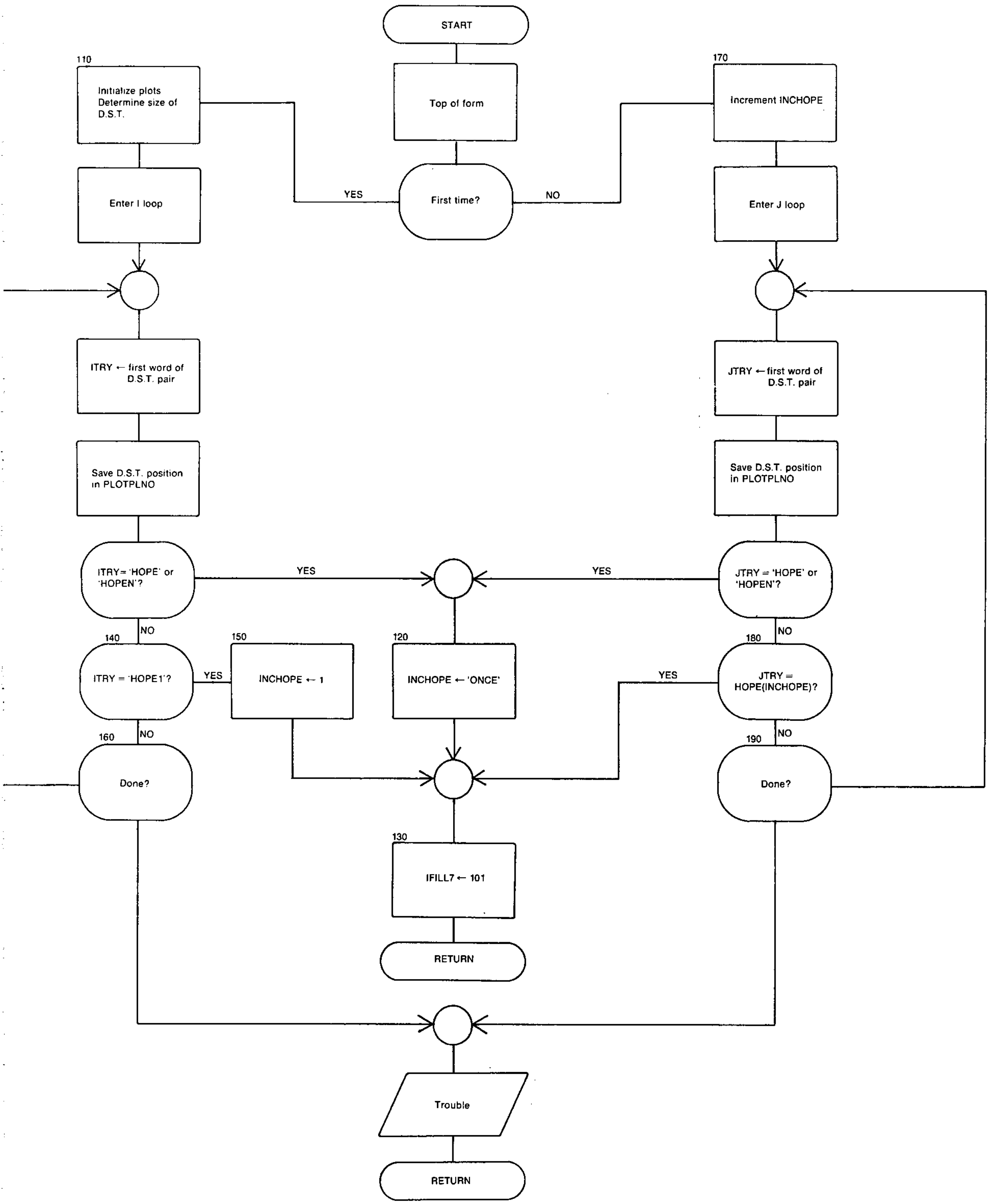


Fig. 16 - Subroutine ISHOPE

Subroutine ISHOPE for the multiple HOPE plane capability. When Program CROSEC is now entered as segment 101, a HOPE plane has already been located. Its constants have been stored in A, B, C, and D of CROSSEC common. Its table location is in PLOTPLNO, and the length of the table is in IDSEND.

3. There are three places in this program where the defined symbol table is searched to identify planes. In MOD 1.0 these spots occur in the vicinity of the original listing numbers CRS 3190, CRS 6240, and CRS 6570. In MOD 2.0 the operation is expanded to exclude picking up *any* of the possible 10 HOPE planes and any plane whose name begins with an X. These changes are found in the vicinity of the corresponding new sequence numbers CRS 2820, CRS 5850, and CRS 6220.

4. Four parameters have been added to the call to Subroutine ISITOK. They are IONCE, KR, IP, and SCALE. In MOD 1.0 this call is at CRS 7170. In MOD 2.0 this call is at CRS 6910.

5. Three parameters have also been added to the call to Subroutine DLINE. They are IP, KR, and SCALE. In MOD 1.0 this call occurs twice, at CRS 6140 and CRS 7470. In MOD 2.0 the calls are at CRS 5750 and CRS 7090.

6. In MOD 2.0, Program CROSEC ends by incrementing IDSEND, filling IFILL7 with 102 and calling PRCNTL. This accomplishes the fetching of Segment 102. CROSEC MOD 1.0 was not segmented and ended by terminating the plot and returning to FINI from whence it was called.

7. A few of the comments have been changed to describe MOD 2.0 features. They are self-identifying.

The purpose of Program CROSEC in its present form is to provide control in Segment 101 while the intersections between the current HOPE plane and all other allowed planes are processed. The flowchart in Fig. A2 of Ref. 1 is still applicable except for the first decision block, which should be deleted.

Subroutine TESTHOPE – This subroutine, which creates the HOPE plane coordinate system, was originally described within the framework of CROSEC (MOD 1.0) on pp. 31 through 36 in NRL Report 7025. For CROSEC (MOD 2.0) the following changes have been made:

1. The parameters have been eliminated by the addition of CROSSEC common.

2. For the case when the HOPE plane is parallel to the YZ plane the X′ axis is now chosen to be the Y axis and the Y′ axis to be the Z axis because of its more natural orientation. The parameters for the FILXMAT9 call are

0,1.0,0,0, 0,0,1.0,0, 1.0,0,0, D*A, XMAT9.

3. Three new cases have been added in case D is zero, meaning the HOPE plane passes through the origin. The solution approach is essentially the "intersection" set of solutions described in NRL Report 7202, in the discussion related to the establishment of the cylinder's system. After selecting the X′-axis direction cosines, in one of three possible ways, and picking up A, B, C for the Z′ axis, the Y′ axis is computed as the vector cross product of Z′ and X′.

4. The error message now begins "FAILED 9 TESTS..." instead of the previous "6."

A corrected flowchart is given in Fig. 17.

Subroutine FILXMAT9 (U1,..., U12 XMAT9)– There are no changes to Subroutine FILXMAT9 except that it is passed through the utility program TIDY. See p. 37 in Ref. 1.

Subroutine GENSOL (L_L, L_2, L_3, A, B, C, D, XMAT9) – There are no changes, except that Subroutine GENSOL is passed through TIDY. See pp. 39-41 in Ref. 1.

Function ISITOK, (XR, YR, ZR, XI, YI, Z, RAWXMIN, RAWYMIN, RAWZMIN, RAWXMAX, RAWYMAX, RAWZMAX, YES, NO, KR, IR, SCALE) – Function ISITOK has been modified. Only two tests are now made on the point of intersection sent over from CROSEC. (1) Were its raw (i.e., X, Y, Z) values within the HOXYMIN and HOXYMAX values?; and (2) Will it fit within the 10- by 10-in. plotting square? If the tests are passed then the KR array is loaded. The point is unscaled in this version of the function because of the revised DLINE.

Comparison against previously accepted points has been eliminated to conserve storage space.

The remnants of the revised function have been passed through TIDY.

The updated flowchart is shown in Fig. 18.

Subroutine DLINE (NAME1, IP, KR, SCALE) – The changes in DLINE, since it was first described in NRL Report 7025, p. 48, are

1. The subroutine has been passed through TIDY.

2. COMMON/4/ has been eliminated in favor of an increased number of parameters.

3. The ordering is now done on unscaled x′ and y′ values.

4. The form of the test statements on DELX and DELY has been changed. (Old statements were 110 and 120; now they occur just prior to and at statement 160.)

5. Extra check for no solution in vicinity of old statement 133; now at statement 220.

6. Some changes in the comments.

7. A local scaling function has been added and is now called just before the plotting action.

Since the changes are of a minor nature, though important, the flowchart has not been redrawn. Refer to Fig. A9 in Ref. 1.

Segment 102

Program CYLINDER – This program represents an extension of the concepts developed in Ref. 1 for searching the defined symbol table for a canonical form of a given type. Here it is the canonical forms for cylinders and spheres that are being sought. The program is called from PRCNTL as Segment 2 of Overlay I. It has been preceded by the identification of a HOPE plane and the determination of its constants and conversion matrix XMAT9. This information is stored in the labeled common CROSSEC. The flowchart for Program CYLINDER is given in Fig. 19.

TL is set to 5 and represents the tool length, the height of the finite cylinder. (If not directly applicable in Section I, at least consider its introduction as preparation for Section III application, or temporary until cylinders have a defined length.)

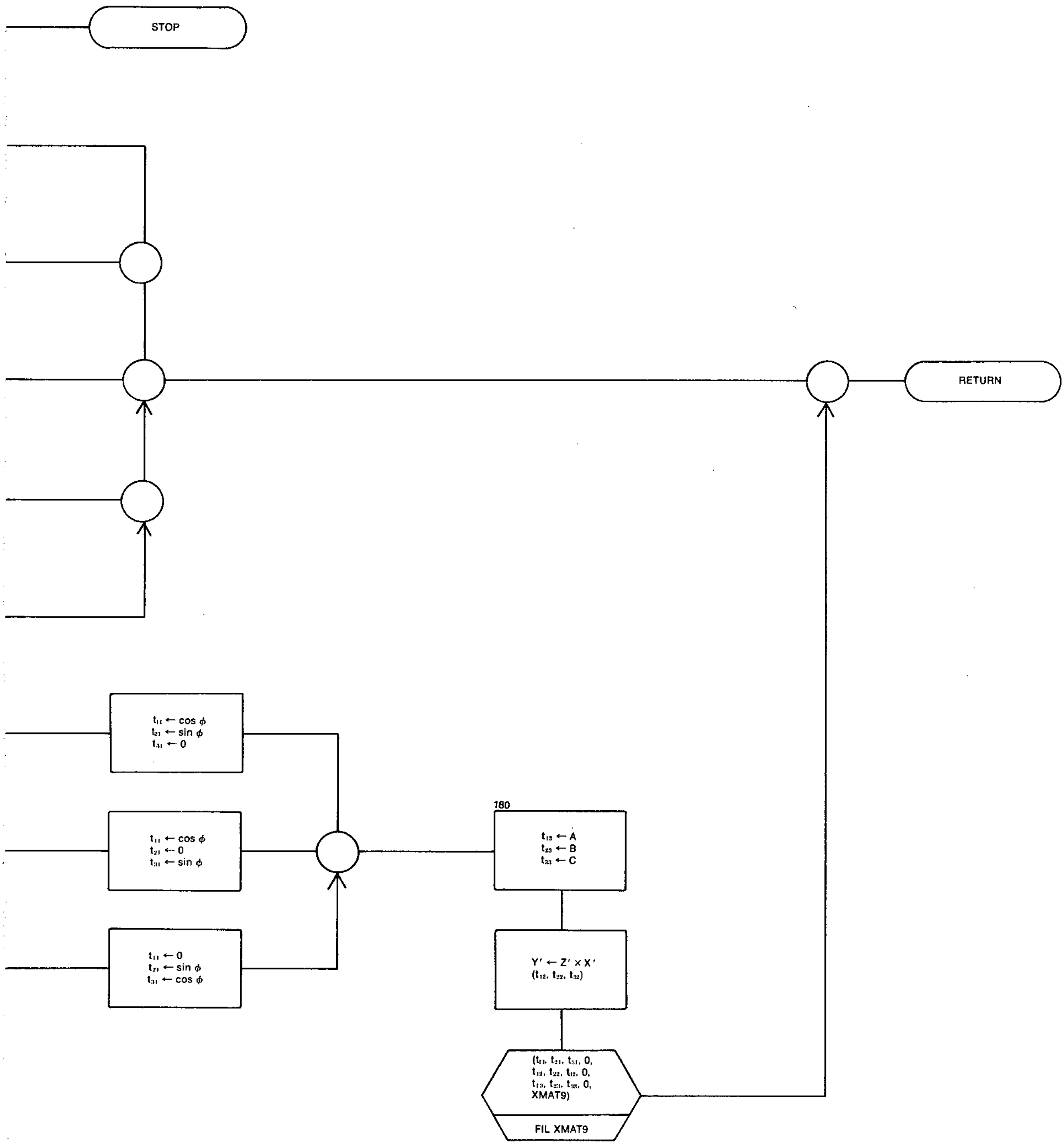
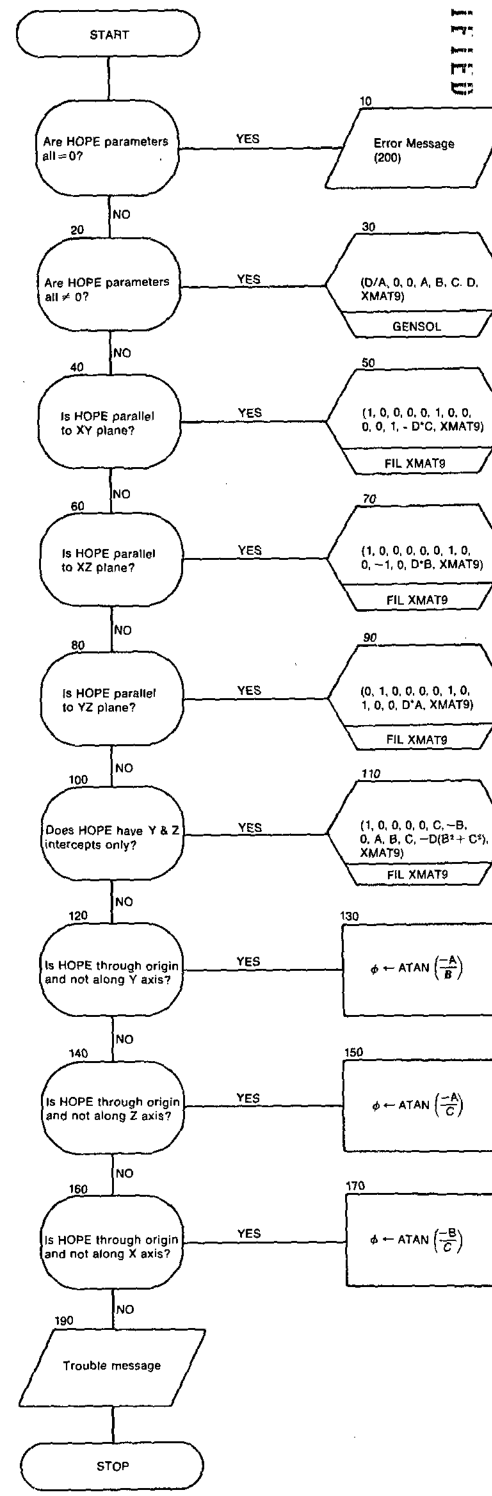


Fig. 17 - Subroutine TESTHOPE


START
Are HOPE parameters all = 0?
YES
10
Error Message (200)
NO
20
Are HOPE parameters all ≠ 0?
YES
30
(D/A, 0, 0, A, B, C, D, XMAT9)
GENSOL
NO
40
Is HOPE parallel to XY plane?
YES
50
(1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, - D*C, XMAT9)
FIL XMAT9
NO
60
Is HOPE parallel to XZ plane?
YES
70
(1, 0, 0, 0, 0, 0, 1, 0, 0, −1, 0, D*B, XMAT9)
FIL XMAT9
NO
80
Is HOPE parallel to YZ plane?
YES
90
(0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, D*A, XMAT9)
FIL XMAT9
NO
100
Does HOPE have Y & Z intercepts only?
YES
110
(1, 0, 0, 0, 0, C, −B, 0, A, B, C, −D(B² + C²), XMAT9)
FIL XMAT9
NO
120
Is HOPE through origin and not along Y axis?
YES
130
φ ← ATAN (−A/B)
NO
140
Is HOPE through origin and not along Z axis?
YES
150
φ ← ATAN (−A/C)
NO
160
Is HOPE through origin and not along X axis?
YES
170
φ ← ATAN (−B/C)
NO
190
Trouble message
STOP

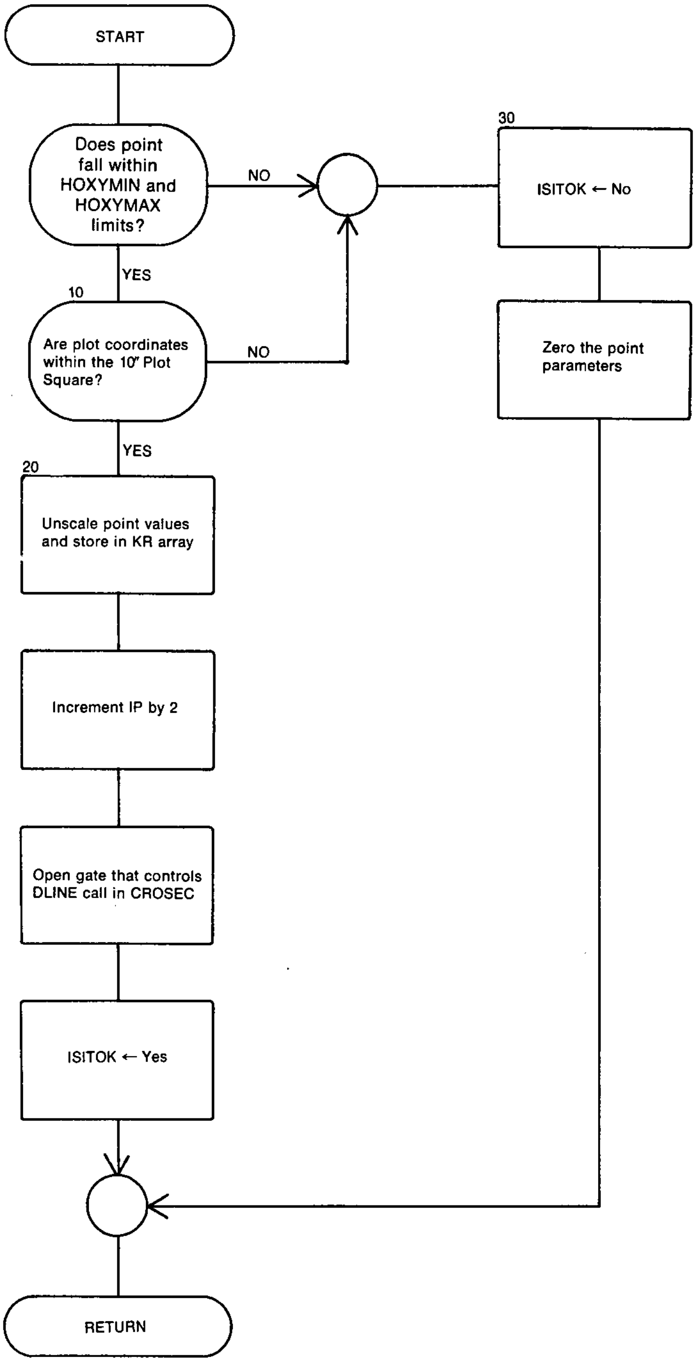


Fig. 18 - Function ISITOK

The current value for the index I is obtained from ILOOP and a two-word pair is obtained from the canonical table in the JTABL array.

If between statement 30 and 50 ISAM is found to be equal to 4 then a circle has identified, similarly a 5 identifies a cylinder, and a 13 a sphere. The circle/cylinder development continues at statement 60, while the sphere treatment takes up at statement 100.

The circle/cylinder name is stored in NAMECYL and its constants are stored as follows:

(X_c, Y_c, Z_c) for the point's coordinates

(U_x, U_y, U_z) for the vector's direction cosines,

R for the radius.

These constants are checked by Function S32TO48 for the special case of -1 while being fetched from the DEFSTO array. The axis vector is tested for authenticity. The sum of the squares of its components must be less than 0.0001 away from unity before the vector will be accepted for further processing. If the vector does not pass this test, there is an identifying printout of the circle/cylinder's name and its constants. A jump is made to the end of the loop at statement 230.

If the axis vector is acceptable then Subroutine TESTCYL is called for cylinder orientation analysis and direction cosine definition.

IANSWER is quizzed upon return from TESTCYL to determine if a local coordinate system for the cylinder has been established. If so, the processing continues at statement 160. Otherwise, a jump is made to statement 230.

The sphere treatment commences at statement 100. First its name is saved and its center and radius are fetched. Then the distance DS from the center to the HOPE plane is computed.

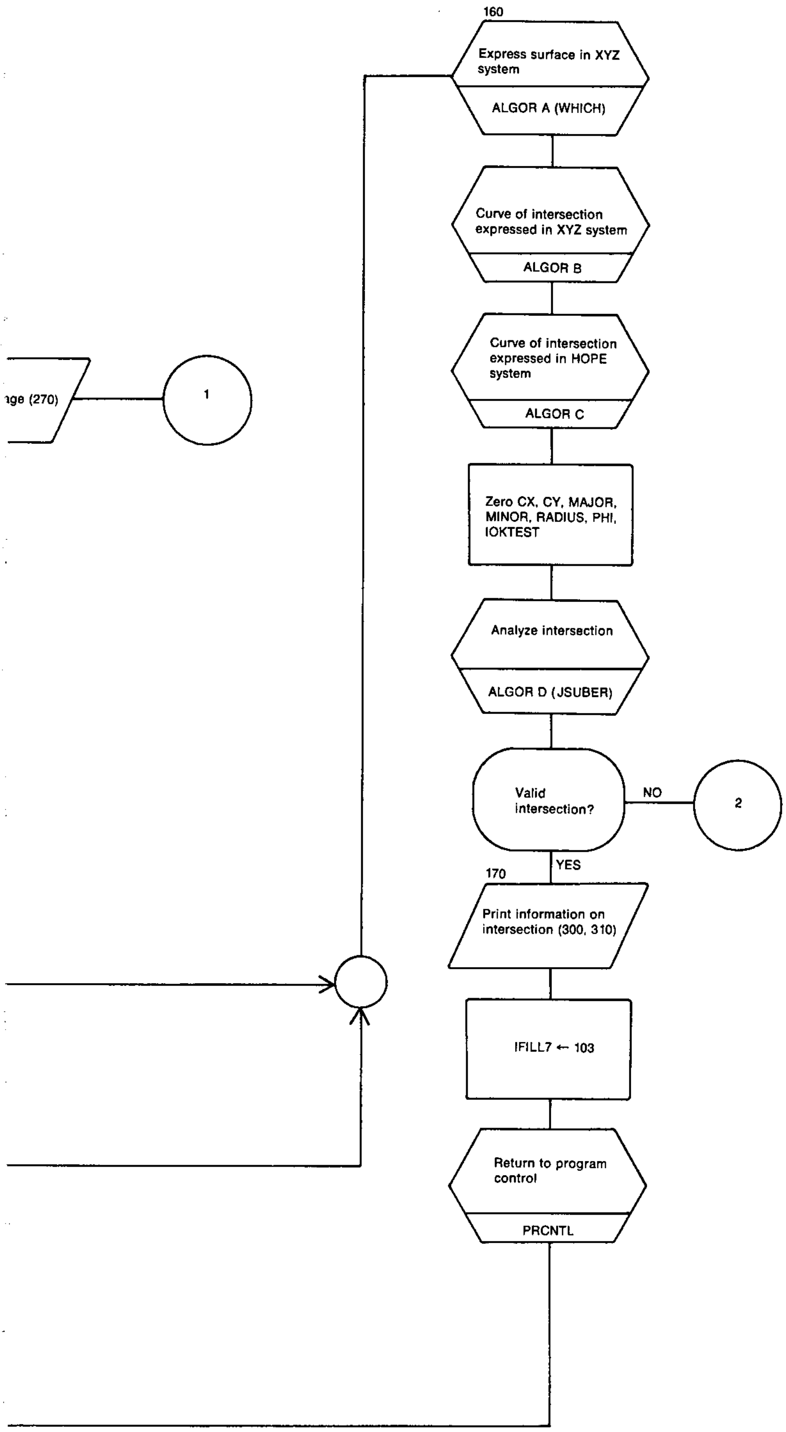
If DS is less than the radius of the sphere, then a legitimate circle of intersection is present. A jump is made to statement 150 where a standard set of direction cosines of the sphere's coordinate system is assigned via a call to FDCOSCYL. Then the flag WHICH is set equal to the word "sphere" and the information about the sphere is printed out. Processing continues for this case at statement 160.

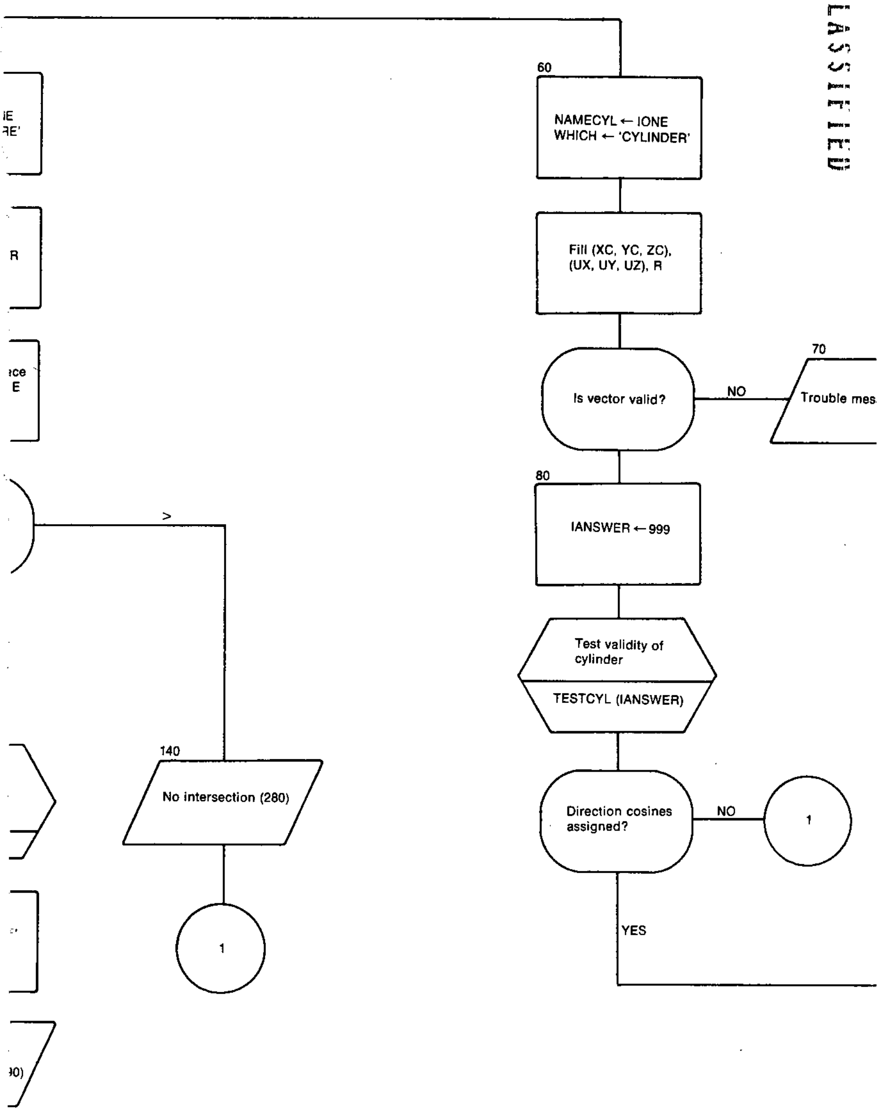
If DS is equal to the radius of the sphere then the plot is a tangent point. If this is the case the point is computed and plotted as a triangular symbol, and a jump made to statement 230.

If DS is greater than the radius of the sphere, then no circle of intersection exists for the current HOPE plane and sphere. A message to this effect is printed and a jump made to statement 230.

Processing for all three situations (circle, cylinder, or sphere) continues at statement 160 by calling on Subroutine ALGORA, which computes the coefficients needed to express the cylinder/sphere in X, Y, Z coordinates. The flag WHICH directs ALGORA to the right set of equations for this computation.

Then a call is made to ALGORB, where a simultaneous solution of the HOPE plane equation and the cylinder/sphere equation is performed resulting in the curve of intersection in the X, Y, Z system. A call to ALGORC converts the curve of intersection into coefficients in the HOPE plane coordinate system.


160
Express surface in XYZ system
ALGOR A (WHICH)
Curve of intersection expressed in XYZ system
ALGOR B
Curve of intersection expressed in HOPE system
ALGOR C
nge (270)
1
Zero CX, CY, MAJOR, MINOR, RADIUS, PHI, IOKTEST
Analyze intersection
ALGOR D (JSUBER)
Valid intersection?
NO
2
YES
170
Print information on intersection (300, 310)
IFILL7 ← 103
Return to program control
PRCNTL



CYLINDER (control for segment 2 of overlay 1)

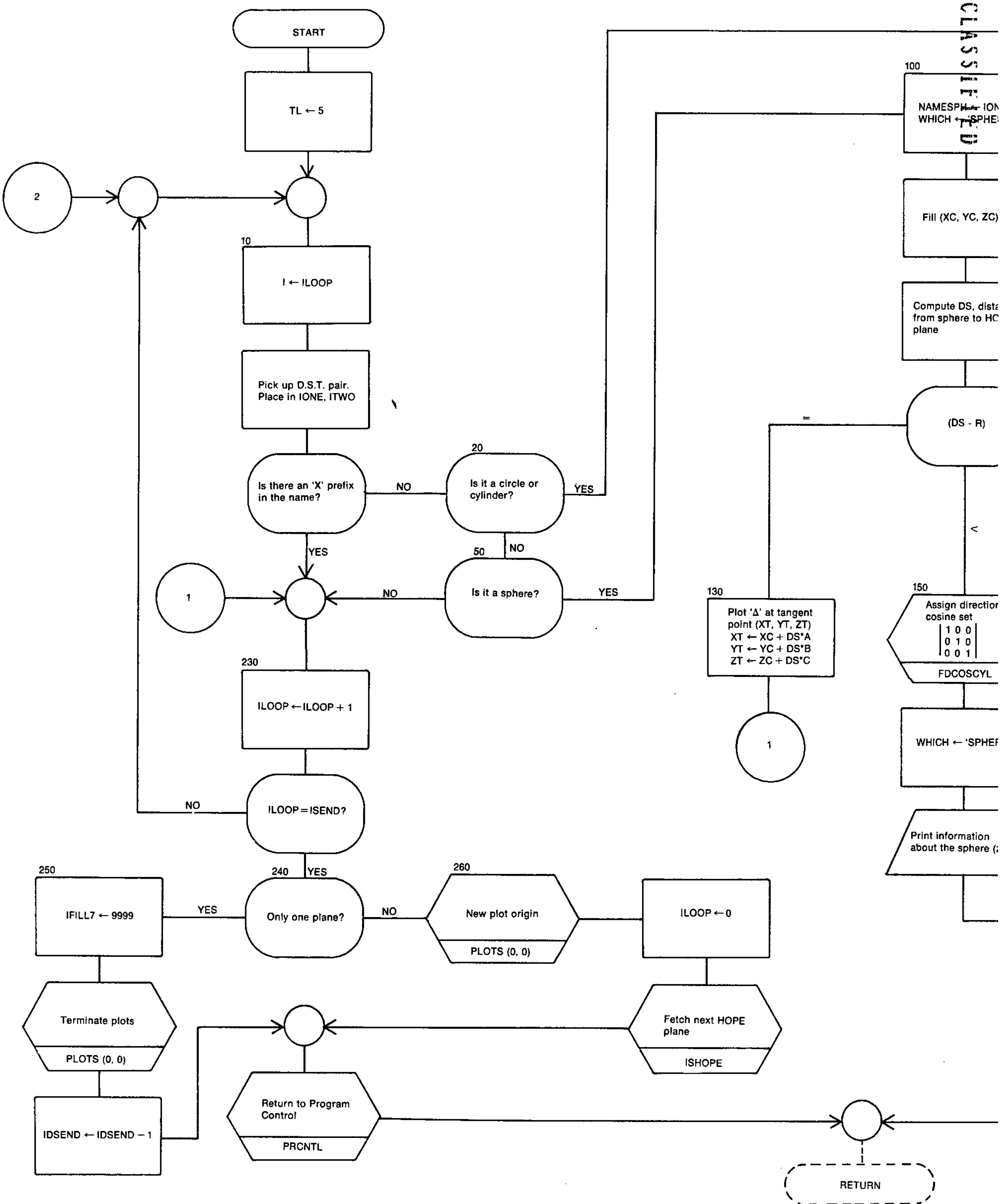


Fig. 19 - Program

After zeroing the variables involved, a call is made on Subroutine ALGORD which analyzes the curve of intersection.

Upon return from ALGORD the result of the curve analysis is contained in the variables CX, CY, MAJOR, MINOR, RADIUS, PH1, and IOKTEST, which are the center location, semimajor and semiminor axes, circle radius, angle of axis rotation, and a flag, respectively.

Storage limitations require loading a new segment by setting IFILL7 to 103 and calling PRCNTL. At statement 230, the ILOOP counter is incremented and compared with IDSEND, the length of the DST table. If the table has not been completely searched a return is made to statement 10. Otherwise, IONCE is examined to see if only one HOPE plane is present. If such is the case IFILL7 is set to 9999, the plot terminated, and PRCNTL called.

If another HOPE plane is present the plot is terminated, ILOOP is zeroed, ISHOPE is called, and the process terminated by a PRCNTL call.

Subroutine TESTCYL (XC, YC, ZC, UX, UY, UZ, R, DCOSCYL, IANSWER) – This subroutine is called from Program CYLINDER. The parameters are

XC, YC, ZC - a cylinder's defined point,

UX, UY, UZ - its axis vector's components,

R - its radius

DCOSCYL - direction cosine array (base address)

IANSWER - a Yes or No flag.

The purpose is to establish a local coordinate system for a cylinder by considering twelve possible solutions. The first six of these are "standard" solutions. If the cylinder's axis vector $\overline{U}$ is parallel to $+\overline{X}$, $-\overline{X}$, $+\overline{Y}$, $-\overline{Y}$, $+\overline{Z}$, or $-\overline{Z}$, then a predetermined set of direction cosines is assigned to DCOSCYL via a call to FDCOSCYL. Statements 110, 120, 130, 140, 150, and 160 are the IF statement comparisons with these six directions, and statements 170, 180, 190, 200, 210, and 220 are the corresponding calls on FDCOSCYL.

IANSWER returns a zero if a solution has been found.

If a standard solution is not applicable, then three possible intersection solutions are considered. These are entered by asking the questions:

Is the point at the origin or along the X axis at the same time that the vector is *not* parallel to the Y axis? If so, then do the X projection solution.

If *not*, is the point along the Y axis at the same time that the vector is *not* parallel to the Z axis? If so, then do the Y projection solution.

If *not*, is the point along the Z axis at the same time that the vector is *not* parallel to the X axis? If so, then do the Z projection solution.

Finally, three possible intercept solutions are considered which are entered by asking the questions:

Is the point off the X axis, and is the vector *not* parallel to the X axis. If so, do the X intercept solution.

Otherwise, is the point off the Y axis, and is the vector *not* parallel to the Y axis? If so, do the Y intercept solution.

Otherwise, is the point off the Z axis, and is the vector *not* parallel to the Z axis? If so, do the Z intercept solution.

If no solution has been found then the message, "NO LOCAL COORDINATE SYSTEM FOUND," is printed, IANSWER is set to 1, and a return is made.

All intersection and intercept solutions initially compute a set of direction cosines for the $\vec{X}_c$ axis using their particular criteria and formula, after which they all transfer to statement 360 for their conclusion.

At statement 360 the direction cosines for the $\vec{Z}_c$ axis are equated to those of the $\overline{U}$ vector and the $\vec{Y}_c$ axis's direction cosines are computed using the cross product of $\vec{Z}_c$ and $\vec{X}_c$.

The subroutine concludes by inserting the computed direction cosines for the three axes $\vec{X}_c$, $\vec{Y}_c$, $\vec{Z}_c$ into the DCOSCYL array by calling FDCOSCYL, setting IANSWER to 0, and returning.

Figure 20 diagrams Subroutine TESTCYL.

Function S32TO48 (AR) – The purpose of this function is to correct a deficiency in the CDC implementation of the APT system.

It has been found that a -1 parameter in an APT cylinder definition is not identifiable in a Fortran IF statement. The -1 after processing, gets stored as 5776377777777774 in octal which is -1.00000000008731 instead of 5776377777777777 in octal which is exactly -1.00000000000000.

To correct this problem the following question is asked. Is the argument equal to "5776377777777774," or is the absolute value of the argument plus 1, less than 0.00001? If so, then the fixed-pointed argument IC is "refloated," before returning the function argument, to drop the unwanted decimal portion; otherwise, the argument is merely returned. In this way the specific example and a general condition are both considered.

The name of the function stands for "Stretching from 32 bits to 48 bits." This difficulty is possibly related to an earlier version of APT with a smaller word size, if not 32 bits perhaps 36.

This function, then, is designed to improve the handling of -1 in the computer and is called from Program CYLINDER as the canonical values for the cylinder and the sphere are picked up. The function is also called from Subroutine ALGORA. The flowchart for this function is Fig. 21.

Subroutine FDCOSCYL (U1, U2, U3, U4, U5, U6, U7, U8, U9, DCOSCYL) – The purpose of this subroutine is to fill in the DCOSCYL array. There are 10 parameters, of which the first nine are direction cosines (three sets of three each), and the tenth is the array name DCOSCYL which is an acronym for "direction cosines cylinder." FDCOSCYL is an acronym for "Fill DCOSCYL." The call for this subroutine is made from Subroutine TESTCYL. Figure 22 diagrams this subroutine.

Subroutine ALGORA (WHICH) – This subroutine is called from Program CYLINDER. Its purpose is to compute the coefficients of the second-degree equation that defines either a cylinder or a sphere in the X, Y, Z coordinate system. The coefficients are computed as a part of the conversion from a local coordinate system to the X, Y, Z system.

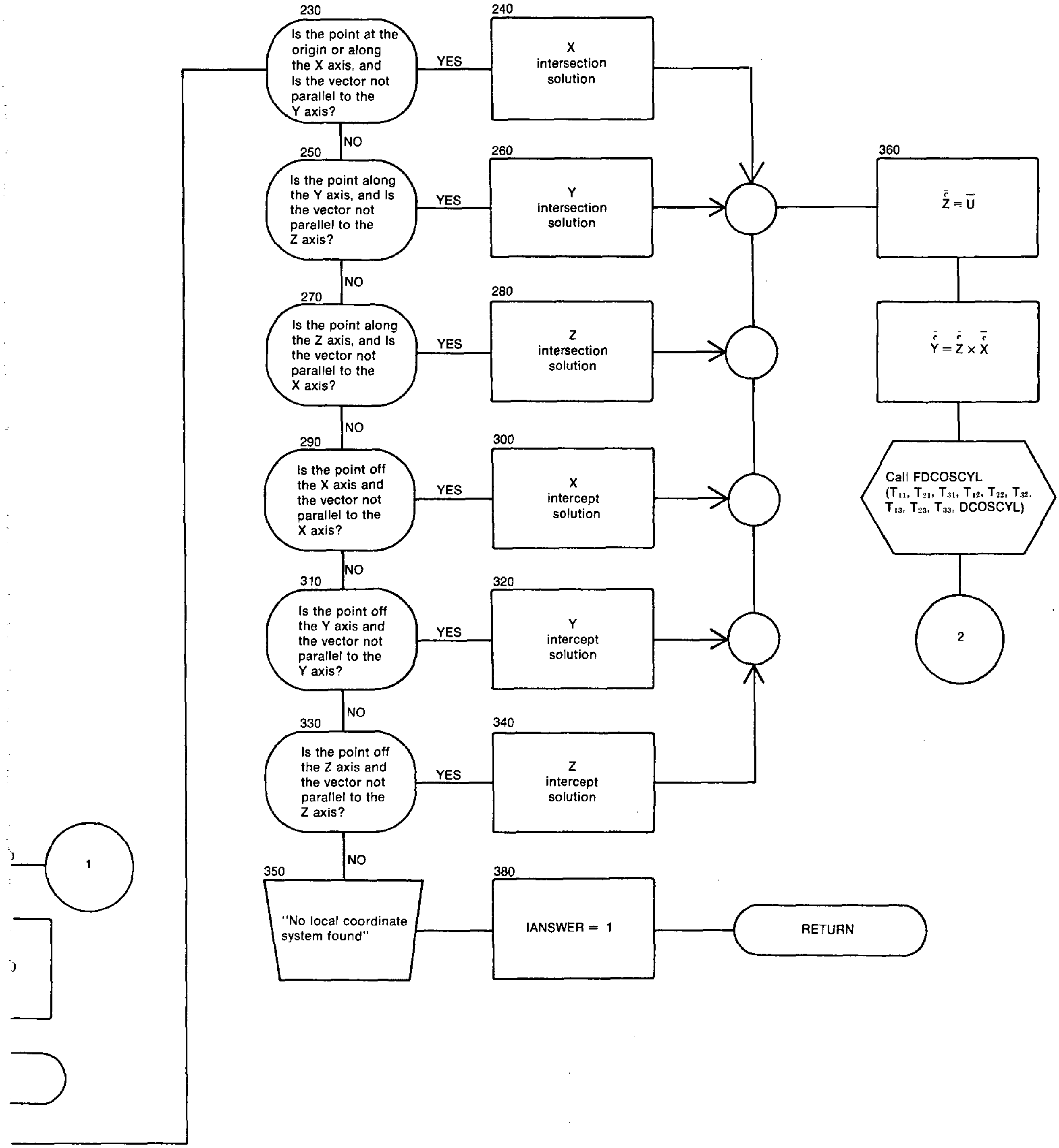
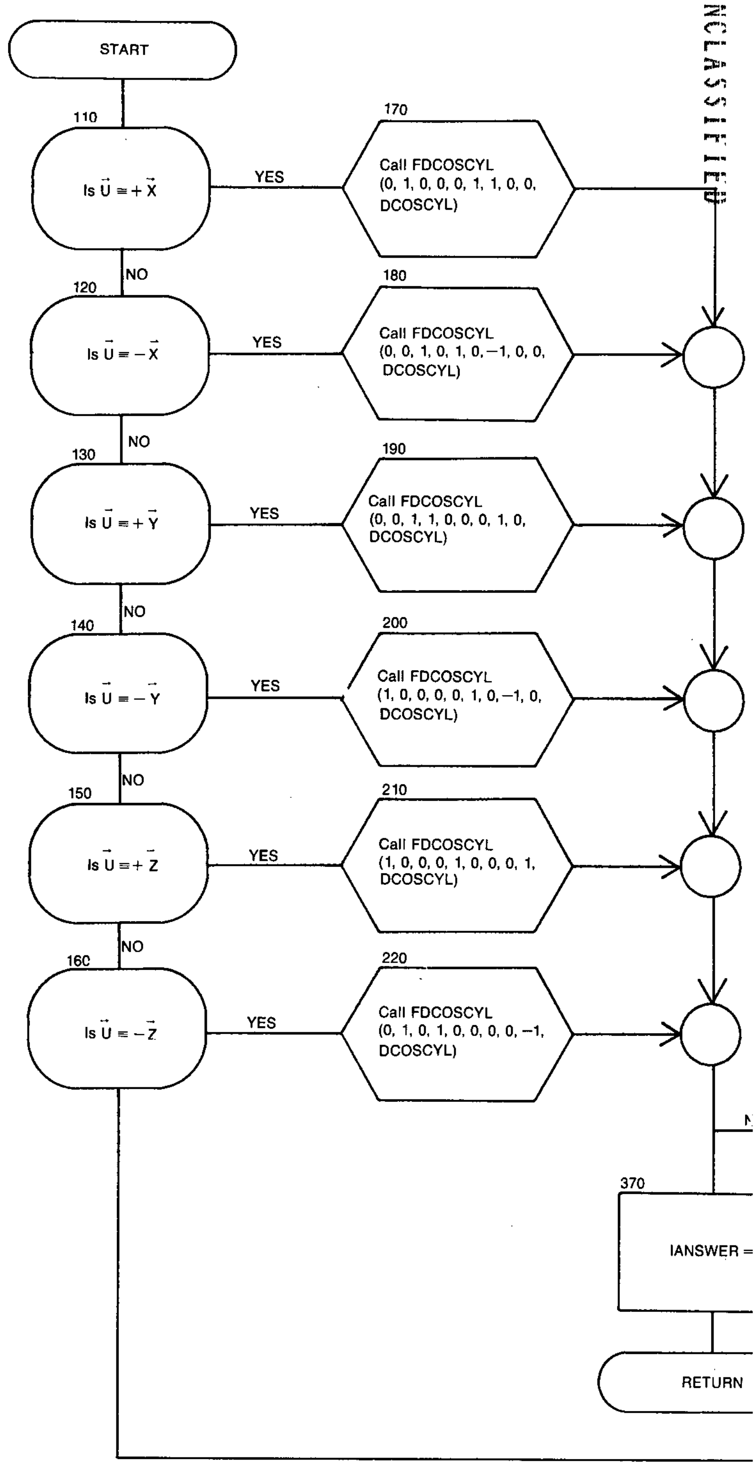


Fig. 20 - Subroutine TESTCYL


START
110
Is $\vec{U} \equiv +\vec{X}$
YES
170
Call FDCOSCYL (0, 1, 0, 0, 0, 1, 1, 0, 0, DCOSCYL)
NO
120
Is $\vec{U} \equiv -\vec{X}$
YES
180
Call FDCOSCYL (0, 0, 1, 0, 1, 0, −1, 0, 0, DCOSCYL)
NO
130
Is $\vec{U} \equiv +\vec{Y}$
YES
190
Call FDCOSCYL (0, 0, 1, 1, 0, 0, 0, 1, 0, DCOSCYL)
NO
140
Is $\vec{U} \equiv -\vec{Y}$
YES
200
Call FDCOSCYL (1, 0, 0, 0, 0, 1, 0, −1, 0, DCOSCYL)
NO
150
Is $\vec{U} \equiv +\vec{Z}$
YES
210
Call FDCOSCYL (1, 0, 0, 0, 1, 0, 0, 0, 1, DCOSCYL)
NO
160
Is $\vec{U} \equiv -\vec{Z}$
YES
220
Call FDCOSCYL (0, 1, 0, 1, 0, 0, 0, 0, −1, DCOSCYL)
370
IANSWER =
RETURN

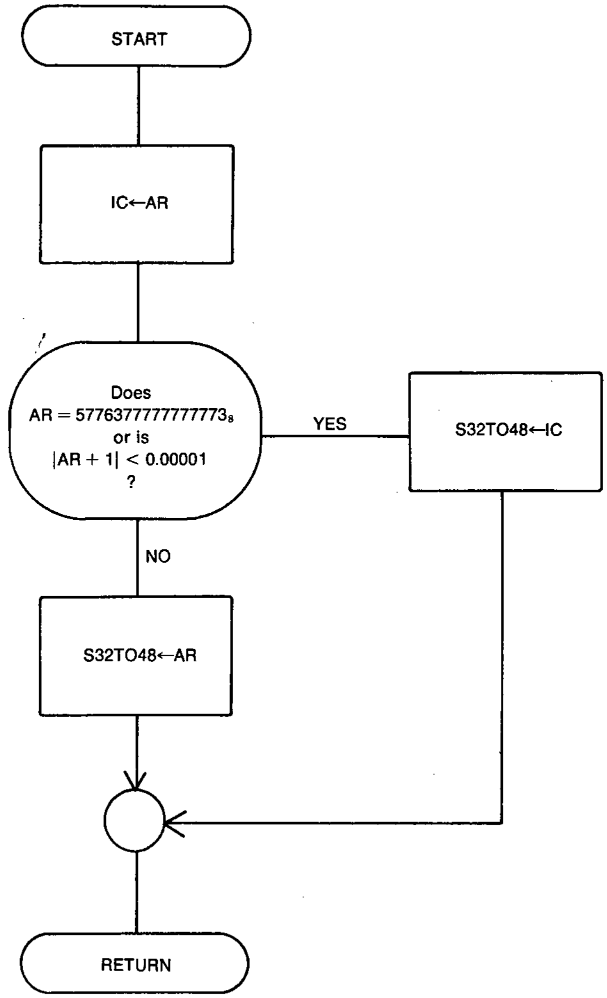


Fig. 21 - Function S32TO48

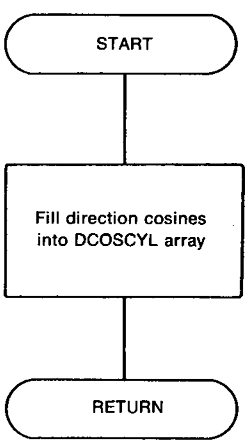


Fig. 22 - Subroutine FDCOSCYL

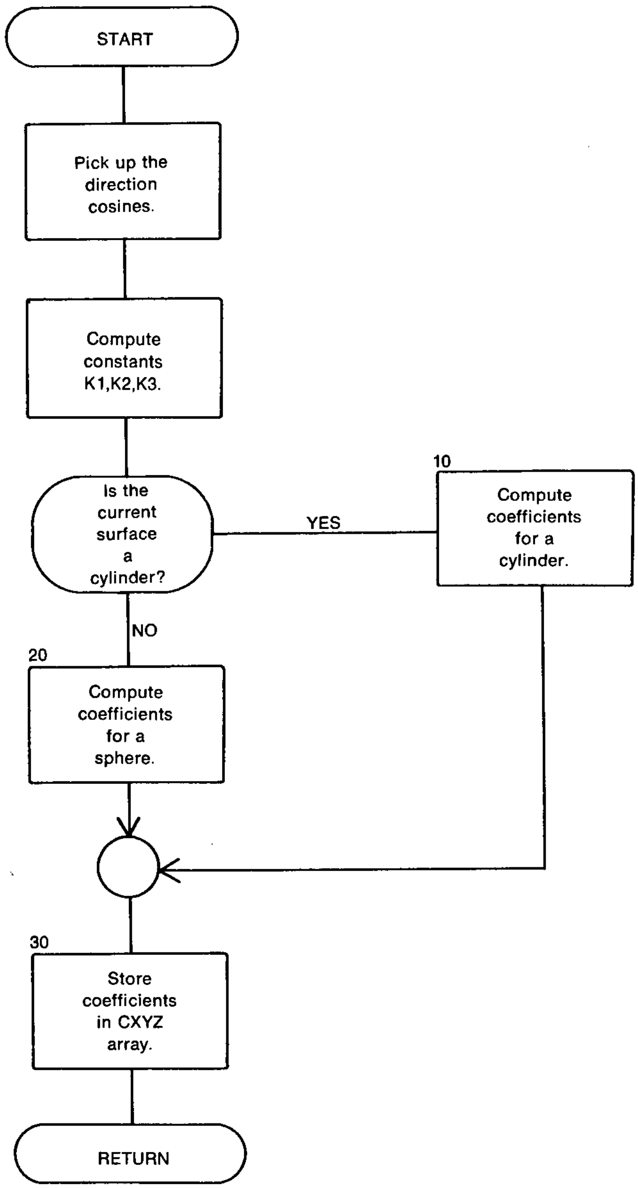


Fig. 23 - Subroutine ALGORA

The variables from CROSSEC common pertinent to this subroutine are

The point (X_c, Y_c, Z_c),

The radius r,

DCOSCYL, the direction cosines of the cylinder/sphere, and

CXYZ, the array of computed coefficients.

The parameter WHICH contains a Hollerith code word, either "CYLINDER" or "SPHERE."

NRL Report 7202 derives the equations used in this subroutine.

The direction cosines are stretched by Function S32TO48 in the replacement from DCOSCYL to the "T" notation with subscripts.

Three constants K1, K2, K3 are defined.

A branching is made on the value of WHICH. Statement 10 commences the computation of the cylinder coefficients, and statement 20 for those of the sphere. (It is easily noticed that the equations for the sphere involve one more term than those for the cylinder. This is related to the fact that the sphere needs one more dimension explicitly defined in its local coordinate system equation.)

The subroutine closes by storing the coefficients into the CXYZ array.

Figure 23 describes this subroutine.

Subroutine ALGORB – This subroutine is called from Program CYLINDER. Its purpose is to combine coefficients from the HOPE plane equation with the coefficients of the cylinder/sphere created in ALGORA. The coefficients resulting from this combination define the curve of intersection in the XYZ system.

The variables from CROSSEC common pertinent to this subroutine are

The CXYZ array with the cylinder/sphere coefficients,

The HOPE plane constants A, B, C, D, and

The EXYZ array for storing the intersection's coefficients.

In essence those four replacements take place

$$A_{14} \leftarrow A_{14} + \frac{A}{2},$$

$$A_{24} \leftarrow A_{24} + \frac{B}{2},$$

$$A_{34} \leftarrow A_{34} + \frac{C}{2},$$

and

$$A_{44} \leftarrow A_{44} - D,$$

for the single power coefficients in X, Y, Z and the constant term, respectively.

Coefficients are transferred between the arrays, even if no changes are necessary, for bookkeeping convenience.

Figure 24 corresponds to this subroutine.

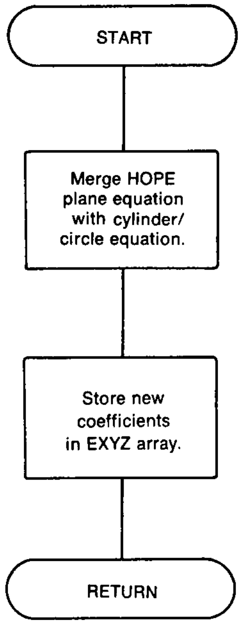


Fig. 24 - Subroutine ALGORB

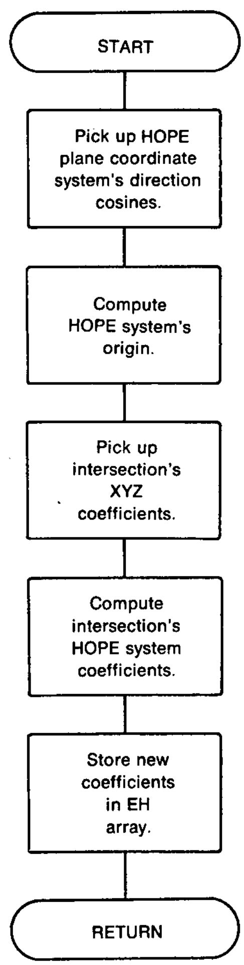


Fig. 25 - Subroutine ALGORC

Subroutine ALGORC – This subroutine is called from Program CYLINDER. Its purpose is to convert a curve of intersection from XYZ coordinates into HOPE coordinates. The curve of intersection results from the cutting of a cylinder or a spherical surface by the HOPE plane.

The elements of CROSSEC common that are pertinent to this subroutine are

EXYZ, the XYZ system coefficient array which was filled in ALGORB;

XMAT9, the direction cosine array for the HOPE coordinate system;

A, B, C, D, the HOPE plane constants; and

EH, the HOPE system coefficient array that is filled by this subroutine.

The programming commences with the direction cosines from the XMAT9 array replaced into an H-prefix series of two-digit subscripts (11, 21, 31, 12, 22, 32, 13, 23, 33). Similarly, the EXYZ array coefficients are replaced into an A-prefix series of two-digit subscripts (11, 12, 13, 14, 21, 22, 23, 24, 31, 32, 33, 34, 41, 42, 43, 44).

The HOPE plane origin is defined in the X,Y, Z system as $(X_0, Y_0, Z_0) = (D*A, D*B, D*C)$.

Then the computation of the HOPE system coefficients takes place using the series of equations developed in NRL Report 7202. These coefficients are identified by the prefix P and the 16 sets of coefficients identical to those of the A series already given. The equations involve differing combinations of the H series, A series, and origin terms.

The subroutine concludes by storing the P series of coefficients into the EH array.

A flowchart of this subroutine is shown in Fig. 25.

Subroutine ALGORD (JSUBER, NAMECYL) – This subroutine is called from Program CYLINDER, and its purpose is to analyze the curve of intersection as it is expressed in HOPE coordinates. The input consists of the EH array (filled in ALGORC) in the CROSSEC common. Output variables are described below. (The input parameter JSUBER is an APT error indicator used by Subroutine QUAD that solves quadratic equations and is found in Section I common.)

The first phase of the subroutine identifies nine elements of the EH array in an "E" prefix series with a two-digit subscript (11, 12, 13, 21, 22, 23, 31, 32, 33). This accomplishes the elimination of all Z′ terms, since the curve of intersection is to be treated in two variables only, X′ and Y′.

Next, the three invariants for a plane, second-order curve are computed using the E series of coefficients. The invariants are used to determine if a degenerate curve is present, and if not, whether the curve is a circle or an ellipse.

The coordinates for the center of the curve are computed and stored in the variables CX, CY.

A quadratic solution is made of the curve's characteristic equation obtaining the two Lambda roots, L1 and L2, $(L1 \geq L2)$. If these roots are equal a circle exists, and a jump is made to statement 340. Otherwise an ellipse is present and the flow is at statement 120.

For a circle the radius is computed.

For an ellipse the semimajor and semiminor axes are computed and stored in MAJOR and MINOR, respectively. The angle of rotation PH1 between the HOPE X′ axis and the ellipse's major axis is also computed.

The concluding portion of the subroutine deals with the setting of certain flags. If a successful solution for either a circle or ellipse has been obtained, IOKTEST is set to 0. If there has been a failure, IOKTEST is set to 1. To return the information that a *circle* has *not* been obtained, RADIUS is set equal to 99999. To return the information that an *ellipse* has *not* been obtained, MAJOR is set equal to 99999.

A flow diagram of the subroutine is shown in Fig. 26.

Subroutine HOPARCYL (LINE, HO, TL, AP, SCALE, A, B, C, D, XMAT9, XC, YC, ZC, UX, UY, UZ, R, NAMECYL) – This subroutine is called from ALGORD if the HOPE plane is parallel to the axis of the cylinder. If the HOPE plane intersects the cylinder while parallel, this subroutine will plot either a single line or a rectangle, depending on the circumstances, to represent a finite length cylinder.

The parameters are

LINE returns a success or failure message;

HO, intersection coefficient array;

TL, length of cylinder;

AP, the invariant which when equal to zero indicates a single line solution and when less than zero a pair of parallel lines;

SCALE, the plotting scale computed in Program CROSEC;

A, B, C, D, the HOPE plane constants;

XMAT9, the HOPE plane coordinate system array;

XC, YC, ZC, UX, UY, UZ, R, the cylinder point and axis vector, and radius, and

NAMECYL, the name of the cylinder for output.

The action begins by printing the name of the cylinder as the first part of the output message, the rest of which is printed later on in the program.

The intersection coefficients are picked up with an A series of subscripts (11, 12, 13, 21, 22, 23, 31, 32, 33), and the DD distance from the cylinder's center point to the HOPE plane is computed. If the absolute value of DD is greater than the cylinder's radius, a "NOLINE" message is given to "LINE," followed by a return. Otherwise the projection of the cylinder's axis, "TL units long" onto the HOPE plane is computed in terms of the HI and LO points in HOPE coordinates.

A major branching occurs when AP is tested. If AP is less than zero, the special parallel lines solution commences at statement 110; otherwise, the special single line solution proceeds at statement 190 (provided AP is zero). If AP is neither less than zero or equal to zero, a skewed line(s) solution possibility is examined starting at statement 310.

The special parallel lines solutions considered beginning at statement 110 are those parallel to the X′ axis or the Y′ axis symmetrical about either the origin or some other point. (See NRL Report 7202, for equation detail.) All of the possibilities are considered in statements 120, 150, 160, and 180. Subsequent to statements 130, 140, 170, and 180, there are four possible calls to ENDRAW to plot the properly scaled, TL-in.-long rectangle, followed by a conclusion to the output message which gives the line coordinates

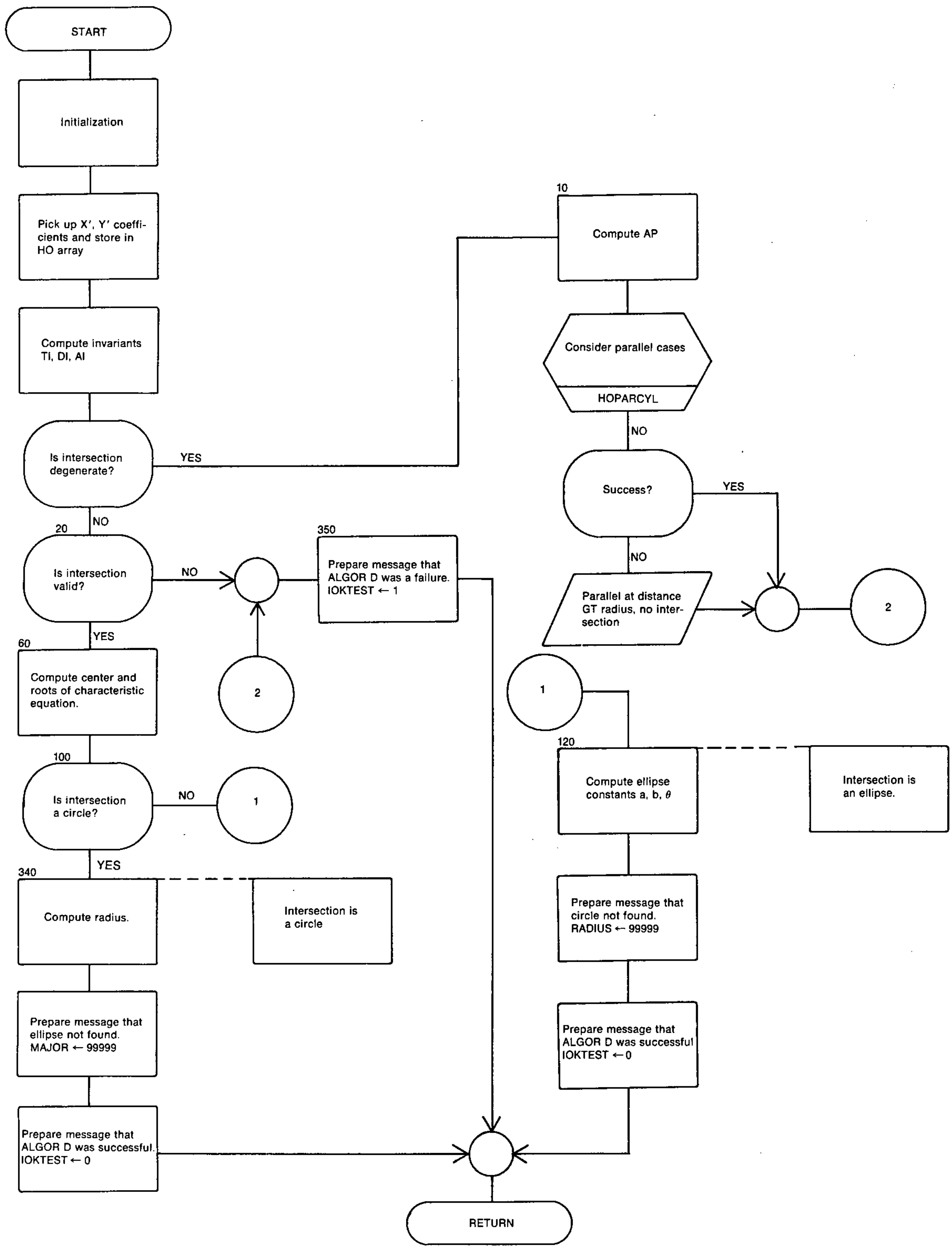


Fig. 26 - Subroutine ALGORD

and an unconditional jump statement 300, where "LINE" is given "YESLINE" and the return is made. This is the common exit for all successes.

The special single line solutions considered commencing at statement 190 are single horizontal or vertical lines passing through either the X'Y' origin or some other point. All of the possible conditions are examined in statements 200, 210, 230, 250, 260, and 280. There are four possible calls to DRAW at statements 220, 240, 270, and 290, where a single line is drawn followed by the conclusion described above.

Commencing at statement 310 the skewed line possibilities are considered. If a single line is present, it is plotted between the "LO" and "HI" points via a call to DRAW at statement 320. If a skewed, parallel set is present, the slopes and intercepts for the rectangle are computed and plotted. The skewed solutions conclude in the manner described above.

The simplified flowchart is given in Fig. 27.

Subroutine DRAW – This subroutine is called from Subroutine HOPARCYL and has two x', y' points as parameters which are so oriented as to be either horizontal or vertical with respect to the X', Y' system. If entry is made through DRAW a single line is plotted, whereas if entry is made through ENDDRAW a rectangle is plotted. The flowchart is given in Fig. 28.

Segment 103

Program DISCRETE — The cylinder/sphere treatment continues in Segment 3 of Overlay 1. The variables filled in ALGORD are examined. If an improper curve of intersection is present IOKTEST will be 1; otherwise, 0. If a curve is present then it is necessary to distinguish between a circle and an ellipse. If the curve is a circle then MAJOR will be equal to 99999. If the curve is an ellipse then RADIUS will be equal to 99999. If these tests fail an error message is printed and transfer made to statement 140.

If a circle is present its points are calculated by calling on Subroutine POINTSC, followed by a translation correction.

If an ellipse is present its points are calculated by calling on Subroutine POINTSE, followed by a translation-rotation correction.

In either case the set of points is stored in the XS, YS arrays with ISET specifying the number of points.

The circle line of flow transfers to statement 130 where the points are plotted. The line of flow for the ellipse is more involved. The end points of the major axis are computed in the cylinder coordinate system and used in a series of tests. The heights above the base plane are D1Z and D2Z.

First it is determined which end of the major axis is lower, the positive end or the negative end. This information is stored in IWHIPART and used as a parameter in the ELBOUNDS calls made further on.

Next the extent of truncation is determined. If the ellipse lies totally within or totally without the finite cylinder, then no truncation is performed;

$$\text{i.e., } \{(D1Z<0 \wedge D2Z<0)^{\vee} \\ (D1Z \geqslant TL \wedge D2Z \geqslant TL)^{\vee} \\ (D1Z>0 \wedge D1Z \leq TL \wedge D2Z>0 \wedge D2Z \leq TL)\}$$

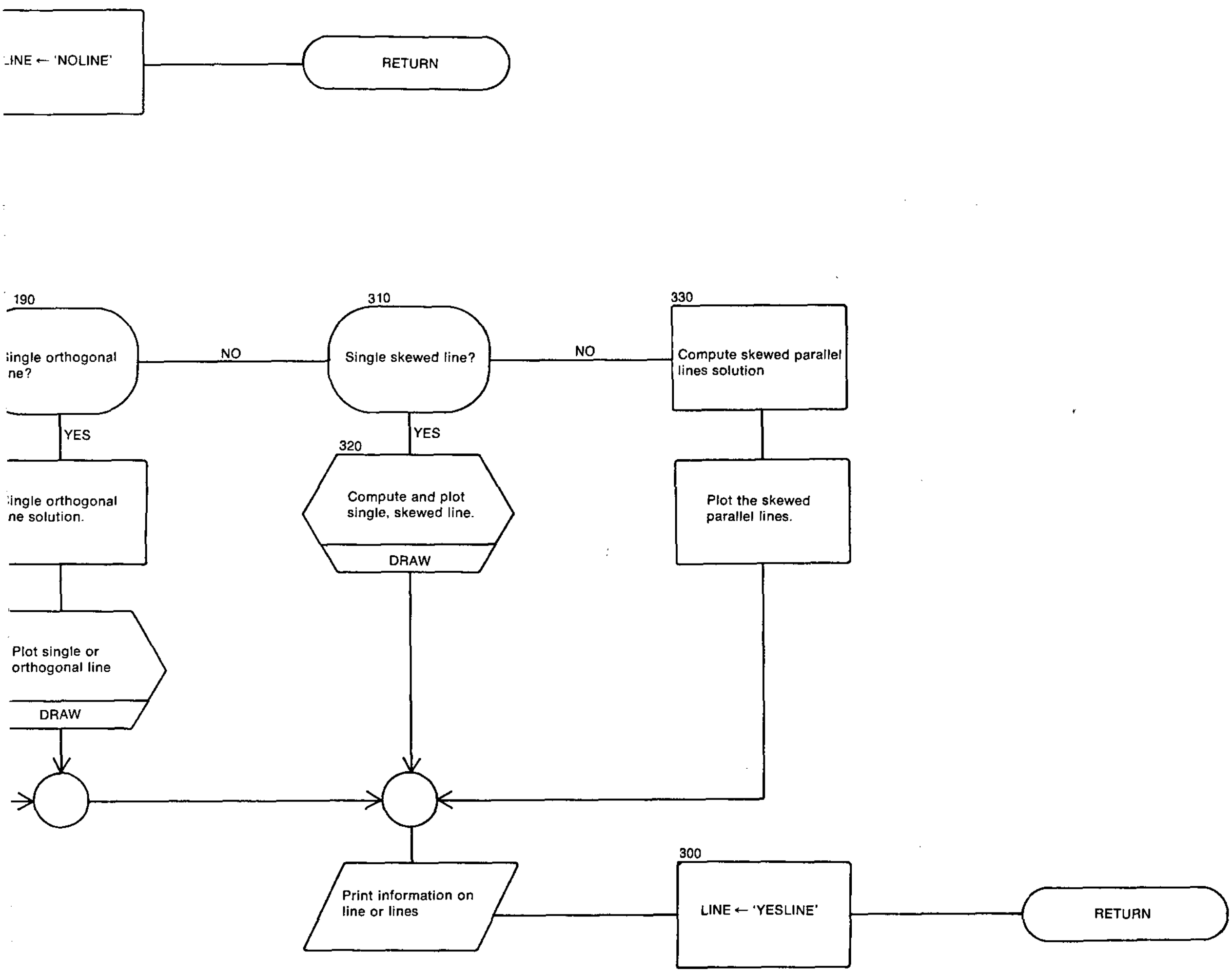
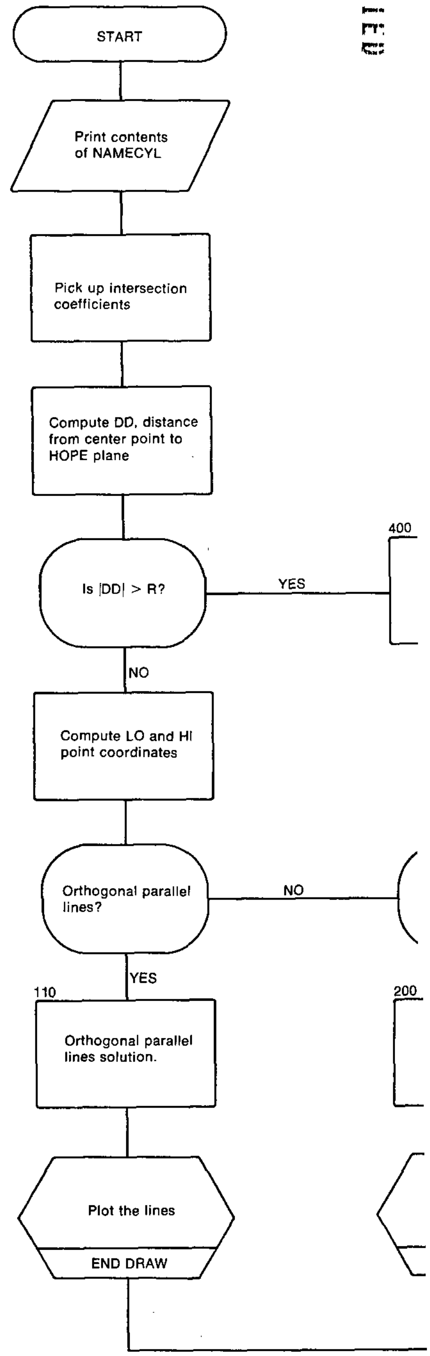


Fig. 27 - Subroutine HOPARCYL


START
Print contents of NAMECYL
Pick up intersection coefficients
Compute DD, distance from center point to HOPE plane
Is |DD| > R?
YES
400
NO
Compute LO and HI point coordinates
Orthogonal parallel lines?
NO
YES
110
200
Orthogonal parallel lines solution.
Plot the lines
END DRAW

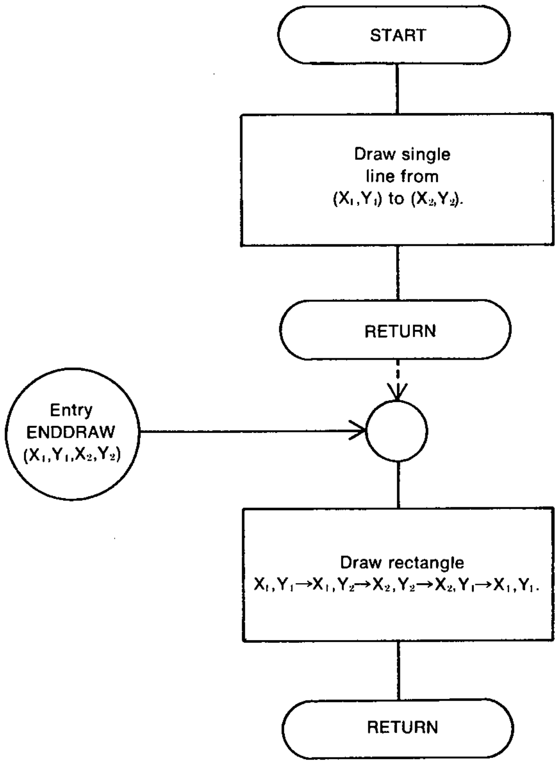


Fig. 28 - Subroutine DRAW

Only bottom base plane truncation is needed if the following set of conditions is true:

$$\{(D1Z<0 \wedge D2Z>0 \wedge D2Z \leq TL) \vee (D2Z<0 \wedge D1Z>0 \wedge D1Z \leq TL)\}.$$

Only top base plane truncation is needed if the following set of conditions is true:

$$\{(D2Z>TL \wedge D1Z>0 \wedge D1Z \leq TL) \vee (D1Z>TL \wedge D2Z>0 \wedge D2Z \leq TL)\}.$$

Truncation by both planes is needed if the following set of conditions is true:

$$\{(D1Z<0 \wedge D2Z>TL) \vee (D1Z>TL \wedge D2Z<0)\}.$$

NOTE - The finite cylinder is positive TL units long. Above the bottom base plane is its positive direction, below is negative with respect to the cylinder's Z axis. For an end point to be "in" the cylinder, its end distance (D1Z or D2Z) must be positive and less than or equal to TL.

Truncation of the ellipse is accomplished by making appropriate calls to ELBOUNDS. For a bottom base plane truncation JE is coded 0, and (X_c, Y_c, Z_c) are sent over, while for a top base plane truncation JE is coded 1 and (X_{t1}, Y_{t1}, Z_{t1}) are sent over. Additional parameters are JSUBER (an APT trouble flag for the subroutine that solves a quadratic equation), and IWHIPART. JE returns with a YES or NO answer regarding the success or failure of the truncation.

If JE is NO the plotting is bypassed and transfer made to statement 140.

The discrete set of points defining the curve of intersection is plotted by calling Subroutine PLOTSET, followed by a jump to statement 140.

At statement 140 the ILOOP value is incremented and compared with IDSEND. If ILOOP < IDSEND the canonical table has not been completely searched for cylinders/spheres, so Segment 2 of Overlay 1 is reloaded by setting IFILL7 to 102 and calling PRCNTL.

If ILOOP = IDSEND, then the table search with the present HOPE plane has finished. If only one HOPE plane is defined, INCHOPE will contain the word "ONCE" so further Section I processing is terminated by setting IFILL7 to 9999, terminating the plot and calling PRCNTL. Otherwise, ISHOPE is called to find the next HOPE plane, the plot zeroed and Program DISCRETE terminated with a call to PRCNTL.

Program DISCRETE's flowchart is Fig. 29.

Subroutine POINTSE – This subroutine is called from Program DISCRETE if the curve of intersection is an ellipse. Its purpose is to compute the X′, Y′ coordinates for a set of points to be used in plotting the ellipse.

The CROSSEC common information used is

MAJOR, the length of the semimajor axis;

MINOR, the length of the semiminor axis; and

ISET, the number of points desired (set to 400).

Specifically, this subroutine computes a subset of points (called a "feeder set") of values for the first quadrant, stores them in the X and Y arrays, and then calls on Subroutine EXPAND to fill in the XS and YS arrays with the full set of points.

If ISET points are wanted, the feeder set will consist of M points where M = (ISET - 4)/4, and the radian increment DTH, expressed in radians, is

$$\text{DTH} = \frac{\pi/2}{(\text{M} + 1)} .$$

The eccentricity EC of the ellipse is computed as

$$\text{EC} = \sqrt{1 - \frac{(\text{MINOR})^2}{(\text{MAJOR})^2}} .$$

Computations for the feeder set assume the ellipse is centered at (0, 0) and that the first point is at (0, MINOR) and the last point at (MAJOR, 0).

The loop that computes the feeder set decrements DTH from TH which has an initial value of ((π/2) + DTH) radians and compute X′, Y′ values from the equations,

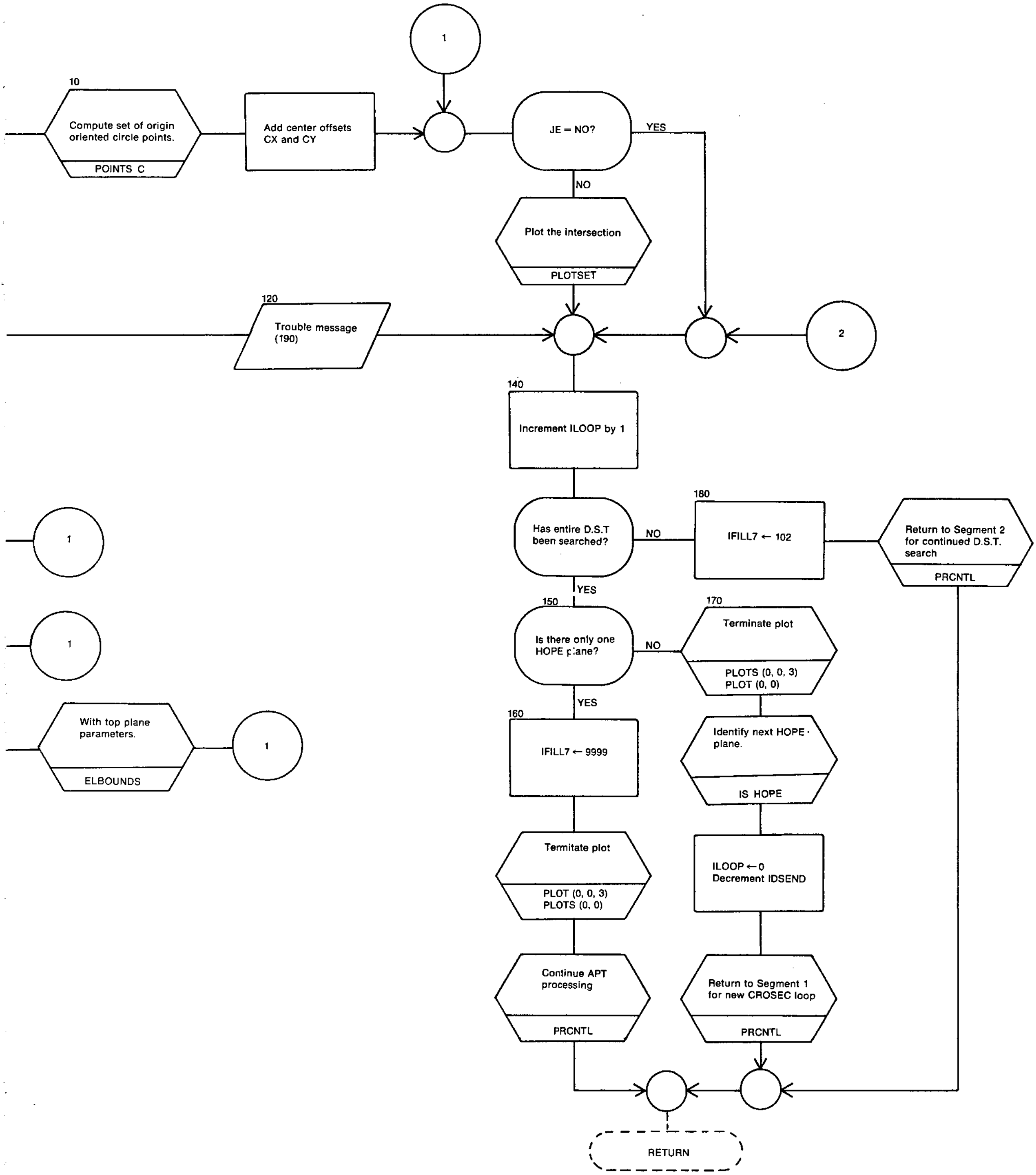
$$X' = \text{RA} \cos (\text{TH})$$

$$Y' = \text{RA} \sin (\text{TH}),$$

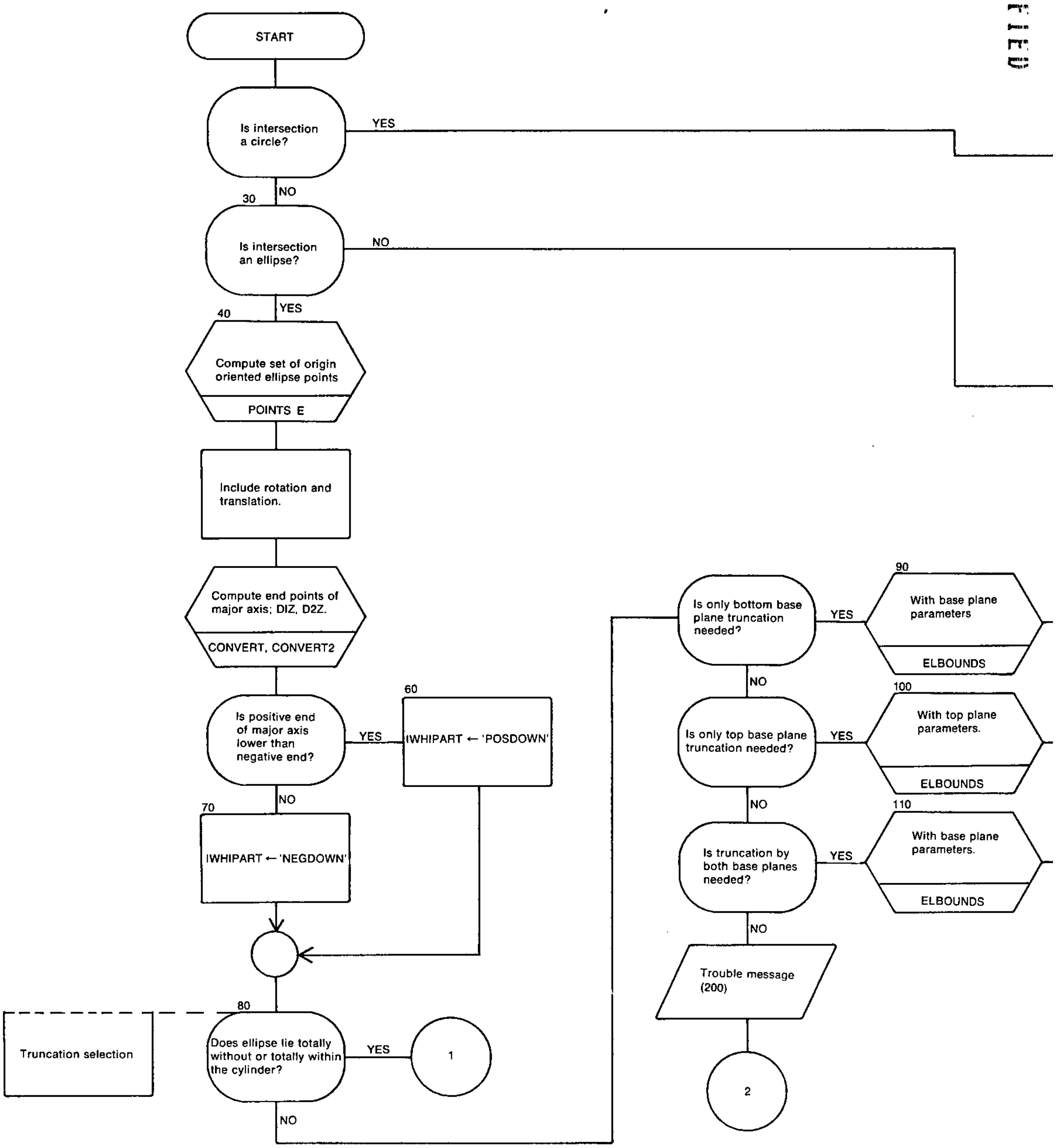
where

$$\text{RA} = \sqrt{\frac{\text{MINOR}^2}{1 - (\text{EC})^2 \cos (\text{TH})}} .$$

(See Fig. 3b in NRL Report 7202.)



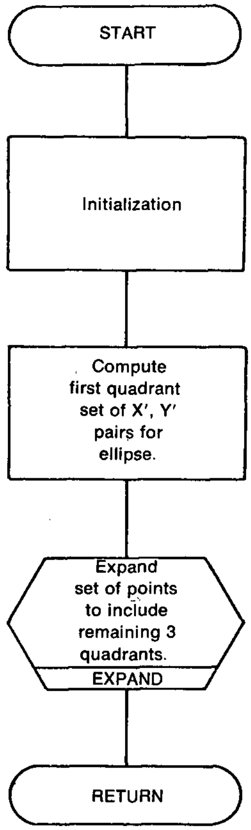
Program DISCRETE



START
Is intersection a circle?
YES
NO
30
Is intersection an ellipse?
NO
YES
40
Compute set of origin oriented ellipse points
POINTS E
Include rotation and translation.
Compute end points of major axis; DIZ, D2Z.
CONVERT, CONVERT2
Is positive end of major axis lower than negative end?
YES
60
IWHIPART ← 'POSDOWN'
NO
70
IWHIPART ← 'NEGDOWN'
Truncation selection
80
Does ellipse lie totally without or totally within the cylinder?
YES
1
NO
Is only bottom base plane truncation needed?
YES
90
With base plane parameters
ELBOUNDS
NO
Is only top base plane truncation needed?
YES
100
With top plane parameters.
ELBOUNDS
NO
Is truncation by both base planes needed?
YES
110
With base plane parameters.
ELBOUNDS
NO
Trouble message (200)
2

Fig. 29 -

Fig. 30 - Subroutine POINTSE



The feeder set is expanded into a full ISET point description of the ellipse by calling on Subroutine EXPAND with M as a parameter.

The flowchart for the subroutine is Fig. 30.

Subroutine POINTSC— This subroutine is called from Program DISCRETE. Its purpose is to compute a set of points to be used to plot a circle. Input information, contained in CROSSEC common, is

RADIUS, the radius of the circle, and

ISET, the number of points desired (set to 200).

Specifically, this subroutine computes a subset (called a "feeder set") of values for the first quadrant and stores them in the X and Y arrays, and then in turn calls on Subroutine EXPAND to fill in the XS and YS arrays with the full set of points.

If ISET points are wanted, the feeder set will consist of M points where M = (ISET - 4)/4, and the radian increment DTH, expressed in radians, is

$$DTH = \frac{\pi/2}{(M+1)} .$$

The feeder set assumes that the circle is centered at (0, 0) and that the first point is at (0, R) and its last point at (R, 0), where R = RADIUS. The loop that computes the feeder

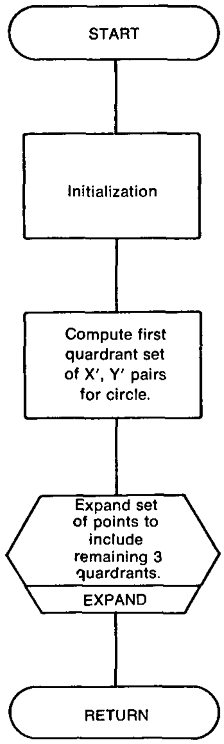


Fig. 31 - Subroutine POINTSC

set decrements DTH from TH which has an initial value of $\pi/2$ radians, and computes the x', y' values of the points from the equations

$$x' = R\cos(TH)$$

and

$$y' = R\sin(TH).$$

(See Fig. 3a in NRL Report 7202.)

EXPAND is called with parameter "M" to fill up the remaining three quadrants of points.

The flowchart for the subroutine is Fig. 31.

Subroutine EXPAND (M) – The purpose of the subroutine is to take the basic 90° set of points (the feeder set) contained in the X and Y arrays, and create a full 360° set of ISET points stored in the XS, YS arrays.

This subroutine is called either from Subroutine POINTSC or Subroutine POINTSE where the feeder set is filled. The parameter M = (ISET - 4)/4.

Pertinent elements from CROSSEC common are

The X and Y arrays containing feeder set,

The (CX, CY) HOPE coordinates for the center of the curve, and

The XS, YS arrays for the output set of points.

Table 3
The Elements in the Four DO Loops Which Fill the XS, YS Arrays Using the X,Y Feeder Set

Quadrant Number	LOOP		Sign Following		X,Y Array "N" Modifier	
	Lower Limit	Upper Limit	CX	CY	Initial Value (equivalent)	Increment
1	1	M + 2	+	+	0	0
2	M + 3	2M + 3	+	-	2	2
3	2M + 4	3M + 4	-	-	2M + 2	2
4	3M + 5	4M + 4	-	+	2M + 4	2

The subroutine consists of four DO Loops, one for each quadrant of the curve. Each time around each loop an XS value and a YS value are loaded. For the location of the center (0, 0) is used, and the feeder set provides the amount of offset needed for the point. The offset is some times additive and sometimes subtractive with respect to the center, depending on the quadrant.

Table 3 summarizes the information from which the loops are built. For each of the four quadrants, the following information is given in the table:

1. The upper and lower limits for the loop index J. J is the first term in the subscript expression for the X,Y arrays.

2. The arithmetic sign used in the arithmetic statements for XS(J) and YS(J).

3. The initial value for N, the subscript modifier which is subtracted from J to correctly identify which feeder set value is needed.

4. The increment applied to N each time around the loop.

The flow chart for this subroutine is Fig. 32.

Subroutine PLOTSET – The purpose of this subroutine is to plot the set of points in the XS, YS arrays which define the curve of intersection between the HOPE plane and a cylinder/sphere. (If an ellipse of intersection is bounded by a finite length cylinder then the XS, YS array contains a truncated set of points.)

In addition to the XS, YS array the subroutine uses SCALE from CROSSEC common. Each coordinate from XS, YS is multiplied by SCALE and then increased by 5 to obtain a plotting coordinate.

If the *first* point is outside the allowable limits a message with its X, Y values is printed and an immediate return is made. If not, the first point values are saved and the pen moved to and lowered at the first point. Then a loop is entered which tours the pen to all the points in the XS, YS arrays. Exiting the loop causes the pen, still lowered, to be moved from the last point of the array to the first point to assure that the curve is closed. The pen is then raised and the subroutine is finished. Within the loop, if the plotting values in either X or Y are less than 0 or greater than 10.0 the pen is raised at that point. (The cylinder/circle intersection is not subjected to the HOXYMIN and HOXYMAX limits in CROSEC MOD 2.0.)

The flowchart for the subroutine is given in Fig. 33.

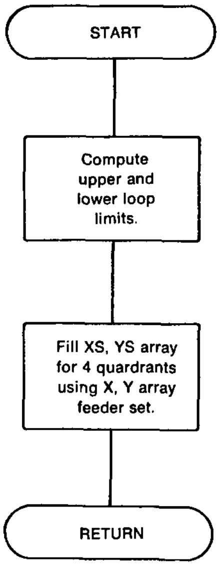


Fig. 32 - Subroutine ELBOUNDS

Subroutine ELBOUNDS (F1, F2, F3, JSUBER, JE, IWHIPART) – This subroutine is called from Program DISCRETE. Its purpose is to compute the coordinates of the clip line, and having done that to call in CLIP to perform the clipping. The clip line is computed by getting the line of intersection between a base plane and the HOPE plane; and then bounded by intersecting this line with the ellipse to get two points (X_1, Y_1) and (X_2, Y_2). With end planes that are perpendicular to the cylinder's axis it follows that the clip line is parallel to the minor axis of the ellipse.

The parameters are

F_1, F_2, F_3, an end plane axis point;

JSUBER, an error detection flag for QUAD,

JE, enters with end plane identification (1 is top, 0 is bottom) and returns with a YES or NO message regarding successful clipping, and

IWHIPART, major axis orientation information needed by CLIP.

The action commences by placing the end plane code into IPLANE, computing the HOPE origin and picking up the ellipse coefficients with an E series of subscripts (11, 12, 13, 21, 22, 23, 31, 32, 33). The coefficients of the line of intersection, P, Q, R, S, are computed by subtraction and then converted to the HOPE coordinate system.

Next it is necessary to compute the two clip points, the intersections of the truncation line with the ellipse. The question is asked, "Is the truncation line horizontal?" If so, at statement 110 the two clip point Y′ values are identical and equal to the constant term S divided by the Y coefficient Q and a quadratic solution for the X′ values of the clip points is performed. Otherwise, at statement 140 the line of intersection is expressed in terms of its slope SL with respect to the X′ axis and the intersection EW on the Y′ axis; a quadratic solution for the Y′ clip point values is performed, followed by a computation of the X′ values from the line equation.

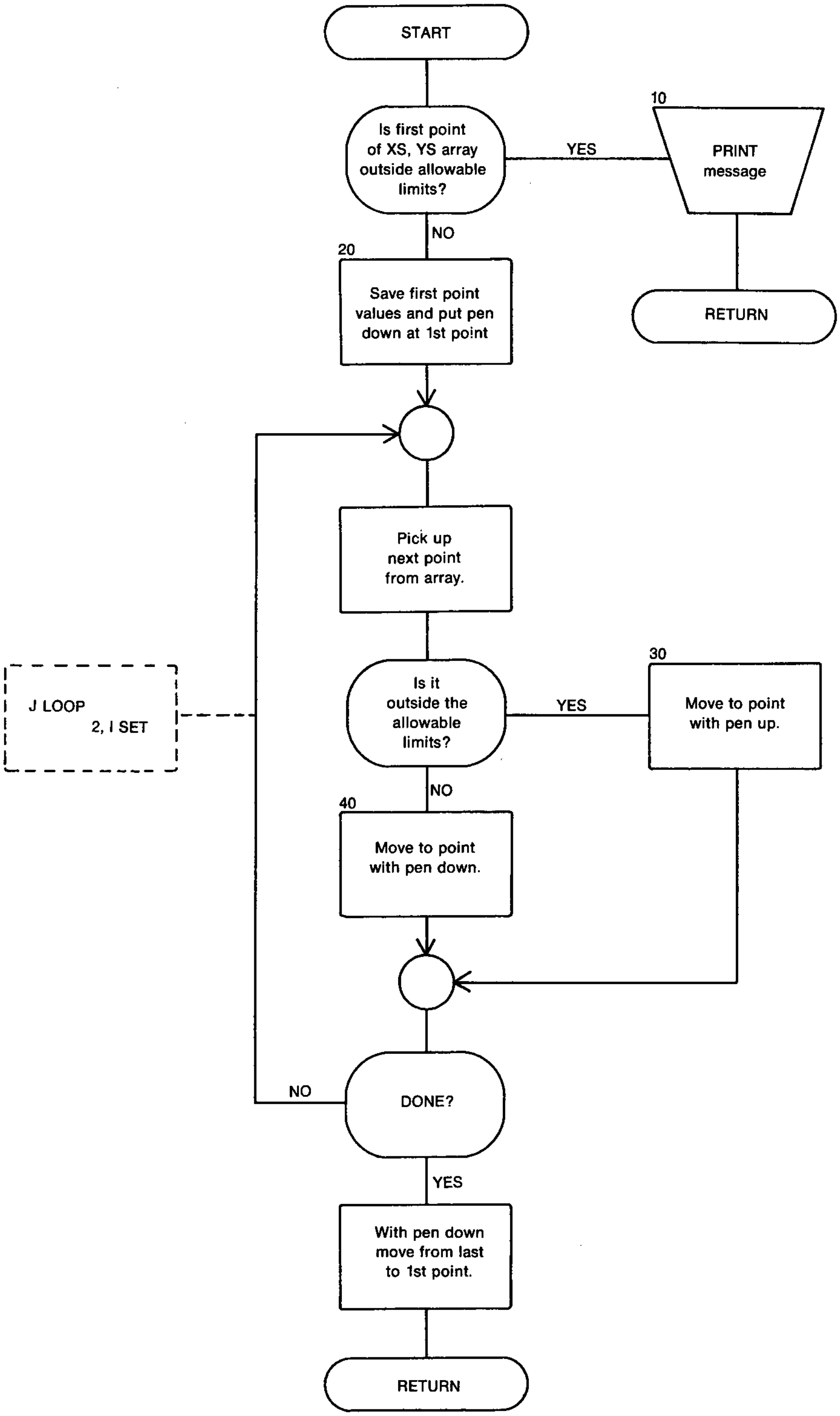


Fig. 33 - Subroutine EXPAND

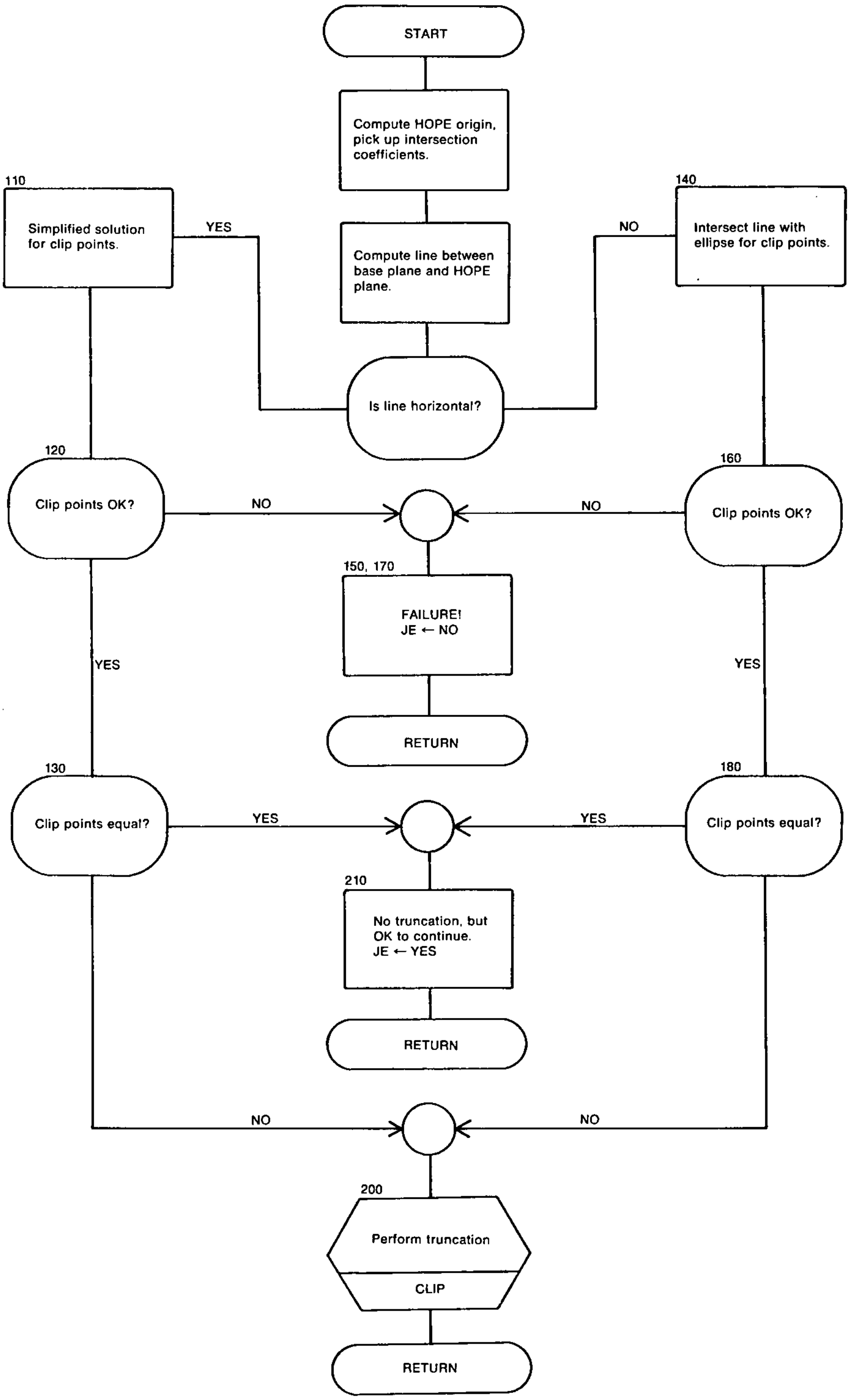


Fig. 34 - Subroutine PLOTSET

If either quadratic solution is improper, an exit is made with JE set to "NO." If the truncation line is merely tangent to the ellipse, then the two clip points are identical and an exit is made at statement 210 with JE set to "YES," signifying plotting is to continue without truncation.

If the two clip points are not equal then Subroutine CLIP is called at statement 200 followed by a return. ELBOUNDS' flowchart is shown in Fig. 34.

Subroutine CLIP (X_1, Y_1, X_2, Y_2, IPLANE, JE, IWHCPART)— The purpose of this subroutine is to create a changed set of discrete points for the ellipse. This change takes into account the presence of the bounding line called the "clip" line. The parameters (X_1, Y_1), (X_2, Y_2) define the clip line. IPLANE defines which of two end planes is being treated, JE returns a YES or NO message, and IWHCPART describes the orientation of the major axis. Pertinent variables from CROSSEC common that are used in this subroutine are the (XS, YS) arrays for the points of the ellipse and ISET, the number of points. Two local arrays (XSS, YSS) are used for temporary storage of a subset of points. This subroutine is called from Subroutine ELBOUNDS.

The (XS, YS) arrays and the clip line points (X_1, Y_1), (X_2, Y_2) are center oriented, and then the subroutine accomplishes an ordering between the clip points such that (X_1, Y_1) will always be the most positive point with respect to both the X' and Y' axes. If it is necessary to reverse the points a call is made to Subroutine REVERSE.

Next the flag ISW is set according to the sign* of X_1 and X_2, the IWHCPART message, and the plane coded in IPLANE. Table 4 lists the choices. This flag controls the transfer or nontransfer of points from the set to the subset. If ISW is 1 the transfer is to take place. The processing is terminated if ISW cannot be set with JE returning a NO.

Table 4
Conditions for Setting the Flag ISW

Position of Major Axis	Clip Line Positive		Clip Line Negative	
	Top Plane	Bottom Plane	Top Plane	Bottom Plane
Negative end down (IWHCPART contains NEGDOWN)	1	0	0	1
Positive end down (IWHCPART contains POSDOWN)	0	1	1	0

NOTE: For the initial ISW setting, the condition of three variables must be considered. Is the clip line positive or negative with respect to x'? Is the major axis X' directed toward or away from the base plane? Which end plane is being considered, top or bottom?

If the clip line is positive, processing continues after the setting of ISW at statement 310; otherwise, with a negative clip line, the flow goes to statement 510.

At statement 310 the DO 400 loop is entered. In this loop the relative position of the first clip point (X_1, Y_1) within the (XS, YS) set is initially searched for since the clip line is positive or coincident with the Y' axis. (The clip line points lie on the ellipse but are

*The signs must be identical.

not explicitly in the XS, YS set. The first XS, YS point in the origin centered condition begins at (0, b) where b is the semimajor axis.) After each conditional comparison for the relative position determination, an XS, YS point is passed to the subset if ISW is 1; otherwise, no point is passed and the index is incremented.

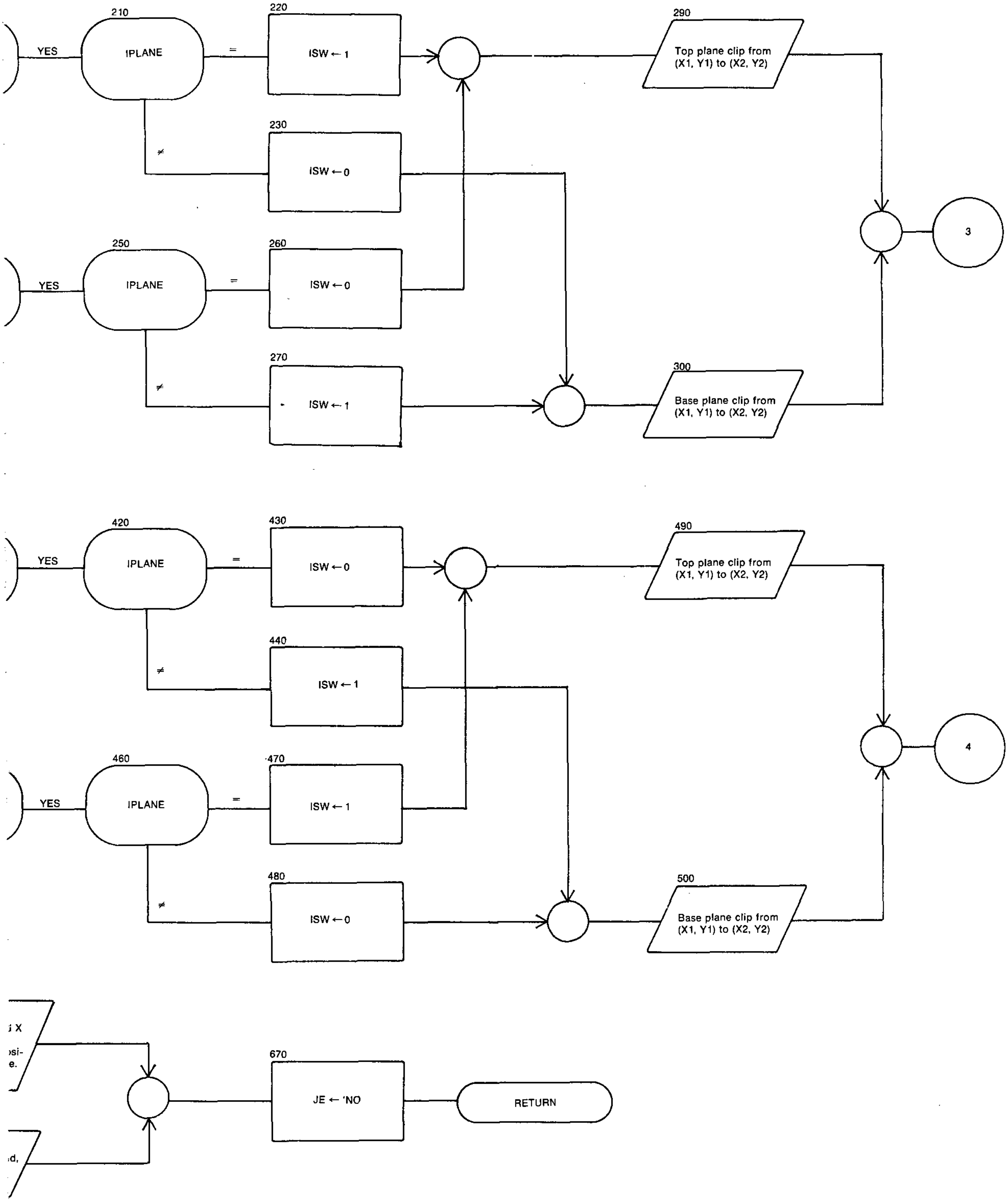
When the first point is found relative to the XS, YS set it is transferred to the subset. Then since a transition point has been reached, ISW is changed and X_1 is flagged with 99.99.

The search for the relative position of the second point (X_2, Y_2) commences. When it is found a test is necessary to determine if the process is complete or if the remainder of the XS, YS set must be passed to the subset.

At statement 510 a similar loop (DO 600) is entered, only this time the second point's relative position is searched for first because the clip line is negative.

Control, after both loops, reverts to statement 610. The first point from XS, YS is put on the end of the subset to complete the loop. Then the (XS, YS) arrays are zeroed, the (XSS, YSS) subset is passed to (XS, YS), the count ISET is reset by ISUBSET, and JE set to YES. Just prior to the return the original rotation and offset is put back into the discrete set if necessary. (If the rotation was zero, no translation was performed at the start of the subroutine.)

The flowchart is shown in Fig. 35.



35 - Subroutine CLIP (part 1 of 2 parts)

UNCLASSIFIED

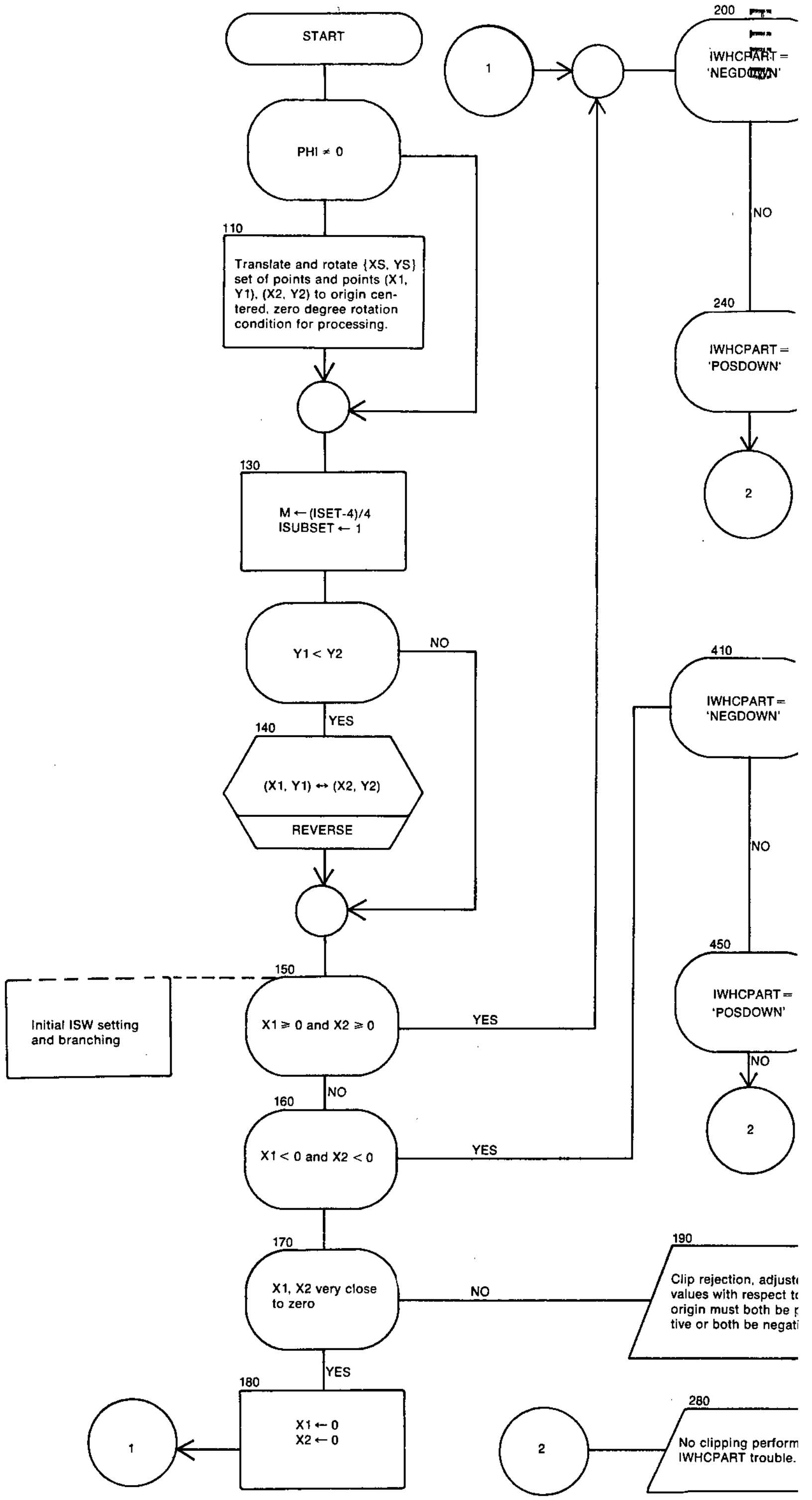
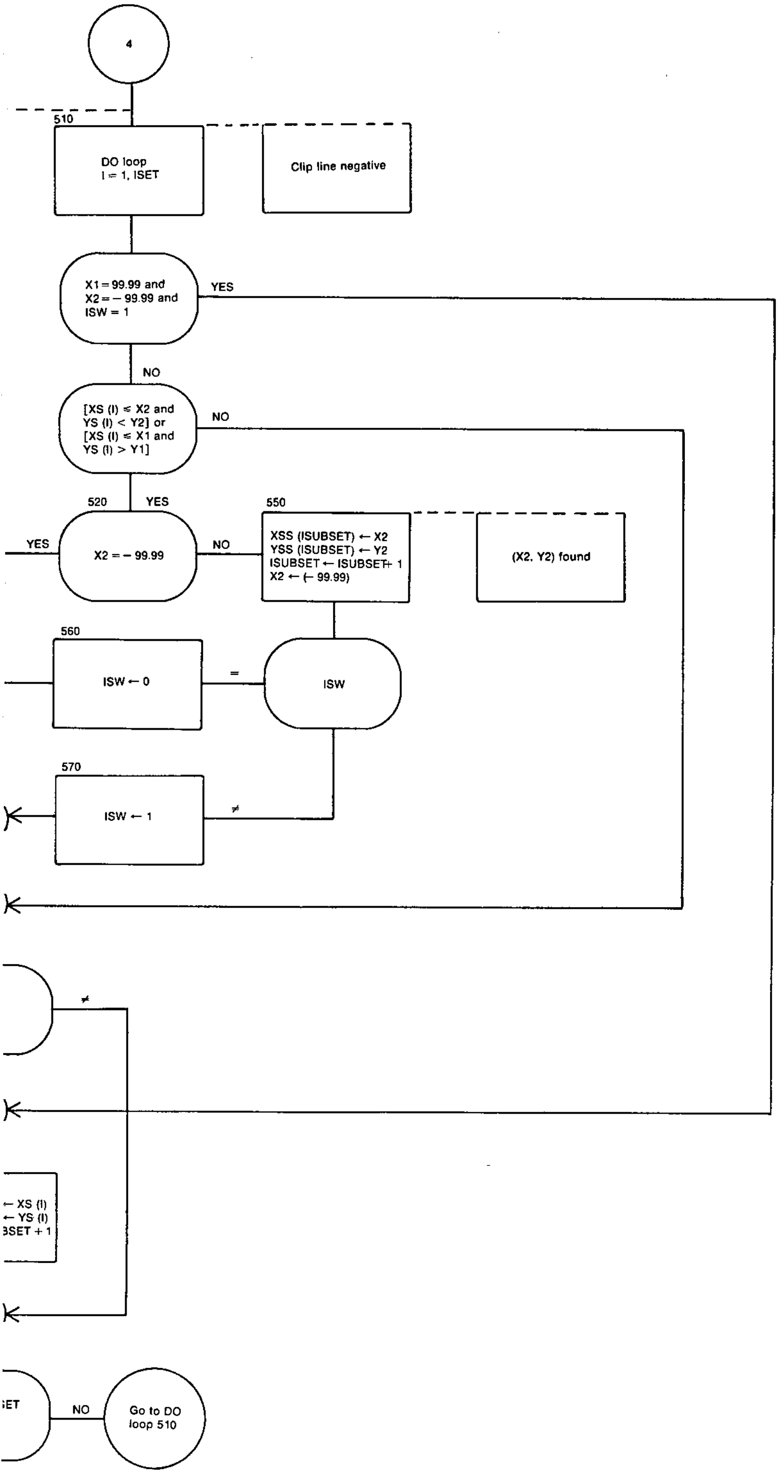
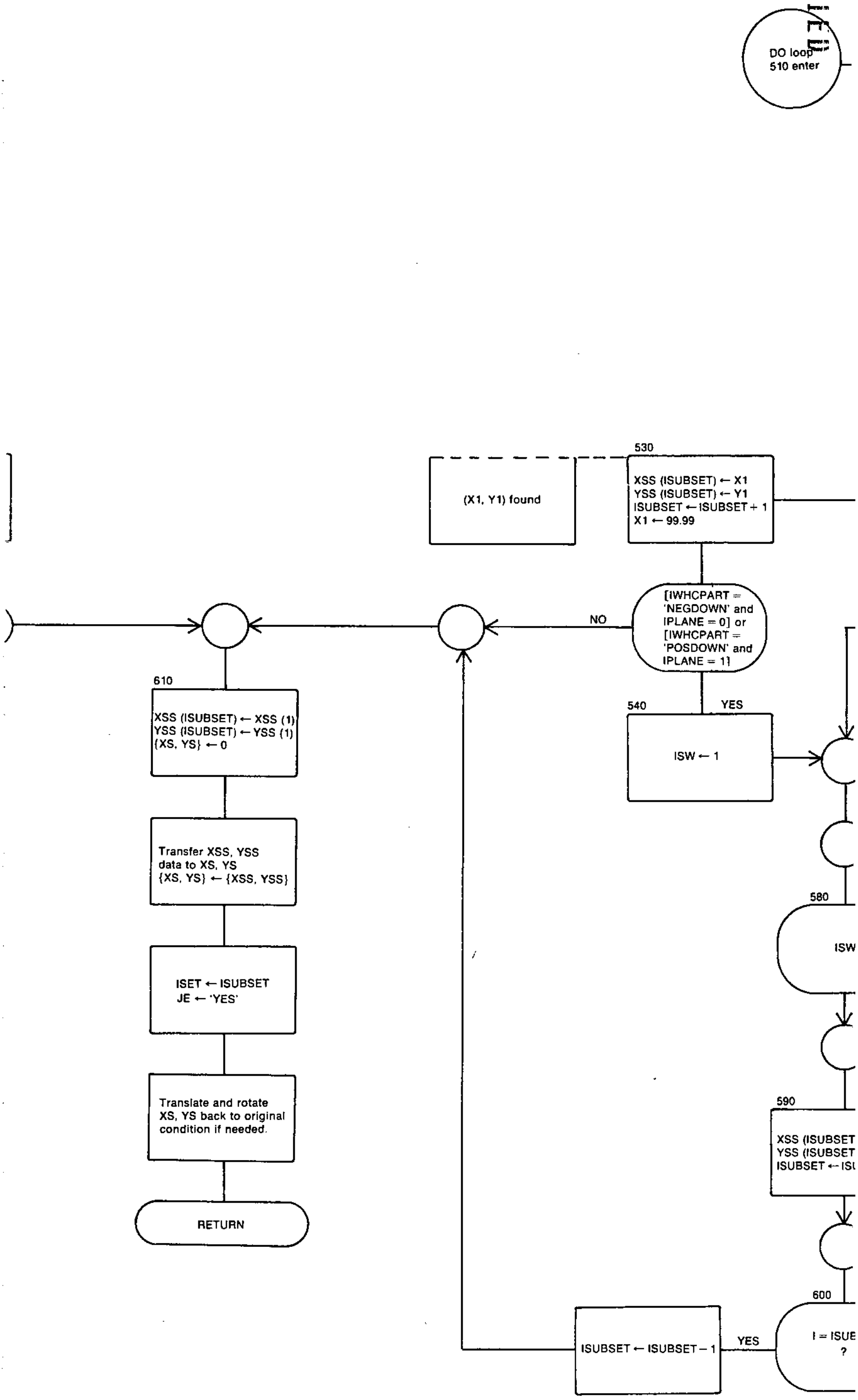


Fig.

UNCLASSIFIED





35 - Subroutine CLIP (part 2 of 2 parts)

UNCLASSIFIED

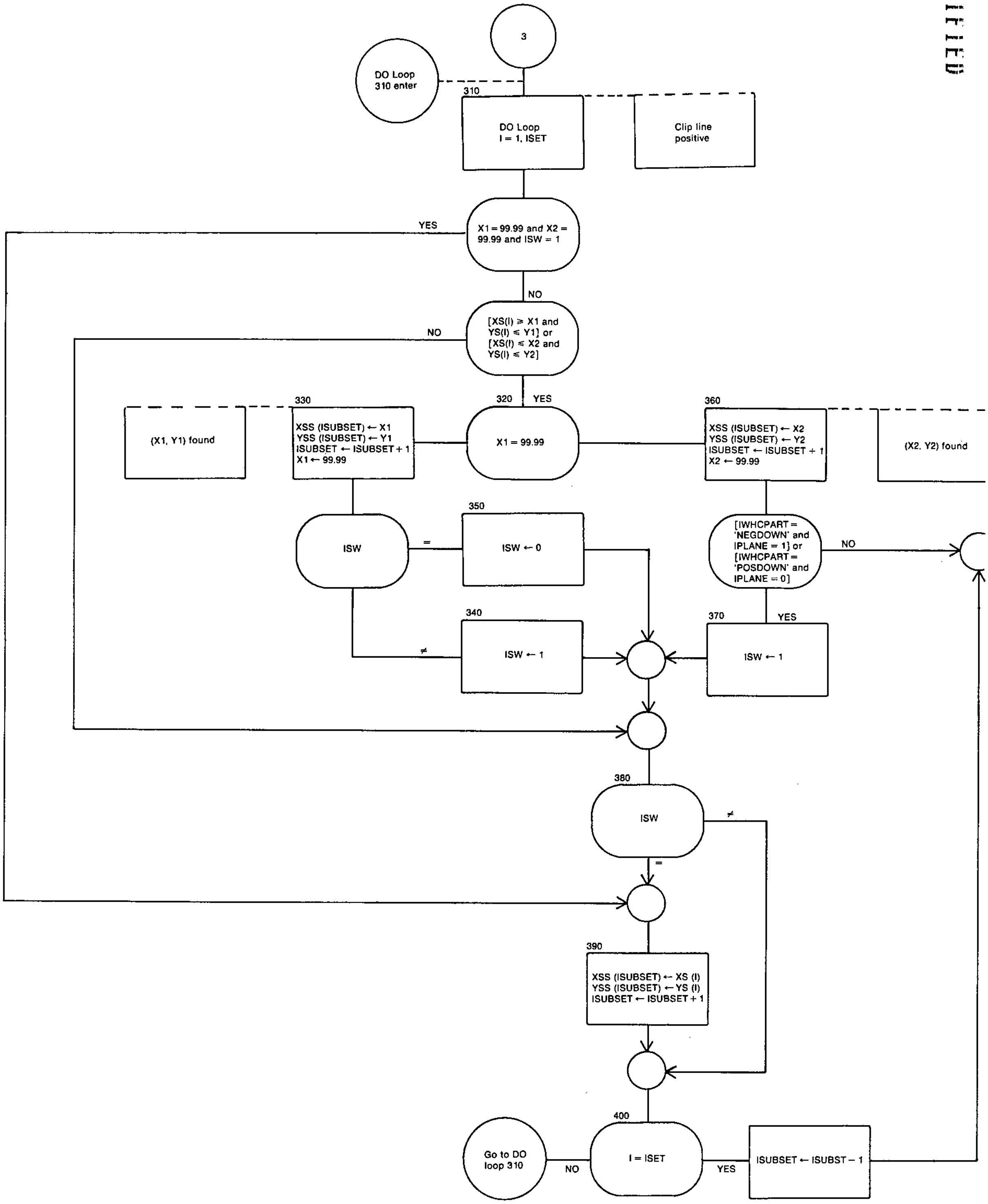


Fig.

Subroutine CONVERT (P, Q, RR, XX, YY, ZZ) – This subroutine performs one of two possible conversions between three coordinate systems, depending on how it is entered. If entered at its start it will convert the point (P, Q, RR) in the HOPE plane's coordinate system into the point (XX, YY, ZZ) in the XYZ system. But if entrance is made through the entry point CONVERT2, the point (XX, YY, ZZ) in the XYZ system is converted into the point (P, Q, RR) in the cylinder's coordinate system. This subroutine is called from Program DISCRETE.

Pertinent elements from CROSSEC common used in this subroutine are

A, B, C, D, the HOPE plane constants,

XMAT9, the HOPE plane's coordinate system's direction cosine matrix that was filled in TESTHOPE and FILXMAT9.

XC, YC, ZC, the cylinder's axis point, and

DCOSCYL, the cylinder system's direction cosine that was filled in TESTCYL and DFCOSCYL.

The conversions performed by this subroutine can be expressed in matrix form as follows:

Convert

$$\left[XX, YY, ZZ\right] \leftarrow \left[P, QQ, RR, 1\right] \begin{bmatrix} XMAT9(1) & XMAT9(2) & XMAT9(3) \\ XMAT9(4) & XMAT9(5) & XMAT9(6) \\ XMAT9(7) & XMAT9(8) & XMAT9(9) \\ D*A & D*B & D*C \end{bmatrix}$$

Convert2

$$\left[P, Q, RR\right] \leftarrow \left[(XX\text{-}XC), (YY\text{-}YC), (ZZ\text{-}ZC)\right] \begin{bmatrix} DCOSCYL(1) & DCOSCYL(4) & DCOSCYL(7) \\ DCOSCYL(2) & DCOSCYL(5) & DCOSCYL(8) \\ DCOSCYL(3) & DCOSCYL(6) & DCOSCYL(9) \end{bmatrix}$$

Subroutine CONVERT is charted in Fig. 36.

Subroutine REVERSE (M_1, N_1, M_2, N_2) – This subroutine is called from Subroutine CLIP and reverses the coordinates of two points. The two-dimensional point (M_1, N_1) swaps values with the two-dimensional point (M_2, N_2); i.e., the values originally held by (M_1, N_1) are given to (M_2, N_2) and the values originally held by (M_2, N_2) are given to (M_1, N_1).

The flowchart for REVERSE is given in Fig. 37.

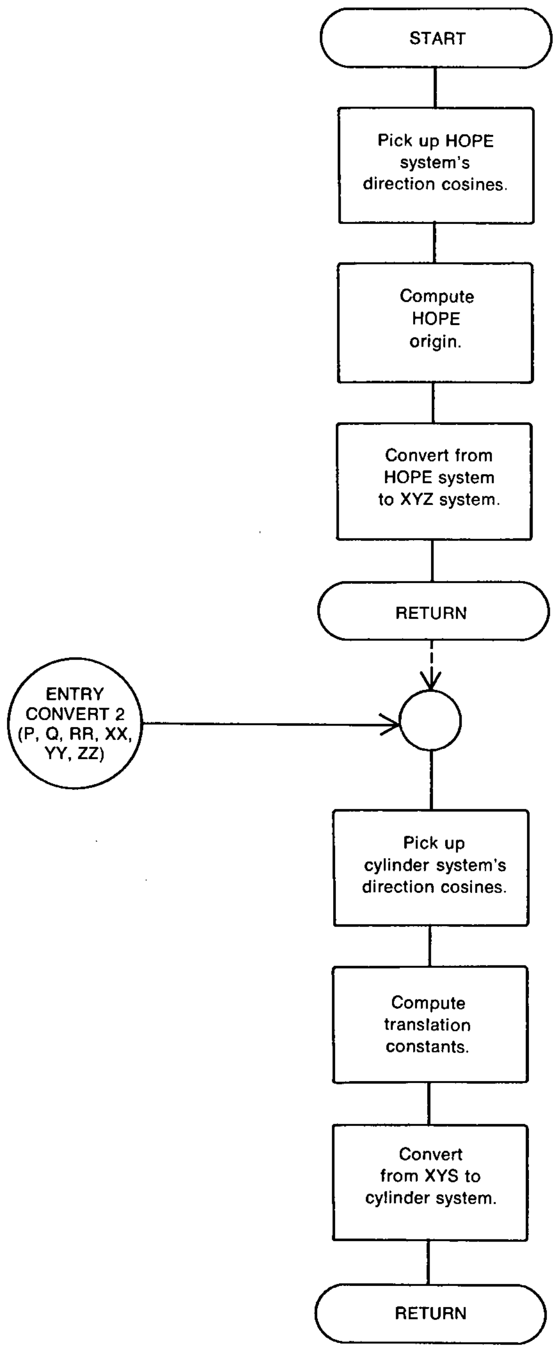


Fig. 36 - Subroutine CONVERT

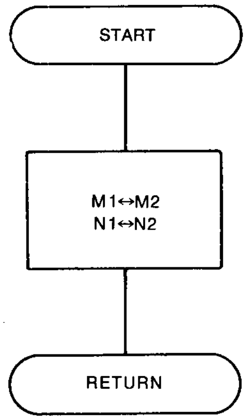


Fig. 37 - Subroutine REVERSE

ACKNOWLEDGMENT

The author wishes to express his appreciation:

To the Engineering Services Division for their support of this work which extends the usefulness of the APT system to NRL and the Navy;

To Mr. Lloyd Murphy, Mr. Nick Maddage, and Mr. Jay Williams, personnel of that division who work with APT every day and are to be thanked for their patience and encouragement over many long months of contact on this project;

To John Ihnat, who as immediate supervisor is due a vote of appreciation for his guidance and direction; and finally, and perhaps most important of all,

To the Research Computation Center, whose operating staff conduct the complicated business of batching the jobs through the CDC 3800, and whose long-suffering consulting staff not only interpret the unintelligible, but maintain an unfailing optimism about tomorrow's output.

ANNOTATED REFERENCES

1. Thompson, K.P., "Cross-Sectional Plots of Plane Intersections: An Adaptation of the APT System," NRL Report 7025, Jan. 27, 1970
   CROSEC (MOD 1.0) that produces plane intersections in the HOPE plane is described in formulas and in Fortran language.

2. Thompson, K.P., "CROSEC Computational Manual I (NRL APT System) Equations for Intersections Between Planes and Cylinders (Infinite and Bounded), Planes and Spheres," NRL Report 7202, Jan. 26, 1971
   The formula derivation is continued to include cylinders and spheres. Provides the theoretical foundation for CROSEC (MOD 2.0).

3. Thompson, K.P., "Cross-sectional Graphics with Section I Canonical Forms," Proceedings, Fall 1970 APT Technical Meeting, IIT Research Institute, 10 West 35th St., Chicago, Illinois, 60616
   CROSEC (MOD 2.0) is reported on to the APT community.

4. Roberts, L.G., "Machine Perception of Three-Dimensional Solids," Technical Report No. 315, Lincoln Laboratory, MIT, May 22, 1963
Quoting from the report, "A new mathematical method was conceived which utilizes volume inequality matrices to find out whether a point is inside or outside a volume. This test can then be extended by linear inequality solutions to tell which segment of a line is behind a volume."

5. Appel, A., "The Notion of Quantitative Invisibility and the Machine Rendering of Solids," Proceedings of ACM National Conference, 1967, Washington, D.C.:Thompson Book Co., 1967
Develops a concept of "implied vorticity" and a "tri-sense test" to determine the visibility or invisibility of a line. The test involves detecting a change of sign in the computation of the area of a triangle.

6. Appel, A., "Some Techniques for Shading Machine Renderings of Solids," Proceedings of the Spring Joint Computer Conference, Vol. 32, pp. 37-45, 1968, Washington, D.C.: Thompson Book Co.
Among other topics the author speaks of "the method of cutting planes" which involves the "intersections of a plane that passes through the observation point and assemblies of planes which can enclose one or more polyhedra."

7. Galimberti, R., and Montanari, U., "An Algorithm for Hidden Line Elimination," CACM, Vol. 12, April 1969, pp. 206-211
Constructs a normal for every face from the points defining the face. Two adjacent face normals are used to determine concaveness or convexness, which in turn is used in testing a line for visibility.

8. Bouknight, W.J., "An Improved Procedure for Generation of Half-Tone Computer Graphics Presentations," Report R-432, Coordinated Science Laboratory, University of Illinois, Sept. 1, 1969, also reported as, "A Procedure for Generation of Three-Dimensional Half-Toned Computer Graphics Presentations," CACM, Sept. 1970, pp. 527-536
Develops a "LINESCAN Algorithm" which involves the detection of both the projection of the end points of a plane-line segment and the projection of the intersection of two or more planes; the latter is called the "implicitly defined line."

9. Loutrel, P.P., "A Solution to the Hidden-Line Problem for Computer-Drawn Polyhedra," IEEE Trans. Comput. C-19 (No. 3):205-213, Mar. 1970
A polyhedron is a finite set of polygons arranged in space in such a way that every side of each polygon belongs to just one other polygon with the restriction that no subset has the same property. Intersecting polyhedra are not allowed. Utilizes normals to faces and traces out a path of potentially visible edges.

10. Lerman, H.N., "A Planar Solution to 3-Dimensional Plotting," Software Age, July 1970, pp. 16-19
The method described in this paper solves the polyhedra problem without requiring that the curvature of its edges be known and without breaking the edges to solve the window problem. The normal to the plane is determined by the cross product of two of the edges in the plane.

Appendix A

LISTING OF CROSEC (MOD 2.0)

This appendix contains the complete Fortran IV listing of the programming elements in CROSEC (MOD 2.0) in the order in which they are discussed and flowcharted in the text. The list below gives the beginning and ending sequence numbers for PRCNTL, LBSRCH, and FINI and the three-letter prefixes used for the remaining programs, functions, and subprograms. Use these keys in a page-thumbing search for a particular listing.

PRCNTL	00230000 through 00231260
LBSRCH	00240000 through 00240880
FINI	15450000 through 15451000
	prefix
ISHOPE	ISH
MM	MM
CROSEC	CRS
TESTHOPE	TEH
FILXMAT9	FXM
GENSOL	GEN
ISITOK	IOK
DLINE	DLN
CYLINDER	CYL
TESTCYL	TCY
S32TO48	STR
FDCOSCYL	FCY
ALGORA	ALA
ALGORB	ALB
ALGORC	ALC
ALGORD	ALD
HOPARCYL	HOP
DRAW	DRW
DISCRETE	DIS
POINTSE	POE
POINTSC	POC
EXPAND	EXP
PLOTSET	PLS
ELBOUNDS	ELB
CLIP	CLP
CONVERT	CON
REVERSE	REV

```
      SUBROUTINE  PRCNTL
      COMMON /TIMETEST/        KSETIME1, KSETIME2, KSETIME,  KSETADR
      COMMON /TAPETABL/ NTAPEX(10)
C
C...     COMMON AREA FOR CDC APT 3 SECTION 0
C
C******           ******          ******          ******          ******
      COMMON /SYSTEMZ/          SYSTEM(4),KAPTCN,   KAPTTR,    KAPTIO,
     A     KAPTIO,   KFLAGS(10),KO,       K1,        K2,        K3,
     B     K4,       K5,        K6,       K7,        K8,        K9,
     C     IFILL1,   IFILL2,    IFILL3,   IFILL4,
     D     KFLAG0,   KFLAG1,    KFLAG2,   KFLAG3,    KFLAG4,
     E     IFILL5,   IWAVEN,    IPTNLY,   NOPOST,    IFILL6,    KAUTOP,
     F     ICLPRT,   INDEXX,    IPLOTR,   IFILL7,    NOPLOT,    KDYNFG,
     G     LOCJPT,   LOCBEG,    KSECIN,   NCLREC,    LOCMAC,    KPOCKET,
     H     IFILL8,   IFILL9,    IFILL10,  IPOSTP(1),NUMPST,     IPOSTFL(18),
     I     TAPETB(1),CANTAP,    CLTAPE,   POCTAP,    PLOTAP,    SRFTAP,
     J     LIBTAP,   CRDTAP,    IFILL11,  CORTAP,    TAPES1,    NUCLTP,
     K     CLCNVTAP, TAPES4,    FORTIN,   INTAPE,    IOUTAP,    PUNTAP,
     L     LSTFLG,   NUNITS,    KONVTCL,  KINTRUPT,
     M     PI,       PI02,      DGTRD,    RDTDG,     ONE,
     N     EXTRAD(20)
      EQUIVALENCE   (PROTAP,  TAPETB)
C******           ******          ******          ******          ******
C
       EQUIVALENCE(INCHOPE,IFILL8)
      EQUIVALENCE (KOM,SYSTEM)
      TYPE INTEGER POCTAP
      DIMENSION KOM(1)
          KSECIN = 0
C...      FREE THE TIME INTERRUPT
          CALL TIMESET (0)
      IRET = 0
      IF(IWAVEN.EQ.0)9001,9000
 9000 PRINT 9996
 9001 I=IWAVEN+1
      GOTO(9010,9005,9100),I
C
C..   ERROR PROCEDURE.GO TO NEXT PART PROGRAM
C
 9005 GOTO 9040
C
C..   HAVE WE JUST RETURNED FROM SECTION 1
C
 9010 IF(KFLAG1.EQ.0)9300,9020
C
C..   NO
C
 9020 DO 9030 I=2,8,2
      IF(KFLAGS(I).EQ.0)9030,9200
 9030 CONTINUE
C     END OF PART PROGRAM.PRINT ELAPSED TIME AND GO TO INITIAL
C
C
C
 9040 KOM(2)=TIMEF(1)
```

```
      KOM(2)=(KOM(2)-KFLAG4)/1000                                       00230550
      MIN=KOM(2)/60                                                     00230560
      ISEC=KOM(2)-60*MIN                                                00230570
      WRITE (IOUTAP,9990) MIN,ISEC                                      00230580
C...       CHECK FOR MORE PART PROGRAMS                                 00230590
 9044 IF (LSTFLG ,EQ. 1) 9045,9050                                      00230600
C...       HAS CLTAPE CONVERSION TAKEN PLACE                            00230608
 9045 IF (CLCNVTAP) 9046,9049                                           00230610
C...       YES - REWIND CLCNVTAP                                        00230612
 9046 CALL TAPEOP (CLCNVTAP,2,IRET)                                     00230614
      CALL TAPEOP (CLCNVTAP,1,IRET)                                     00230616
 9049 CONTINUE
       CALL PLOTS(0,0)
       CALL PLOT(0,0,3)
       CALL STOP PLOT
       CALL EXIT
C...        INITIALIZE FOR NEXT PART PROGRAM                            00230620
 9050 CONTINUE                                                          00230630
      CALL TAPEOP (POCTAP,1,IRET)                                       00230650
      CALL TAPEOP (SRFTAP,1,IRET)                                       00230660
      CALL INITIAL                                                      00230670
      RETURN                                                            00230680
 9100 ICLPRT = 1                                                        00230690
      IWAVEN=1                                                          00230700
      I=6                                                               00230710
 9200 KFLAGS(I)=0                                                       00230720
 9250 I=I-1                                                             00230730
      KOM(2)=TIMEF(1)                                                   00230740
      KOM(2)=(KOM(2)-KFLAG4)/1000                                       00230750
      MIN=KOM(2)/60                                                     00230760
      ISEC=KOM(2)-60*MIN                                                00230770
      WRITE (IOUTAP,9991) KFLAGS(I),MIN,ISEC                            00230780
      IF (I .EQ. 3) 9255,9260                                           00230790
C...       BEGIN SECTION 2, INITIALIZE POCTAP IF NECESSARY              00230800
 9255 IF (KPOCKET .EQ. 1) 9256,9260                                     00230810
 9256 POCTAP = NTAPEX(4)                                                00230820
C...       REWIND POCTAP                                                00230830
      CALL TAPEOP (POCTAP,1,IRET)                                       00230840
 9260 CALL LBSRCH (KFLAGS(I),LI)                                        00230850
 9300 KFLAG1=1                                                          00230860
C     RESET KFLAG1 TO ZERO TO ALLOW FOR CROSEC
       KFLAG1=0
C     IF IFILL7 IS NON-ZERO
C     A CROSEC CALL IS WANTED OR IS IN PROGRESS
       IF(IFILL7.EQ.0)9031,9051
C     IF IFILL7 IS 101
C     THE FIRST SEGMENT IS TO BE CALLED FOR PROCESSING
C     OF THE LINE AND PLANE INTERSECTIONS
 9051  IF(IFILL7.EQ.101)9052,9053
 9052  CALL LBSRCH(101,LI)
C     IF IFILL7 IS 102
C     THE SECOND SEGMENT IS TO BE CALLED FOR INITIAL OR
C     CONTINUED PROCESSING OF THE SEARCH FOR CYLINDERS AND SPHERES
 9053  IF(IFILL7.EQ.102)9054,9055
 9054  CALL LBSRCH(102,LI)
C     IF IFILL7 IS 103
```

```
C      THE THIRD SEGMENT IS TO BE CALLED FOR
C      DISCRETE POINT PROCESSING OF A CIRCLE
C      OR ELLIPSE OF INTERSECTION
 9055   IF(IFILL7.EQ.103)9056,9057
 9056   CALL LBSRCH(103,LI)
C      IF IFILL7 IS 9999
C      ALL CROSS SECTION PROCESSING IS COMPLETED
 9057   IF(IFILL7.EQ.9999)9031,9058
 9059   FORMAT (1X,*IFILL7 FLAG TROUBLE IN PRCNTL. WAS EQUAL TO*,I8)
 9058   PRINT 9059,IFILL7
 9031   CONTINUE
C      RESET FLAGS FOR NORMAL APT PROCEDURE
        IFILL7=0
        IFILL8=0
        IFILL9=0
        KFLAG1=1
C
C..    CHECK THE CLPROCESSOR FLAGS
C      CHECK NOPLOT FLAG- IF ON VOID PLOTTING
C
       IF(NOPLOT.EQ.0)9303,9302
 9302 IPLOTR=0
 9303 DO 9310 I=36,38,1
       IF(KFLAGS(I).EQ.0)9310,9305
 9305 KFLAGS(6)=1
       GOTO 9320
 9310 CONTINUE
C...        EXECUTE SECTION 3 IF CONVERT FLAG IS SET
       IF (KONVTCL.EQ. 1) KFLAGS(6) = 1
 9320 IF(NOPOST.EQ.0)9326,9325
 9325 KFLAGS(8)=0
 9326 IF (KAUTOP .EQ. 0) 9330,9327
 9327 CALL BUFFTP (PROTAP,3)
 9328 KFLAGS(4)=0
       GOTO 9020
C      IPTNLY=0. NORMAL FLOW,=1,SKIP SECTION 2 AND CHANGE TAPES
C      IPTNLY=2- SKIP SECTION 2 AND DO NOT INTERCHANGE TAPES
C      IPTNLY=3- DO NOT SKIP SECTION 2, POSITION CLTAPE AT
C      RECORD NUMBER NCLREC
C
 9330 I=IPTNLY+1
        GO TO (9360,9340,9327,9380),I
 9340 CALL TAPECH(PROTAP,CLTAPE)
       CALL TAPECH (PROTAP,NUCLTP)
       GOTO 9328
C..    GO EXECUTE SECTION 2
 9360 I=4
       GOTO 9200
 9380 CALL BUFFTP (CLTAPE,2)
           CALL SEARCH(CLTAPE,NCLREC,KOM)
       NCLREC=NCLREC-1
       IF(KOM.GE.0)9040,9360
 9990 FORMAT(44H1PART PROGRAM EXECUTION TIME =            ,I5,7H  MIN
    1 ,I2,4H SEC,/1H1)
 9991 FORMAT(28H0START EXECUTION OF SECTION ,I5,20X,
    115H ELAPSED TIME = ,I5,7H  MIN  ,I2,4H SEC/1H0)
```

11/02/70

```
9996 FORMAT(7H FAILED)                                             00231250
     END                                                           00231260
```

PRCNTL 11/02/70 ED 0

```
                                    IDENT     PRCNTL
PROGRAM LENGTH                00476
ENTRY POINTS     PRCNTL       00071
BLOCK NAMES
                 TIMETEST     00004
                 TAPETABL     00012
                 SYSTEMZ      00175
EXTERNAL SYMBOLS
                 THEND,
                 Q1Q10100
                 Q8QDICT,
                 TIMESET
                 TIMEF
                 TAPEOP
                 PLOTS
                 PLOT
                 STOPPLOT
                 EXIT
                 INITIAL
                 LBSRCH
                 BUFFTP
                 TAPECH
                 SEARCH
                 STH,
                 QNSINGL,
   00254 SYMBOLS
```

11/02/70

```
      SUBROUTINE LBSRCH (N,LI)                                          00240000
C,,,  N IS THE NAME OR IDENTIFICATION NUMBER OF THE SECTION DESIRED     00240010
C,,,  LI IS THE RETURN VARIABLE                                         00240020
C                                                                       00240030
C,,,     COMMON AREA FOR CDC APT 3 SECTION 0                            00240040
C                                                                       00240050
C******         ******          ******          ******          ****** 00240060
      COMMON /SYSTEMZ/          SYSTEM(4),KAPTCN,    KAPTTR,   KAPTIO,  00240070
     A     KAPTID,   KFLAGS(10),KO,        K1,        K2,       K3,     00240080
     B     K4,       K5,        K6,        K7,        K8,       K9,     00240090
     C     IFILL1,   IFILL2,    IFILL3,    IFILL4,                      00240100
     D     KFLAG0,   KFLAG1,    KFLAG2,    KFLAG3,    KFLAG4,           00240110
     E     IFILL5,   IWAVEN,    IPTNLY,    NOPOST,    IFILL6,   KAUTOP, 00240120
     F     ICLPRT,   INDEXX,    IPLOTR,    IFILL7,    NOPLOT,   KDYNFG, 00240130
     G     LOCJPT,   LOCBEG,    KSECIN,    NCLREC,    LOCMAC,   KPOCKET,00240140
     H     IFILL8,   IFILL9,    IFILL10,   IPOSTP(1),NUMPST,    IPOSTFL(18),00240150
     I     TAPETB(1),CANTAP,    CLTAPE,    POCTAP,    PLOTAP,   SRFTAP, 00240160
     J     LIBTAP,   CRDTAP,    IFILL11,   CORTAP,    TAPES1,   NUCLTP, 00240170
     K     CLCNVTAP, TAPES4,    FORTIN,    INTAPE,    IOUTAP,   PUNTAP, 00240180
     L     LSTFLG,   NUNITS,    KONVTCL,   KINTRUPT,                    00240190
     M     PI,       PI02,      DGTRD,     RDTDG,     ONE,              00240200
     N     EXTRA0(20)                                                   00240210
      EQUIVALENCE   (PROTAP,  TAPETB)                                   00240220
C******         ******          ******          ******          ****** 00240230
C                                                                       00240240
      DIMENSION NONTP(25),ISPECL(14)                                    00240250
C,,,      LIST THE AVAILABLE OVERLAY AND SEGMENT NUMBERS ON LIBTAP      00240260
       DATA (NONTP=100,101,102,103,200,300,400,401,402,403,404,         00240271
     1   14(7777))                                                      00240273
C,,,      LIST THE SPECIAL PROGRAM AND ITS SECTION OF EXECUTION         00240280
      DATA (ISPECL = 6HDYNDMP,200,6HCLCHCK,200,6HCOMDMP,200,8(0))       00240290
      DATA (NS = 0)                                                     00240294
C,,,      SAVE LAST SEGMENT NUMBERS                                     00240295
      NSSV = NS                                                         00240296
C,,,      IS N AN INTEGER OR A BCD WORD                                 00240300
      NN = N                                                            00240310
      IF (NN / 10000B .EQ. 0) 10,100                                    00240320
C,,,      INTEGER FOUND OBTAIN OVERLAY AND SEGMENT NUMBERS              00240330
   10 NO = NN / 100                                                     00240340
      NS = NN - (NO * 100)                                              00240350
C,,,      SEARCH FOR NN IN OVERLAY LIST                                 00240360
   15 DO 20 I = 1,25                                                    00240370
      IF (NN .EQ. NONTP(I)) 30,20                                       00240380
   20 CONTINUE                                                          00240390
C,,,      OVERLAY NOT FOUND PRINT ERROR MESSAGE                         00240400
      WRITE (IOUTAP,25) NO,NS,N                                         00240410
   25 FORMAT ( 10H   OVERLAY I4, 10H   SEGMENT I4, 10H   NAMED   ,A8,   00240420
     1    19H   IS NOT ON LIBTAP /)                                     00240430
   28 CALL PRCNTL                                                       00240440
   30 CONTINUE                                                          00240450
      KSECIN = NO                                                       00240460
      IF (NS .EQ. 0)32,34                                               00240465
   32 CALL OVERLAY (NO,NS,LIBTAP,0)                                     00240470
      GO TO 35                                                          00240472
   34 KRECALL = 0                                                       00240474
C,,,      IF SEGMENT IS IN CORE, RETURN TO IT                           00240475
```

```
      IF (NS .EQ. NSSV) KRECALL = 8H,RECALL,
      CALL SEGMENT (NO,NS,LIBTAP,KRECALL)
   35 LI = -999
      RETURN
C...      BCD NAME FOUND.  SEARCH FOR NAME IN ISPECL
  100 DO 110 I = 1,14,2
      IF (NN .EQ. ISPECL(I)) 120,110
  110 CONTINUE
      IF (NO .EQ. 1) 115,111
C...      NOT IN SECTION 1, BCD NAME NOT LOCATED. - ERROR
  111 WRITE (IOUTAP,112) NN
  112 FORMAT (14H ROUTINE NAMED  A8, 28H NOT LOCATED IN LBSRCH TABLE/)
      GO TO 28
C...      N IS ASSUMED TO BE A SYSTEMS MACRO
  115 LI = 0
      RETURN
C...      NAME FOUND, IS ROUTINE TO BE EXECUTED IN THIS OVERLAY
  120 NNN = ISPECL(I+1)
      NNO = NNN / 100
      NNS = NNN - (NNO * 100)
      IF (NNO .EQ. NO) 130,200
C...             YES, CHECK CURRENT SEGMENT
  130 IF (NNS .EQ. NS) 150,135
  135 IF (NNS .EQ. 0) 150,140
C...      EXECUTE THE CALL TO NEW SEGMENT
  140 NN = NNN
      NS = NNS
      GO TO 15
C...      RETURN TO CALLING ROUTINE FOR EXECUTION
  150 LI = -I
      RETURN
C...             NO, CHECK FOR ANOTHER MATCH IF CURRENT OVERLAY IS NOT 1
  200 IF (NO .EQ. 1) 250,210
  210 I2 = I + 2
      DO 220 J = I2,14,2
      IF (NN .EQ. ISPECL(J)) 215,220
  215 I = J
      GO TO 120
  220 CONTINUE
C...      RETURN TO CALLING ROUTINE TO BUILD SPECIAL PROGRAM RECORDS
  250 LI = I
      RETURN
      END
```

LBSRCH   11/02/70   ED   0

```
                                IDENT     LBSRCH
PROGRAM LENGTH              00370
ENTRY POINTS     LBSRCH     00113
BLOCK NAMES
                 SYSTEMZ    00175
EXTERNAL SYMBOLS
                 THEND,
                 Q8QDICT,
                 PRCNTL
                 OVERLAY
                 SEGMENT
                 STH,
                 QNSINGL,
   00231 SYMBOLS
```

11/02/70

```
      SUBRØUTINE FINI
      BANK,(FINI  ),   /SYSTEMZ/,/SECT1LØG/,/VØCABTBL/
C,,,              F I N I  R Ø U T I N E
C,,,      THIS RØUTINE PRØDUCES THE FINAL PRØTAPE RECØRD FØR A
C,,,      PART PRØGRAM
C
C,,,     CØMMØN AREA FØR CDC APT 3 SECTIØN 0
C
C******          ******           ******           ******           ******
      CØMMØN /SYSTEMZ/          SYSTEM(4),KAPTCN,   KAPTTR,   KAPTIØ,
     A     KAPTID,   KFLAGS(10),KØ,     K1,       K2,       K3,
     B     K4,       K5,       K6,      K7,       K8,       K9,
     C     IFILL1,   IFILL2,   IFILL3,  IFILL4,
     D     KFLAG0,   KFLAG1,   KFLAG2,  KFLAG3,   KFLAG4,
     E     IFILL5,   IWAVEN,   IPTNLY,  NØPØST,   IFILL6,   KAUTØP,
     F     ICLPRT,   INDEXX,   IPLØTR,  IFILL7,   NØPLØT,   KDYNFG,
     G     LØCJPT,   LØCBEG,   KSECIN,  NCLREC,   LØCMAC,   KPØCKET,
     H     IFILL8,   IFILL9,   IFILL10, IPØSTP(1),NUMPST,   IPØSTFL(18),
     I     TAPETB(1),CANTAP,   CLTAPE,  PØCTAP,   PLØTAP,   SRFTAP,
     J     LIBTAP,   CRDTAP,   IFILL11, CØRTAP,   TAPES1,   NUCLTP,
     K     CLCNVTAP, TAPES4,   FØRTIN,  INTAPE,   IØUTAP,   PUNTAP,
     L     LSTFLG,   NUNITS,   KØNVTCL, KINTRUPT,
     M     PI,       PIØ2,     DGTRD,   RDTDG,    ØNE,
     N     EXTRA0(20)
      EQUIVALENCE   (PRØTAP,  TAPETB)
C******          ******           ******           ******           ******
C
C,,,     CØMMØN AREA FØR CDC APT 3 SECTIØN 1
C
C******          ******           ******           ******           ******
      CØMMØN /SECT1LØG/         ITAB1,    ITAB2,    ITAB3,
     A     ITAB4,    ITAB5,    ISNAM,    ITAB11,   ITAB12,   ITAB13,
     B     JENDPTPP, JENDCAN,  JENDSTØR, JSTRTCAN, JENDSYM,
     C     JCANTEMP, JRPTAB,   JLPTAB,   MAXNST,
     D     JINWD,    JCHAR,    IWDERR,   JBUFL,    NUPERP,   NUPUN,
     E     JSTYPE,   JVARS2,   SCHERR,   NMACV,    MACASN(25),
     F     INDXPT,   IPTP,     IXPT,     MØDE,     EØCFLG,   LPNDFL,
     G     TRMFLG,   INTRUPT,  JUMPFL,   ICDERR,   DEBUG,
     H     MACMØDE,  NESTFL,   NRESULT,  IPTLIM,   JEXEC,
     I     KTYPE,    MACTYP,
     J     IPARTERR, FINIS,    IØFLG,    MACDEL,   JSUBER,   NUMBERR,
     K     DEFSTØ(85),    DEFTAB(1000),  ZSUR(30),
     L     XMAT4(16),XMAT3(16),XMAT2(16),XMAT1(16),TMATX(16),
     M     ISTDMØDE, ISTDLIT,  ISTDTBL,  ISTDINDX, ISTDTYPE, ISTDWD,
     N     JPTIND,   KPTCØDE,  KPTNAME,  KPTTYPE,  KPTNUM,   KPTINDX,
     Ø     KPTSUB,   KØMFLG,   KØMPØP,   NØSUBS,   KANFLG,   KRFSYS,
     P     KANREC,   KANCNT,   INAME,    KANSURF,  KANINDX,
     Q     JPRELEN,  NEWCARD,  JGØRIT,   NUMSTID,  NUMCSEQ,
     R     IRECIX,   IRECNØ,   JTLPØS,   ITITLE(9),LSRECN,
     S     NNØDEFX,  NNØDEFI,  NIDUM,    ISLASH,   IEQUAL,   IBLANK,
     T     IDUMMY,   N10000,   N7777,    MASKU,    MASKL,    IDIV,
     U     MACREL,   MACLØC,   MACBEGN,  MACLAST,  MACLEVEL,
     V     MACNAME(3), MACINDX(3), NMV,  JRESTØR,  MACPSH(3,25),
     W     JTEMP1,   JTEMP2,   JTEMP3,   JTEMP4,   JTEMP5,   JTEMP6,
     X     JTEMP7,   JTEMP8,   JTEMP9,   EXTRA1(20)
      EQUIVALENCE (DEFANS(1),IDEFSTØ(4),DEFSTØ(4)),(LSTYPE, KTYPE),
```

```
     1    (PTNUM, KPTNUM)
      DIMENSION IDEFSTO(85),  DEFANS(26),  IDEFTAB(1000)  ,
     1    ILPTAB(200), IRPTAB(200), ITNTAB(200), JPROTP(100),
     2    IRECSV(200), MACVAR(25),  MACNOR(25),  INWORD(10), IBUUP(2),
     3    PISTO(6), IPISTO(6), IDREC(4)
      EQUIVALENCE (INWORD(14), IBUUP(16), ILPTAB(990), IRPTAB(790),
     1    ITNTAB(590), PISTO(390), IPISTO(390), IDREC(384),
     2    JPROTP(380), IRECSV(280),  MACVAR(75), MACNOR(50),
     3    IDEFTAB(1000), DEFTAB(1000))
C
      COMMON /VOCABTBL/ KOM(100)
C
      COMMON /2/    JTABNUM,JTABL(100)
C******          ******           ******            ******          ******
       COMMON/CROSSEC/
     1  A,B,C,D,IDSEND,XMAT9(16),
     2   ARRAY (254),
     3   XC,YC,ZC,UX,UY,UZ,R,
     4 SCALE,
     5  DCOSCYL(9),
     6  HO(9),
     7      XS(400),YS(400),
     8  X(103),Y(103),
     9  MAJOR,MINOR,PHI,CX,CY,RADIUS,ISET,
     D  CXYZ(16),EXYZ(16),EH(16),
     E  IOKTEST,
     F  TL,
     G      ILOOP,IP,KR(200),IONCE
      LSTFLG = 0
C...      IS AN IMPLICIT CHECK SURFACE PENDING
      IF (IRECSV) 11,15
   11 CALL ERRMSG (650)
      IRECSV(1) = 0
   15 JGORIT = 0
      JPROTP(2) = 14000
      JPRELEN = 2
      CALL RITAPE
C...      MOVE SRFTAP TO END
      IF (LSRECN .GT. 0) 16,18
   16 LSRECN = LSRECN + 1
      CALL SEARCH (SRFTAP,LSRECN,IOFLG)
   18 CALL TAPEOP (SRFTAP,2,IOFLG)
C...      WRITE END OF FILE ON PROTAP AND REWIND
      CALL TAPEOP (PROTAP, 2, IOFLG)
      CALL TAPEOP (PROTAP, 1, IOFLG)
      CALL BUFFTP (CANTAP,3)
C...      CHECK HERE FOR MORE PART PROGRAMS
      READ (INTAPE,50) SYSTEM(1)
      IF (EOF,INTAPE) 19,20
C...      SET LAST PART PROGRAM FLAG
   19 LSTFLG = 1
   20 BACKSPACE INTAPE
C...      IF TIME INTERRUPT OCCURRED WE WANT CLPRNT OF THE PROTAP
      IF (KINTRUPT .EQ. 1) 25,30
   25 KINTRUPT = IWAVEN = 0
   30  CALL ISHOPE
```

11/02/70

```
      CALL PRCNTL                                              15450972
   50 FORMAT (A8)                                              15450980
      RETURN                                                   15450990
      END                                                      15451000
```

```
   FINI                                          11/02/70        ED      0
                                IDENT     FINI
PROGRAM LENGTH              00122
ENTRY POINTS     FINI       00006
BLOCK NAMES
                 SYSTEMZ    00175
                 SECT1LOG   02642
                 VOCABTBL   00144
                 2          00145
                 CROSSEC    03037
EXTERNAL SYMBOLS
                 THEND,
                 Q8QDICT,
                 ERRMSG
                 RITAPE
                 SEARCH
                 TAPEOP
                 BUFFTP
                 ISHOPE
                 PRCNTL
                 Q8QIFEOF
                 BSP,
                 TSH,
                 QNSINGL,
   00445 SYMBOLS
```

11/02/70

```
      SUBROUTINE ISHOPE
      COMMON /TIMETEST/      KSETIME1, KSETIME2, KSETIMF,  KSETADR
C
C     ...     COMMON AREA FOR CDC APT 3 SECTION 0
C
C     *          ******          ******          ******          ******
      COMMON /SYSTEMZ/       SYSTEM(4),KAPTCN,   KAPTTR,   KAPTIO,
     1    KAPTID,   KFLAGS(10),KO,      K1,       K2,       K3,
     2    K4,       K5,       K6,       K7,       K8,       K9,
     3    IFILL1,   IFILL2,   IFILL3,   IFILL4,
     4    KFLAG0,   KFLAG1,   KFLAG2,   KFLAG3,   KFLAG4,
     5    IFILL5,   IWAVEN,   IPTNLY,   NOPOST,   IFILL6,   KAUTOP,
     6    ICLPRT,   INDEXX,   IPLOTR,   IFILL7,   NOPLOT,   KDYNFG,
     7    LOCJPT,   LOCBEG,   KSECIN,   NCLREC,   LOCMAC,   KPOCKET,
     8    IFILL8,   IFILL9,   IFILL10,  IPOSTP(1),NUMPST,   IPOSTFL(18),
     9    TAPETB(1),CANTAP,   CLTAPE,   POCTAP,   PLOTAP,   SRFTAP,
     $    LIBTAP,   CRDTAP,   IFILL11,  CORTAP,   TAPES1,   TAPES2,
     $    TAPES3,   TAPES4,   FORTIN,   INTAPE,   IOUTAP,   PUNTAP,
     $    LSTFLG,   LTVFLG,   KONVTCL,  KINTRUPT,
     $    PI,       PI02,     DGTRD,    RDTDG,    ONE,
     $    EXTRA0(20)
      EQUIVALENCE    (PROTAP,   TAPETB)
C     *          ******          ******          ******          ******
C
C     ...     COMMON AREA FOR CDC APT 3 SECTION 1
C
C     *          ******          ******          ******          ******
      COMMON /SECT1LOG/      ITAB1,    ITAB2,    ITAB3,
     1    ITAB4,    ITAB5,    ISNAM,    ITAB11,   ITAB12,   ITAB13,
     2    JENDPTPP, JENDCAN,  JENDSTOR, JSTRTCAN, JENDSYM,
     3    JCANTEMP, JRPTAB,   JLPTAB,   MAXNST,
     4    JINWD,    JCHAR,    IWDERR,   JBUFL,    NUPERP,   NUPUN,
     5    JSTYPE,   JVARS2,   SCHERR,   NMACV,    MACASN(25),
     6    INDXPT,   IPTP,     IXPT,     MODE,     EOFFLG,   LPNDFL,
     7    TRMFLG,   INTRUPT,  JUMPFL,   ICDERR,   DEBUG,
     8    MACMODE,  NESTFL,   NRESULT,  IPTLIM,   JEXEC,
     9    KTYPE,    MACTYP,
     $    IPARTERR, FINIS,    IOFLG,    MACDEL,   JSUBER,   NUMBERR,
     $    DEFST0(85),    DEFTAB(1000),  ZSUR(30),
     $    XMAT4(16),XMAT3(16),XMAT2(16),XMAT1(16),TMATX(16),
     $    ISTDMODE, ISTDLIT,  ISTDTBL,  ISTDINDX, ISTDTYPE, ISTDWD,
     $    JPTIND,   KPTCODE,  KPTNAME,  KPTTYPE,  KPTNUM,   KPTINDX,
     $    KPTSUB,   KOMFLG,   KOMPOP,   NOSUBS,   KANFLG,   KRFSYS,
     $    KANREC,   KANCNT,   INAME,    KANSURF,  KANINDX,
     $    JPRELEN,  NEWCARD,  JGORIT,   NUMSTID,  NUMCSEQ,
     $    IRECIX,   IRECNO,   JTLPOS,   ITITLE(9),LSREEN,
     $    NNODEFX,  NNODEFI,  NIDUM,    ISLASH,   IEQUAL,   IBLANK,
     T    IDUMMY,   N10000,   N7777,    MASKU,    MASKL,    IDIV,
     U    MACREL,   MACLOC,   MACBEGN,  MACLAST,  MACLEVEL,
     V    MACNAME(3), MACINDX(3), NMV,  JRESTOR,  MACPSH(3,25),
     W    JTEMP1,   JTEMP2,   JTEMP3,   JTEMP4,   JTEMP5,   JTEMP6,
     X    JTEMP7,   JTEMP8,   JTEMP9,   EXTRA1(20)
      EQUIVALENCE (DEFANS(1),IDEFST0(4),DEFST0(4)),(LSTYPE, KTYPE),
     1    (PTNUM, KPTNUM)
      DIMENSION IDEFST0(85),   DEFANS(26),   IDEFTAB(1000)  ,
     1    ILPTAB(200), IRPTAB(200), ITNTAB(200), JPROTP(100),
```

```
     2     IRECSV(200), MACVAR(25),  MACNOR(25),  INWORD(10), IBUUP(2),
     3     PISTO(6), IPISTO(6), IDREC(4)
      EQUIVALENCE (INWORD(14), IBUUP(16), ILPTAB(990), IRPTAB(790),
     1     ITNTAB(590), PISTO(390), IPISTO(390), IDREC(384),
     2     JPROTP(380), IRECSV(280),  MACVAR(75), MACNOR(50),
     3     IDEFTAB(1000), DEFTAB(1000))
C
      COMMON/VOCABTBL/ KOM(1100)
C
      COMMON /2/ JTABNUM,JTABL(12120)
C     *        ******          ******            *******          ****
       COMMON/CROSSEC/
     1  A,B,C,D,IDSEND,XMAT9(16),
     2   ARRAY (254),
     3   XC,YC,ZC,UX,UY,UZ,R,
     4 SCALE,
     5  DCOSCYL(9),
     6  HO(9),
     7       XS(400),YS(400),
     8  X(103),Y(103),
     9  MAJOR,MINOR,PHI,CX,CY,RADIUS,ISET,
     $  CXYZ(16),EXYZ(16),EH(16),
     $  IOKTEST,
     $  TL,
     $      ILOOP,IP,KR(200),IONCE
       EQUIVALENCE(INCHOPE,IFILL8)
       EQUIVALENCE(PLOTPLNO,IFILL9)
       DIMENSION INCAR(9)
      DATA(INCAR=5HHOPE1,5HHOPE2,5HHOPE3,5HHOPE4,5HHOPE5,5HHOPE6,
     15HHOPE7,5HHOPE8,5HHOPE9)
      INTEGER PLOTPLNO
C
C
C     THE PURPOSE OF THIS SUBROUTINE
C     IS TO SEARCH OUT AND SEQUENTIALLY IDENTIFY ≠≠HOPE≠≠ PLANES
C     IN THE DEFINED SYMBOL TABLE.  UP TO TEN ≠≠HOPE≠≠ PLANES ARE
C     POSSIBLE IN ANY ONE RUN,
C
C     IF HOPE PLANE NUMBERING IS USED THERE SHOULD BE
C     NO MISSING NUMBERS IN THE INTERVAL CHOSEN,
C     I,E, IF THERE ARE FIVE PLANES LET THE NAMES BE
C     HOPE1,HOPE2,HOPE3,HOPE4H,HOPEN
C     OR
C     HOPE1,HOPE2,HOPE3,HOPE4,HOPE
C
C
C     THIS SUBROUTINE IS CALLED INITIALLY FROM ≠≠FINI≠≠,
C     AND LATER BY ≠≠CYLINDER≠≠ AT THE END OF SEGMENT 102,
C     OR BY ≠≠DISCRETE≠≠ AT THE END OF SEGMENT 103,
C
C     THE INFORMATION RESULTING FROM ITS SEARCH IS TRANSMITTED TO
C     CROSEC BY MEANS OF THREE FLAGS, ≠≠IONCE≠≠, ≠≠PLOTPLNO≠≠, AND
C     ≠≠INCHOPE≠≠,
C
C     ≠≠IONCE≠≠, FLAG USED TO INDICATE A SINGLE HOPE PLANE
C     OR THE LAST HOPE PLANE OF CURRENT SERIES,
```

11/02/70

```
C                                                                       ISH1080
C     ≠≠PLOTPLNO≠≠, THE JTABL LOCATION OF CURRENT HOPE PLANE            ISH1090
C                                                                       ISH1100
C     ≠≠INCHOPE≠≠, THE SERIAL NUMBER OF CURRENT HOPE PLANE,             ISH1110
C     THEREBY KEEPING COUNT OF THE NUMBER OF HOPE PLANES PROCESSED      ISH1120
C                                                                       ISH1130
C     A TOP OF FORM ACTION TO SEPARATE CROSEC OUTPUT FROM EARLIER       ISH1140
C     APT OUTPUT.                                                       ISH1150
C                                                                       ISH1160
      PRINT 200                                                         ISH1170
C                                                                       ISH1180
C     DEFINED SYMBOL TABLE BEGINS AT ITAB11 AND ENDS AT ITAB12          ISH1190
C     ITS TOTAL LENGTH IS THEREFORE (ITAB12-ITAB11)+1                   ISH1200
C                                                                       ISH1210
      IF (INCHOPE.EQ.0) 110,170                                         ISH1220
  110 CONTINUE                                                          ISH1230
C                                                                       ISH1240
C     ≠≠ARRAY≠≠, IS USED IN THE PLOTS INITIALIZATION CALL.              ISH1250
C                                                                       ISH1260
      CALL PLOTS (ARRAY,254,10)                                         ISH1270
      IDSEND=(ITAB12-ITAB11)+1                                          ISH1280
      DO 160 I=1,IDSEND,2                                               ISH1290
      ISAVE=I                                                           ISH1300
      PLOTPLNO=ISAVE                                                    ISH1310
      I=I-1                                                             ISH1320
      ITRY=JTABL(ITAB11+I)                                              ISH1330
C                                                                       ISH1340
C     IF ONLY  ONE HOPE PLANE BEING PROCESSED THE NAME CAN BE EITHER    ISH1350
C     ≠≠HOPEN≠≠ OR ≠≠HOPE≠≠.                                            ISH1360
C                                                                       ISH1370
      IF ((ITRY.EQ.5HHOPEN).OR.(ITRY.EQ.4HHOPE)) 120,140                ISH1380
  120 INCHOPE=4HONCE                                                    ISH1390
  130 IFILL7=101                                                        ISH1400
      RETURN                                                            ISH1410
  140 IF (ITRY.EQ.5HHOPE1) 150,160                                      ISH1420
  150 INCHOPE=1                                                         ISH1430
      GO TO 130                                                         ISH1440
  160 I=ISAVE                                                           ISH1450
      PRINT 210                                                         ISH1460
      RETURN                                                            ISH1470
C                                                                       ISH1480
C     IF MORE THAN ONE HOPE PLANE                                       ISH1490
C                                                                       ISH1500
  170 CONTINUE                                                          ISH1510
      INCHOPE=INCHOPE+1                                                 ISH1520
      DO 190 J=1,IDSEND,2                                               ISH1530
      JSAVE=J                                                           ISH1540
      J=J-1                                                             ISH1550
      JTRY=JTABL(ITAB11+J)                                              ISH1560
      PLOTPLNO=JSAVE                                                    ISH1570
C                                                                       ISH1580
C     TERMINATION OF HOPE STRING OF PLANES ACCOMPLISHED BY USE          ISH1590
C     OF ≠≠HOPEN≠≠ OR ≠≠HOPE≠≠                                          ISH1600
C                                                                       ISH1610
      IF ((JTRY.EQ.4HHOPE).OR.(JTRY.EQ.5HHOPEN)) 120,180                ISH1620
  180 IF (JTRY.EQ.INCAR(INCHOPE)) 130,190                               ISH1630
```

11/02/70

```
  190 J=JSAVE                                                          ISH1640
      PRINT 220, INCHOPE                                               ISH1650
      RETURN                                                           ISH1660
C                                                                      ISH1670
  200 FORMAT (1H1)                                                     ISH1680
  210 FORMAT (1X,*NO HOPE, HOPE1, OR HOPEN CARD FOUND*)                ISH1690
  220 FORMAT (1X,*NO HOPE*,I5,* OR HOPEN CARD FOUND*)                  ISH1700
      END                                                              ISH1710
```

```
 ISHOPE                                              11/02/70       ED      0

                                         IDENT      ISHOPE
PROGRAM LENGTH                  00223
ENTRY POINTS       ISHOPE       00051
BLOCK NAMES
                   TIMETEST     00004
                   SYSTEMZ      00175
                   SECT1LOG     02642
                   VOCABTBL     02114
                   2            27531
                   CROSSEC      03037
EXTERNAL SYMBOLS
                   THEND,
                   Q8QDICT,
                   PLOTS
                   STH,
                   QNSINGL,
   00475 SYMBOLS
```

UNCLASSIFIED

11/02/70

```
      SUBROUTINE MM (XP,YP,ZP,XR,YR,ZR,XMAT9)
C
C     THIS SUBROUTINE PERFORMS THE COMPUTATIONS
C     NECESSARY TO CONVERT A SET
C     OF COORDINATES IN THE MAJOR SYSTEM (X,Y,Z)
C     INTO A SET OF COORDINATES IN THE HOPE SYSTEM (X≠,Y≠,Z≠)  .
C
C
C     THIS SUBROUTINE IS STORED IN MAIN BECAUSE
C     IT IS CALLED FROM MORE THAN ONE SEGMENT,
C
C     IT IS CALLED BY PROGRAM CROSEC AND
C     SUBROUTINE DLINE IN SEGMENT 101,
C     ALSO BY PROGRAM CYLINDER AND SUBROUTINE HOPARCYL IN SEGMENT 102,
C
C     (XR,YR,ZR) IS THE MAJOR SYSTEM POINT
C
C     (XP,YP,ZP) IS THE COMPUTED HOPE SYSTEM POINT
C
C     THE XMAT9 ARRAY HOLDS THE CONSTANTS NECESSARY FOR THIS CONVERSION
C
      DIMENSION XMAT9(1)
      XP=XR*XMAT9(1)+YR*XMAT9(2)+ZR*XMAT9(3)+XMAT9(4)
      YP=XR*XMAT9(5)+YR*XMAT9(6)+ZR*XMAT9(7)+XMAT9(8)
      ZP=XR*XMAT9(9)+YR*XMAT9(10)+ZR*XMAT9(11)+XMAT9(12)
      RETURN
      END
```

```
      MM                                              11/02/70        ED       0

                                IDENT      MM
PROGRAM LENGTH              00177
ENTRY POINTS      MM        00003
EXTERNAL SYMBOLS
                  Q8QDICT,
   00060 SYMBOLS
```

11/09/70

```
      PROGRAM CROSEC
      COMMON /TIMETEST/       KSETIME1, KSETIME2, KSETIME,  KSETADR
C
C     ...     COMMON AREA FOR CDC APT 3 SECTION 0
C
C     *          ******          ******          ******          ******
      COMMON /SYSTEMZ/        SYSTEM(4),KAPTCN,  KAPTTR,   KAPTIO,
     1   KAPTID,    KFLAGS(10),K0,       K1,       K2,       K3,
     2   K4,        K5,        K6,       K7,       K8,       K9,
     3   IFILL1,    IFILL2,    IFILL3,   IFILL4,
     4   KFLAG0,    KFLAG1,    KFLAG2,   KFLAG3,   KFLAG4,
     5   IFILL5,    IWAVEN,    IPTNLY,   NOPOST,   IFILL6,   KAUTOP,
     6   ICLPRT,    INDEXX,    IPLOTR,   IFILL7,   NOPLOT,   KDYNFG,
     7   LOCJPT,    LOCBEG,    KSECIN,   NCLREC,   LOCMAC,   KPOCKET,
     8   IFILL8,    IFILL9,    IFILL10,  IPOSTP(1),NUMPST,   IPOSTFL(18),
     9   TAPETB(1),CANTAP,     CLTAPE,   POCTAP,   PLOTAP,   SRFTAP,
     $   LIBTAP,    CRDTAP,    IFILL11,  CORTAP,   TAPES1,   TAPES2,
     $   TAPES3,    TAPES4,    FORTIN,   INTAPE,   IOUTAP,   PUNTAP,
     $   LSTFLG,    LTVFLG,    KONVTCL,  KINTRUPT,
     $   PI,        PI02,      DGTRD,    RDTDG,    ONE,
     $   EXTRA0(20)
      EQUIVALENCE    (PROTAP,  TAPETB)
C     *          ******          ******          ******          ******
C
C     ...     COMMON AREA FOR CDC APT 3 SECTION 1
C
C     *          ******          ******          ******          ******
      COMMON /SECT1LOG/       ITAB1,    ITAB2,    ITAB3,
     1   ITAB4,     ITAB5,     ISNAM,    ITAB11,   ITAB12,   ITAB13,
     2   JENDPTPP, JENDCAN,   JENDSTOR, JSTRICAN, JENDSYM,
     3   JCANTEMP, JRPTAB,    JLPTAB,   MAXNST,
     4   JINWD,     JCHAR,     IWDERR,   JBUFL,    NUPERP,   NUPUN,
     5   JSTYPE,    JVARS2,    SCHERR,   NMACV,    MACASN(25),
     6   INDXPT,    IPTP,      IXPT,     MODE,     EOCFLG,   LPNDFL,
     7   TRMFLG,    INTRUPT,   JUMPFL,   ICDERR,   DEBUG,
     8   MACMODE,   NESTFL,    NRESULT,  IPTLIM,   JEXEC,
     9   KTYPE,     MACTYP,
     $   IPARTERR, FINIS,     IOFLG,    MACDEL,   JSUBER,   NUMBERR,
     $   DEFSTO(85),     DEFTAB(1000),  ZSUR(30),
     $   XMAT4(16),XMAT3(16),XMAT2(16),XMAT1(16),TMATX(16),
     $   ISTDMODE, ISTDLIT,   ISTDTBL,  ISTDINDX, ISTDTYPE, ISTDWD,
     $   JPTIND,    KPTCODE,   KPTNAME,  KPTTYPE,  KPTNUM,   KPTINDX,
     $   KPTSUB,    KOMFLG,    KOMPOP,   NOSUBS,   KANFLG,   KRFSYS,
     $   KANREC,    KANCNT,    INAME,    KANSURF,  KANINDX,
     $   JPRELEN,   NEWCARD,   JGORIT,   NUMSTID,  NUMCSEQ,
     $   IRECIX,    IRECNO,    JTLPOS,   ITITLE(9),LSRECN,
     $   NNODEFX,   NNODEFI,   NIDUM,    ISLASH,   IEQUAL,   IBLANK,
     T   IDUMMY,    N10000,    N7777,    MASKU,    MASKL,    IDIV,
     U   MACREL,    MACLOC,    MACBEGN,  MACLAST,  MACLEVEL,
     V   MACNAME(3), MACINDX(3), NMV,    JRESTOR,  MACPSH(3,25),
     W   JTEMP1,    JTEMP2,    JTEMP3,   JTEMP4,   JTEMP5,   JTEMP6,
     X   JTEMP7,    JTEMP8,    JTEMP9,   EXTRA1(20)
      EQUIVALENCE (DEFANS(1),IDEFSTO(4),DEFSTO(4)),(LSTYPE, KTYPE),
     1   (PTNUM, KPTNUM)
      DIMENSION IDEFSTO(85),  DEFANS(26),  IDEFTAB(1000)  ,
     1   ILPTAB(200), IRPTAB(200), ITNTAB(200), JPROTP(100),
```

```
2    IRECSV(200), MACVAR(25),  MACNOR(25),  INWORD(10), IBUUP(2),
3    PISTO(6), IPISTO(6), IDREC(4)
 EQUIVALENCE (INWORD(14), IBUUP(16), ILPTAB(990), IRPTAB(790),
1    ITNTAB(590), PISTO(390), IPISTO(390), IDREC(384),
2    JPROTP(380), IRECSV(280),  MACVAR(75), MACNOR(50),
3    IDEFTAB(1000), DEFTAB(1000))
C
 COMMON/VOCABTBL/ KOM(1100)
C
 COMMON /2/ JTABNUM,JTABL(12120)
C     *          ******          ******          *******          ****
  COMMON/CROSSEC/
1  A,B,C,D,IDSEND,XMAT9(16),
2   ARRAY (254),
3   XC,YC,ZC,UX,UY,UZ,R,
4 SCALE,
5  DCOSCYL(9),
6  HO(9),
7      XS(400),YS(400),
8  X(103),Y(103),
9  MAJOR,MINOR,PHI,CX,CY,RADIUS,ISET,
$  CXYZ(16),EXYZ(16),EH(16),
$  IOKTEST,
$  TL,
$     ILOOP,IP,KR(200),IONCE
  EQUIVALENCE(INCHOPE,IFILL8)
  EQUIVALENCE(PLOTPLNO,IFILL9)
  DATA (IONEPART=8H        )
C
C
C       *         *         *         *         *         *         *
C       *         *        C O M M E N T        *         *         *
C       *         *         *         *         *         *         *
C       *         *         *         *         *         *         *
C
C     CROSEC (MOD 1) PROVIDES A MEANS OF EXTENDING THE USE OF THE PLANE
C     SURFACES(DEFINED BY THE PART PROGRAMMER IN THE PART PROGRAM) BY
C     PROVIDING A PLOTTING CAPABILITY IN WHICH THE LINES OF
C     INTERSECTION, WITHIN SPECIFIED LIMITS, BETWEEN A CROSS-SECTIONAL
C     PLANE AND ALL OTHER DEFINED PLANES CAN BE SHOWN.
C
C     THE CROSS SECTIONAL PLANE FOR THE PLOT AND ITS DIMENSIONAL LIMITS
C     ARE CONTROLLED BY MEANS OF ONE PLANE DEFINITION (NAMED ≠≠HOPE≠≠)
C     AND TWO POINT DEFINITIONS (NAMED ≠≠HOXYMIN≠≠ AND ≠≠HOXYMAX≠≠)
C     ADDED TO THE PART PROGRAM.
C
C     THE PLOT IS SUPPLEMENTED BY PRINTER OUTPUT THAT IDENTIFIES
C     A LINE OF INTERSECTION BY GIVING
C     ITS SLOPE AND INTERSECTION
C     WITH RESPECT TO THE HOPE PLANE AXES.
C
C     CROSEC (MOD 2.0) PROVIDES CROSS SECTIONS THROUGH CYLINDERS,
C     WHICH INCLUDES CIRCLES, AND SPHERES.
C
C     THE PROGRAM IS WRITTEN IN THE FRAMEWORK OF THE CDC 3800 APT 2.1
C     CONFIGURATION AND THREE SEGMENTS HAVE BEEN ADDED IN SECTION I.
```

```
C      THE FIRST ONE, SEGMENT 101, PROCESSES PLANE INTERSECTIONS,
C      ESSENTIALLY IT PERFORMS THE WORK OF THE OLD CROSEC (MOD 1.0),
C      THE SECOND ONE, SEGMENT 102, PROCESSES CYLINDERS/CIRCLES
C      AND SPHERE INTERSECTIONS,
C      AND THE THIRD ONE, SEGMENT 103, PROCESSES THE DISCRETE POINTS
C      FOR THE INTERSECTIONS FROM SEGMENT 102.
C
C      CROSS SECTIONAL PROCESSING COMMENCES FROM SUBROUTINE FINI WHEN
C      ≠≠IS HOPE≠≠ IS CALLED.
C
C
C         *          *          *          *          *          *          *
C
C      THE ≠≠PROGRAM≠≠ CROSEC CONTROLS SEGMENT 101
C
C      COMMENTS ON THOSE PORTIONS OF CROSSEC COMMON
C      PERTINENT TO SEGMENT 101,
C
C      ≠HOPE≠ IS THE NAME GIVEN TO THE PLANE OF THE CROSS SECTION
C
C       NOTE--THE HOPE SYSTEM IS CALLED THE PRIME SYSTEM. THE TERMS ARE
C      USED INTERCHANGABLY AND IS DENOTED BY THE SYMBOL≠.
C      THIS SYMBOL IS USUALLY USED AS A SUFFIX, SUCH AS X≠
C
C
C      A,B,C,D, ARE THE HOPE PLANE CONSTANTS
C      OBTAINED FROM THE CANONICAL FORM
C
C      ISEND IS THE LENGTH OF THE DEFINED SYMBOL TABLE
C
C
C      THE XMAT9 ARRAY WILL HOLD THE MATRIX OF COEFFICIENTS FOR
C      COORDINATE  CONVERSION (TRANSLATION AND/OR ROTATION)
C      FROM THE MAJOR SYSTEM TO THE HOPE SYSTEM
C
C      FROM THE EQUATIONS
C
C       X≠ = T11(X-X0) + T21(Y-Y0) + T31(Z-Z0)
C
C       Y≠ = T12(X-X0) + T22(Y-Y0) + T32(Z-Z0)
C
C       Z≠ = T13(X-X0) + T32(Y-Y0) + T33(Z-Z0)
C
C
C      WHERE-
C
C          THE X≠ AXIS HAS DIRECTION COSINES- T11, T21, T31
C
C          THE Y≠ AXIS HAS DIRECTION COSINES- T12, T22, T32
C
C          THE Z≠ AXIS HAS DIRECTION COSINES- T13, T23, T33
C
C          THE PRIME SYSTEM ORIGIN IS AT (X0,Y0,Z0)
C
C      IT IS FORMED AS FOLLOWS-
C
```

```
C          I                                                  I
C          I                                                  I
C          I                                                  I
C          I      T11        T21        T31        C1         I
C          I                                                  I
C          I      T12        T22        T32        C2         I
C          I                                                  I
C          I      T13        T23        T33        C3         I
C          I                                                  I
C          I                                                  I
C          I
C
C     WHERE-
C        C1 = -(T11*X0   +   T21*Y0    +   T31*Z0)
C
C        C2 = -(T12*X0   +   T22*Y0    +   T32*Z0)
C
C        C3 = -(T13*X0   +   T23*Y0    +   T33*Z0)
C
C
C     THE KR ARRAY HOLDS THE (X≠Y≠) INFORMATION OF ALL POINTS OBTAINED
C     ON ONE COMPLETE PASS THROUGH THE INNER LOOP COMMENCING AT
C     CARD, ≠≠CRS6160≠≠.
C
C     ≠≠IP≠≠ COUNTS THE NUMBER OF PAIRS
C
      REAL KR
C
C
C     ≠≠IONCE≠≠ IS A FLAG THAT CONTROLS THE CALLING OF SUBROUTINE DLINE
C
C     ≠≠SCALE≠≠, HOLDS THE PLOT SCALE COMPUTED
C     FROM THE HOXYMIN AND HOXYMAX CARDS.
C
C     ≠≠IOKTEST≠≠, IS A FLAG USED TO TEST THE POINTS OF INTERSECTION,
C     SPECIFICALLY, IT RECEIVES THE OUTPLT OF FUNCTION ISITOK.
C
C       *        *        *        *        *        *        *
C     THE PROGRAMMING ACTION STARTS HERE
C
C     ≠≠YES≠≠ AND ≠≠NO≠≠ ARE THE TWO POSSIBLE ANSWERS FOR
C     FUNCTION ISITOK.
C
C
      INTEGER PLOTPLNO
      YES=1
      NO=0
C
C
C       *        *        *        *        *        *        *
C       *        *      C O M M E N T      *        *        *
C       *        *        *        *        *        *        *
C       *        *        *        *        *        *        *
C     DEFINING THE CANONICAL FORM OF THE PLANE AS
C        X * COS ALPHA  + Y * COS BETA + Z * COS GAMMA  = P
C     WHERE THE COSINE TERMS ARE DIRECTION COSINES OF X, Y, AND Z
```

11/09/70

```
C     RESPECTFULLY.
C     THE CANONICAL FORM OF THE ≠HOPE≠ PLANE IS CALLED AND STORED IN
C     THE REFERENCE ARRAY, WHERE
C           ≠RCOSA≠ CONTAINS COSINE ALPHA
C           ≠RCOSB≠ CONTAINS COSINE BETA
C           ≠RCOSG≠ CONTAINS COSINE GAMMA ,AND
C           ≠R HOPE P≠ CONTAINS THE ≠P≠ OR CONSTANT VALUE
C        *         *         *         *         *         *         *
C        *         *         *         *         *         *         *
C
C     ≠≠KANSURF≠≠ IS DESIGNED TO HOLD POINTERS TO CANONICAL FORMS,
C     ON CARD ≠≠CRS2340≠≠ IT IS HOLDING THE POINTER
C     TO THE ≠≠HOPE≠≠ PLANE CANONICAL FORM.
C
      KANSURF=JTABL(ITAB11+PLOTPLNO)
      IHONAME=JTABL(ITAB11+PLOTPLNO-1)
C
C     THE ≠≠CANGET≠≠ SUBROUTINE FETCHES THE CANONICAL FORM POINTED TO
C     BY ≠≠KANSURF≠≠ AND STORES ITS ELEMENTS IN THE ≠≠DEFSTO≠≠ ARRAY.
C
      CALL CANGET
      RCOSA=DEFSTO(4)
      RCOSB=DEFSTO(5)
      RCOSG=DEFSTO(6)
      RHOPEP=DEFSTO(7)
C
C     USING THE SHORTHAND SYMBOLS FOR THE PLANE CONSTANTS-A,B,C,D
C
      A=RCOSA
      B=RCOSB
      C=RCOSG
      D=RHOPEP
      PRINT 370, IHONAME,A,B,C,D
C
C     SUBROUTINE TESTHOPE FILLS IN THE ≠≠XMAT9≠≠ ARRAY
C
      CALL TESTHOPE
C
C     ANOTHER TOP-OF-FORM
C
      PRINT 360
C        *         *         C O M M E N T         *         *         *
C        *         *         *         *         *         *         *
C     SINCE THE COSINE OF THE ANGLE BETWEEN TWO PLANES IS EQUAL TO THE
C     SUM OF THE PRODUCTS OF CORRESPONDING DIRECTION COSINES, THIS FACT
C     IS TAKEN ADVANTAGE OF BY COMPUTING THE ANGLE BETWEEN EACH DEFINED
C     PLANE AND THE REFERENCE PLANE.  SPECIAL ATTENTION IS CALLED TO
C     THOSE PLANES THAT ARE EITHER PARALLEL OR PERPENDICULAR
C     TO THE REFERENCE PLANE.
C        *         *         *         *         *         *         *
C        *         *         *         *         *         *         *
       PRINT 380,IHONAME
      DO 120 J=1,IDSEND,2
      I=ITAB11+J-1
C
C
```

11/09/70

```
C     A MORE DETAILED EXAMINATION OF THE SECOND WORD IN THE DST PAIRS
C     IS NEEDED TO AVOID PICKING UP ITEMS THAT ARE NOT PLANES.
C     ≠≠IONE≠≠ PICKS UP THE FIRST WORD OF THE PAIR
C     ≠≠ITWO≠≠ PICKS UP THE SECOND WORD OF THE PAIR
C
C
      IONE=JTABL(I)
      ITWO=JTABL(I+1)
      IONEPART=IONE.AND.77777777000000008
      IX=(IONE.AND.7700000000000000B)
C
C     BYPASS OCCURS IF THE NAME CONTAINS AN≠≠X≠≠ OR THE WORD≠≠HOPE≠≠.
C
      IF ((IONEPART.EQ.8HHOPE0000).OR.(IX.EQ.8HX0000000)) 120,10
   10 CONTINUE
C
C     IF A NAME IS UNDEFINED THE SECOND WORD CONTAINS THE BCD WORD
C     ≠≠NODEFX S≠≠.
C
C     IF A NAME IS INCORRECTLY DEFINED
C     THE SECOND WORD CONTAINS THE BCD WORD ≠≠NODEFI S≠≠.
C     THIS TEST PREVENTS OPERATING ON THESE SITUATIONS.
C
      IF (ITWO.EQ.NNODEFX.OR.ITWO.EQ.NNODEFI) 120,20
C
C     ≠≠ISTDWD≠≠IS SET EQUAL TO THE SECOND WORD BECAUSE
C     IT IS NOW ASSUMMED TO BE A STANDARD WORD BY APT DEFINITIONS.
C
   20 ISTDWD=ITWO
C
C     SUBROUTINE STDUNPK IS CALLED UPON TO UNPACK THE STANDARD WORD
C
      ISTDMODE=0
      ISTDTBL=0
      ISTDINDX=0
      ISTDTYPE=0
      CALL STDUNPK
C
C     ≠≠ISTDTBL≠≠ RECEIVES BYTA AND A ≠≠4≠≠ REPRESENTS A CANONICAL FORM
C
      IF (ISTDTBL.EQ.4) 30,120
C
C     IT IS NOW SAFE TO PROCEED AND EXAMINE THE CANONICAL FORM ITSELF
C
   30 KANSURF=ITWO
      CALL CANGET
C
C     ≠≠ISAM≠≠ PICKS UP THE 4 RIGHTMOST BITS
C     OF THE FIRST CANONICAL WORD
C
      ISAM=(DEFSTO(1).AND.7777B)
C
C     IF ≠≠ISAM≠≠ IS EQUAL TO A ≠≠3≠≠ THEN IT IS A PLANE CANONICAL FORM
C
      IF ((ISAM.EQ.2).OR.(ISAM.EQ.3)) 40,120
C
```

11/09/70

```
C     THE SUM OF PRODUCTS OF CORRESPONDING DIRECTION COSINES            CRS3310
C     OF TWO PLANES.                                                    CRS3320
C                                                                       CRS3330
   40 COSSAM=RCOSA*DEFSTO(4)+RCOSB*DEFSTO(5)+RCOSG*DEFSTO(6)            CRS3340
C                                                                       CRS3350
C     IF ≠≠COSSAM≠≠ IS NEGATIVE, CHECK IT FURTHER AT STATEMENT 938      CRS3360
C                                                                       CRS3370
C     IF ≠≠COSSAM≠≠IS ZERO THEN THE PLANES ARE PERPENDICULAR            CRS3380
C                                                                       CRS3390
      IF (COSSAM) 100,50,60                                             CRS3400
   50 PRINT 390, IONE                                                   CRS3410
      GO TO 120                                                         CRS3420
C                                                                       CRS3430
C     IF ≠≠COSSAM≠≠IS ≠≠1≠≠ THEN THE PLANES ARE PARALLEL                CRS3440
C                                                                       CRS3450
   60 EXCESS=COSSAM-1.0                                                 CRS3460
      IF (EXCESS) 110,90,70                                             CRS3470
   70 IF (EXCESS.LT.0.0001) 90,80                                       CRS3480
   80 PRINT 400, IONE,COSSAM                                            CRS3490
      GO TO 120                                                         CRS3500
   90 PRINT 410, IONE                                                   CRS3510
      GO TO 120                                                         CRS3520
C                                                                       CRS3530
C     IF ≠≠COSSAM≠≠ BETWEEN -1 AND 0                                    CRS3540
C     PROCEED TO TAKE THE ARC COSINE                                    CRS3550
C     OTHERWISE GO TO END OF LOOP                                       CRS3560
C                                                                       CRS3570
  100 IF (COSSAM.GT.(-1.0)) 110,120                                     CRS3580
C                                                                       CRS3590
C     IF ≠≠COSSAM≠≠ IS NON-ZERO, AND NOT = 1.  TAKE THE ARC COSINE      CRS3600
C     AND CONVERT TO DEGREES                                            CRS3610
C                                                                       CRS3620
  110 ANGSAM=ACOS(COSSAM)*57.32                                         CRS3630
      PRINT 420, IONE,ANGSAM                                            CRS3640
  120 CONTINUE                                                          CRS3650
C                                                                       CRS3660
C     NOTHING IS DONE WITH THIS INFORMATION                             CRS3670
C     ON ANGLES BETWEEN THE PLANES                                      CRS3671
C     OTHER THAN PRINTING IT OUT FOR ITS USEFULNESS TO THE PROGRAMMER   CRS3680
C                                                                       CRS3690
C                                                                       CRS3700
      PRINT 360                                                         CRS3710
C        *       *       *       *       *       *       *              CRS3720
C        *       *       *       *       *       *       *              CRS3730
C        *       *       *       *       *       *       *              CRS3740
C        *       *     C O M M E N T     *       *       *              CRS3750
C        *       *       *       *       *       *       *              CRS3760
C        *       *       *       *       *       *       *              CRS3770
C     THE MINIMUM AND MAXIMUM VALUES OF THE MAJOR COORDINATE SYSTEM     CRS3780
C     VARIABLES (I.E. X,Y,Z) ARE INTRODUCED THROUGH A SEQUENTIAL PAIR   CRS3790
C     OF SPECIALLY NAMED POINTS - ≠≠HOXYMIN≠≠ AND ≠≠HOXYMAX≠≠.          CRS3800
C                                                                       CRS3810
C     THESE VALUES ARE PICKED UP VIA THE DEFINED SYMBOL TABLE           CRS3820
C                                                                       CRS3830
C     THE COORESPONDING VALUES IN THE PRIME SYSTEM ARE COMPUTED.        CRS3840
C     (E.G.X≠MIN= F(XMIN,YMIN,ZMIN)                                     CRS3850
```

```
C
C     A MINIMUM AND A MAXIMUM VECTOR MAGNITUDE IN THE PRIME SYSTEM ARE
C     COMPUTED
C
C     (E.G. VMIN =SQRT(X≠MIN**2 + Y≠MIN**2 + Z≠MIN**2)
C
C     THE LARGEST OF THESE VECTORS IS SCALED TO COVER 5 INCHES OF PLOT
C     LENGTH.
C
C     THE POINTS OF INTERSECTION ARE COMPUTED
C               FIRST IN X,Y,Z  COORDINATES
C               SECOND IN X≠,Y≠,Z≠COORDINATES
C
C     THE Z≠ VALUES ARE IGNORED AND SHOULD BE ZERO
C
C     THE PLOTTING COORDINATES (XP,YP) ARE COMPUTED USING THE SCALE
C     FACTOR AND AN ADDITIVE BIAS OF 5.0 IN BOTH DIRECTIONS.
C
C
C     IF THE RESULTING VALUES OF XP AND YP ARE NEGATIVE OR EXCEED 10.0
C     THEY ARE REJECTED AS EXCEEDING THE BOUNDS OF THE LIMITING VECTOR
C
C
C        *         *         *         *         *         *         *
C        *         *         *         *         *         *         *
C        *         *         *         *         *         *         *
C        *         *         *         *         *         *         *
C
C     THE DST TABLE IS SEARCHED FOR THE NAMES
C     ≠≠HOXYMIN≠≠ AND ≠≠HOXYMAX≠≠.
C
      DO 140 K=1,IDSEND,2
      KSAVE=K
      K=K-1
C                              H O X Y M I N
      IF (JTABL(ITAB11+K).EQ.30466770443145608.AND.JTABL(ITAB11+K+2).EQ.
     1304667704421676 0B) 130,140
C      H O X Y M A X
C
C     INCREMENTAL INDECIES TO SECOND WORD LOCATIONS IN DST
C
  130 IMINNO=K+1
      IMAXNO=K+3
      GO TO 150
  140 K=KSAVE
      PRINT 430
      STOP
C
C     THE MINIMUM VALUES ARE EXTRACTED FROM THE HOXYMIN CANONICAL FORM
C
  150 KANSURF=JTABL(ITAB11+IMINNO)
      CALL CANGET
      RAWXMIN=DEFSTO(4)
      RAWYMIN=DEFSTO(5)
      RAWZMIN=DEFSTO(6)
C
```

```
C     THE MAXIMUM VALUES ARE EXTRACTED FROM THE HOXYMAX CANONICAL FORM
C
      KANSURF=JTABL(ITAB11+IMAXNO)
      CALL CANGET
      RAWXMAX=DEFSTO(4)
      RAWYMAX=DEFSTO(5)
      RAWZMAX=DEFSTO(6)
      PRINT 440
      PRINT 450, RAWXMIN,RAWYMIN,RAWZMIN,RAWXMAX,RAWYMAX,RAWZMAX
C
C     THE COORESPONDING MINIMUM VALUES IN THE HOPE PLANE,
C     XPMIN,YPMIN,ZPMIN
C     ARE COMPUTED IN SUBROUTINE MM USING THE XMAT9 MATRIX
C
      CALL MM (XPMIN,YPMIN,ZPMIN,RAWXMIN,RAWYMIN,RAWZMIN,XMAT9)
C
C     THE SAME WITH THE MAXIMUM VALUES, XPMAX, YPMAX, ZPMAX
C
      CALL MM (XPMAX,YPMAX,ZPMAX,RAWXMAX,RAWYMAX,RAWZMAX,XMAT9)
C
C     THE MINIMUM AND MAXIMUM VECTOR MAGNITUDES ARE COMPUTED
C
      MINVCTOR=SQRT(XPMIN**2+YPMIN**2+ZPMIN**2)
      MAXVCTOR=SQRT(XPMAX**2+YPMAX**2+ZPMAX**2)
C
C     THE GREATEST OF THESE IN MAGNITUDE IS SET EQUAL TO ≠≠REACH≠≠
C
      IF (MINVCTOR.GT.MAXVCTOR) 160,170
  160 REACH=MINVCTOR
      GO TO 180
  170 REACH=MAXVCTOR
      GO TO 180
C
C     THE SCALE FACTOR- REACH MUST FIT INTO 5.0 INCHES OF PLOT LENGTH
C
  180 SCALE=5.0/REACH
C
C     PRINT OUT THE CAL-COMP PLOT SCALE
C
      PRINT 460
      PRINT 470, SCALE
C
C
C     CALCULATE AND PRINT OUT THE GERBER SCALE FACTOR NEEDED TO
C     PRODUCE A 1:1 PLOT, THAT IS, A PLOT ON WHICH ONE UNIT OF LENGTH
C     IS EQUAL TO ONE UNIT OF LENGTH ON THE ORIGINAL PART DRAWING,
C
      SCLFAC=1/SCALE
      PRINT 480, SCLFAC
      PRINT 360
C
C     PLACE A RIGHT ANGLE INTERSECTION AT THE ORIGIN (5.0,5.0)
C
      CALL SYMBOL (5.0,5.0,0.20,3,0,-1)
C
C
```

```
C     PLACE A SCALE MARK IN THE LOWER LEFT HAND CORNER
C     SHOWING THE CAL-COMP DISTANCE CORRESPONDING TO ONE INCH OF
C     PART LENGTH.
C
      CALL PLOT (0,0,3)
      CALL PLOT (0,0,2)
      CALL PLOT (0,-0.25,2)
      CALL PLOT (SCALE,-0,25,2)
      CALL PLOT (SCALE,0,2)
      CALL PLOT (SCALE,0,3)
C
C        *          *          *          *          *          *          *
C        *          *          *          *          *          *          *
C        *          *         C O M M E N T          *          *          *
C        *          *          *          *          *          *          *
C        *          *          *          *          *          *          *
C           THE SUBROUTINE ≠SIMEQ≠ THAT SOLVES THREE PLANE EQUATIONS
C     SIMULTANEOUSLY IS UTILIZED TO OBTAIN POINTS OF INTERSECTION
C     ON THE ≠≠HOPE≠≠ PLANE.  A LINE OF INTERSECTION IS IDENTIFIED
C     WHEN A PLANE IS INVOLVED IN MORE THAN ONE POINT.  FOR THIS
C     REASON AN OUTER LOOP IS CYCLED ≠≠IDSEND≠≠ TIMES, AND AN INNER
C     LOOP CYCLED ≠≠IDSEND * IDSEND≠≠ TIMES SO THAT EACH PLANE IS
C     HELD CONSTANT (ALONG WITH ≠≠HOPE≠≠ OF COURSE) AS ALL OTHER
C     PLANES ARE BROUGHT IN FOR TESTING.
C        *          *          *          *          *          *          *
C        *          *          *          *          *          *          *
C        *          *          *          *          *          *          *
C        *          *          *          *          *          *          *
C
C     TITLE FOR THE NEXT PAGE OF PRINT OUT
C
      PRINT 490, IHONAME
C
C     THE ≠≠DEFTAB≠≠ ARRAY HOLDS THE INPUT INFORMATION FOR
C     SUBROUTINE SIMEQ. HERE THE HOPE PLANE CONSTANTS ARE
C     INSERTED. THEY WILL NOT BE CHANGED SINCE THE HOPE PLANE
C     IS INVOLVED IN ALL THE POINTS OF INTERSECTION
C
      DEFTAB(108)=RCOSA
      DEFTAB(109)=RCOSB
      DEFTAB(110)=RCOSG
      DEFTAB(111)=RHOPEP
C
C     INITIALIZATION OF THREE FLAGS
C
C     ≠≠IONCE≠≠ CONTROLS THE CALL TO SUBROUTINE DLINE
C         A ≠≠1≠≠ VALUE MEANS OFF
C         A ≠≠0≠≠ VALUE MEANS ON
C         THE INITIAL VALUE IS OFF
C         IT IS TURNED ON BY THE ACCEPTANCE OF A VALID POINT IN
C     FUNCTION ISITOK.
C
C         IT IS TURNED OFF AFTER EACH CALL
C
C     ≠≠IP≠≠ IS THE COUNTER IN THE ≠≠KR≠≠ ARRAY
C
```

```
C
C     ≠≠IPTNO≠≠ IS THE INTERSECTION POINT NUMBER COUNTER
C
      IONCE=1
      IP=1
      IPTNO=1
C
C     THE START OF THE OUTER LOOP
C
      DO 330 KK=1,IDSEND,2
      K=KK-1
C
C     SHOULD ≠≠DLINE≠≠ BE CALLED?
C
      IF (IONCE.EQ.1) 200,190
C
C     THE IDEA IS TO CALL DLINE FOR THE PURPOSES OF DRAWING A
C     LINE BETWEEN THE MINIMUM AND MAXIMUM
C     POINTS IN THE SET OF ALL THE INTERSECTION POINTS OBTAINED
C     DURING THE JUST COMPLETED PASS THRU THE INNER LOOP.  IF NO POINTS
C     WERE CREATED , NO CALL IS MADE TO ≠≠DLINE≠≠.
C
  190 CONTINUE
      CALL DLINE (NAME1,IP,KR,SCALE)
C
C     CLOSE THE GATE
C
      IONCE=1
  200 CONTINUE
      I=ITAB11+K
C
C     SEE REMARKS IN VICINITY OF STATEMENT 921
C
      IONE=JTABL(I)
      ITWO=JTABL(I+1)
      IONEPART=IONE.AND.77777777000000000B
      IX=(IONE.AND.77000000000000000B)
C
C     BYPASS OCCURS IF THE NAME CONTAINS AN≠≠X≠≠ OR THE WORD≠≠HOPE≠≠.
C
      IF ((IONEPART.EQ.8HHOPE0000).OR.(IX.EQ.8HX0000000)) 330,210
  210 CONTINUE
      IF (ITWO.EQ.NNODEFX.OR.ITWO.EQ.NNODEFI) 330,220
  220 ISTDWD=ITWO
      CALL STDUNPK
      IF (ISTDTBL.EQ.4) 230,330
  230 CONTINUE
      KANSURF=ITWO
      CALL CANGET
C
C     THE CANONICAL FORM OF A PLANE HAS BEEN ISOLATED AND ITS NAME AND
C     CONSTANTS ARE NOW PICKED UP FOR USE. THESE VALUES WILL BE USED
C     DURING THE COMPLETE PROCESSING OF THE INNER LOOP.
C
      ISAM=(DEFSTO(1).AND.7777B)
      IF ((ISAM.EQ.2).OR.(ISAM.EQ.3)) 240,330
```

```
  240 NAME1=IØNE
      DEFTAB(104)=DEFSTØ(4)
      DEFTAB(105)=DEFSTØ(5)
      DEFTAB(106)=DEFSTØ(6)
      DEFTAB(107)=DEFSTØ(7)
C
C     START ØF THE INNER LØØP
C
      DØ 320 L=1,IDSEND,2
      M=ITAB11+L-1
C
C     SEE REMARKS IN VICINITY ØF STATEMENT 921
C
      IØNE=JTABL(M)
      ITWØ=JTABL(M+1)
      IØNEPART=IØNE.AND.77777777000000008
      IX=(IØNE.AND.77000000000000008)
C
C     BYPASS ØCCURS IF THE NAME CØNTAINS AN≠≠X≠≠ ØR THE WØRD≠≠HØPE≠≠.
C
      IF ((IØNEPART.EQ.8HHØPE0000).ØR.(IX.EQ.8HX0000000)) 320,250
  250 CØNTINUE
      IF (ITWØ.EQ.NNØDEFX.ØR.ITWØ.EQ.NNØDEFI) 320,260
  260 ISTDWD=ITWØ
      CALL STDUNPK
      IF (ISTDTBL.EQ.4) 270,320
  270 CØNTINUE
      KANSLRF=ITWØ
      CALL CANGET
C
C     THE CANØNICAL FØRM ØF THE THIRD PLANE HAS BEEN ISØLATED.
C     ITS NAME AND CØNSTANTS ARE NØW PICKED UP FØR USE.
C
      ISAM=(DEFSTØ(1).AND.7777B)
      IF ((ISAM.EQ.2).ØR.(ISAM.EQ.3)) 280,320
  280 NAME2=IØNE
      DEFTAB(100)=DEFSTØ(4)
      DEFTAB(101)=DEFSTØ(5)
      DEFTAB(102)=DEFSTØ(6)
      DEFTAB(103)=DEFSTØ(7)
C
C     THE ERRØR MESSAGE INDICATØR IS ZERØED
C
      JSUBER=0
C
C     READY TØ SØLVE THREE PLANE EQUATIØNS SIMULTANEØUSLY
C
      CALL SIMEQ
      IF (JSUBER.EQ.1005) 290,300
  290 CØNTINUE
C
C     UNSUCCESSFUL SØLUTIØN
C
      GØ TØ 320
C
C     SATISFACTØRY SØLUTIØN PRØCEED
```

```
C
  300 CONTINUE
C
C     XR,YR,ZR ARE THE COORDINATES OF THE POINT IN THE MAJOR SYSTEM.
C
      XR=DEFTAB(112)
      YR=DEFTAB(113)
      ZR=DEFTAB(114)
C
C     XP, YP, ZP ARE THE COORDINATES
C     OF THE POINT IN THE HOPE PLANE SYSTEM
C
      XP=0
      YP=0
      ZP=0
      CALL MM (XP,YP,ZP,XR,YR,ZR,XMAT9)
C
C     X, Y      ARE THE COORDINATES OF THE POINT FOR THE PLOTTER.
C
      X=XP*SCALE+5.0
      Y=YP*SCALE+5.0
      Z=ZP
C
C     TEST TO DETERMINE IF THE POINT IS ACCEPTABLE-
C         ARE (XR,YR,ZR) WITHIN THE X,Y,Z LIMITS?
C         ARE (X,Y)       WITHIN THE PLOT LIMITS?
C
      ISTESTOK=ISITOK(XR,YR,ZR,X,Y,Z,RAWXMIN,RAWYMIN,RAWZMIN,RAWXMAX,RAW
     1YMAX,RAWZMAX,YES,NO,IONCE,KR,IP,SCALE)
      IF (ISTESTOK.EQ.NO) 320,310
C
C     POINT ACCEPTED CONTINUE
C
  310 CONTINUE
C
C
C
C
      IPTNO=IPTNO+1
  320 CONTINUE
  330 CONTINUE
C     DRAW A LINE THROUGH THE LAST SET OF POINTS IF THEY EXIST
      IF (IONCE.EQ.0) 340,350
  340 CONTINUE
      CALL DLINE (NAME1,IP,KR,SCALE)
  350 CONTINUE
C
C     THE LINE AND PLANE CANNONICAL FORMS HAVE
C     ALL BEEN OPERATED ON.
C     NOW GO TO SEGMENT 2 AND LOOK FOR CYLINDERS AND SPHERES
      IDSEND=IDSEND+1
      IFILL7=102
      CALL PRCNTL
C
  360 FORMAT (1H1)
  370 FORMAT (1X,A8,/,1X,*A,B,C,D*,4F10.7)
```

11/09/70

```
380  FORMAT (30X,*ANGLES BETWEEN *,A8,* AND THE DEFINED PLANES*,/////)CRS7210
390 FORMAT (1X,A8,1X,32HPERPENDICULAR TO REFERENCE PLANE,///)          CRS7220
400 FORMAT (1X,*FOR *,A8,*COSINE ARGUMENT GT 1, IS *,E20.7)            CRS7230
410 FORMAT (1X,A8,1X,27HPARALLEL TO REFERENCE PLANE,///)               CRS7240
420 FORMAT (1X,A8,1X,2HAT,F10.1,1X,26HDEGREES TO REFERENCE PLANE,///) CRS7250
430 FORMAT (1X,*HOXYMIN AND HOXYMAX NOT FOUND*,/,1X,*THEY MUST BE SEQUCRS7260
   1ENTIAL, MIN THEN MAX*)                                             CRS7270
440 FORMAT (1X,*IN ORIGINAL COORDINATE SYSTEM*,/,/,/)                  CRS7280
450 FORMAT (5X,*MINIMUM VALUES*,/,/,3(10X,F10.5,/),/,5X,*MAXIMUM VALUECRS7290
   1S*,/,/,3(10X,F10.5,/))                                             CRS7300
460 FORMAT (/,/,1X,*IN HOPE PLANE*,/,/)                                CRS7310
470 FORMAT (1X,*SCALE..,*,F10.5,* INCHES OF PLOT LENGTH = 1 INCH OF PACRS7320
   1RT LENGTH*)                                                        CRS7330
480 FORMAT (/,/,1X,*TO OBTAIN A 1 TO 1 PLOT ON THE GERBER USE A SCALE*CRS7340
   1,* FACTOR IN BOTH X AND Y OF *,F10.5)                              CRS7350
490 FORMAT (30X,A8,* INTERSECTIONS WITH ALLOWED DEFINED SURFACES*,////CRS7360
   1)                                                                  CRS7370
    END                                                                CRS7380
```

```
  CROSEC                                              11/09/70       ED      0

                                      IDENT      CROSEC
PROGRAM LENGTH                 01467
ENTRY POINTS      CROSEC       00276
BLOCK NAMES
                  TIMETEST     00004
                  SYSTEMZ      00175
                  SECT1LOG     02642
                  VOCABTBL     02114
                  2            27531
                  CROSSEC      03037
EXTERNAL SYMBOLS
                  Q8QENTRY
                  THEND.
                  Q8QSTOPS
                  Q1Q10100
                  Q8QDICT.
                  CANGET
                  TESTHOPE
                  STDUNPK
                  MM
                  SYMBOL
                  PLOT
                  DLINE
                  SIMEQ
                  ISITOK
                  PRCNTL
                  SQRTF
                  ACOSF
                  STH.
                  QNSINGL.
   00663 SYMBOLS
```

11/02/70

```
      SUBROUTINE TESTHOPE
C
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),FH(16),IOKTEST,TL,IL
     3OOP,IP,KR(200),IONCE
C
C
C     THE PURPOSE OF SUBROUTINE TESTHOPE IS TO EXAMINE THE DIRECTION
C     COSINES OF THE HOPE PLANE IN ORDER TO DECIDE ON THE ORIENTATION
C     OF THE HOPE PLANE COORDINATE SYSTEM
C
C
C      A,B,C,D ARE THE HOPE PLANE CONSTANTS
C
C     THE XMAT9 ARRAY HOLDS THE CONVERSION  CONSTANTS BETWEEN
C     THE COORDINATE SYSTEMS AND IS EXPLAINED IN THE CROSEC COMMENTS.
C
C
C     IF ALL 4 OF THE HOPE PLANE CONSTANTS ARE ZERO,
C     NO CROSS SECTIONAL SOULTION IS POSSIBLE,
C
C
      IF (A.EQ.0.AND.B.EQ.0.AND.C.EQ.0.AND.D.EQ.0) 10,20
   10 PRINT 200
      STOP
C
C
C     IF ALL 4 OF THE HOPE PLANE CONSTANTS ARE NON-ZERO,
C      OR IF PARALLEL TO EITHER THE Y AXIS OR THE Z AXIS
C     THEN A GENERAL SOLUTION IS CALLED FOR SINCE NO HOPE PLANE
C     ORIENTATION WITH A MAJOR COORDINATE PLANE EXISTS.
C
   20 IF ((A.NE.0.AND.B.NE.0.AND.C.NE.0.AND.D.NE.0).OR.(A.NE.0.AND.B.EQ.
     10.AND.C.NE.0.AND.D.NE.0).OR.(A.NE.0.AND.B.NE.0.AND.C.EQ.0.AND.D.NE
     2.0)) 30,40
   30 CALL GENSOL (D/A,0,0,A,B,C,D,XMAT9)
      RETURN
C
C
C     HOPE IS PARALLEL TO THE XY PLANE IF-
C
   40 IF (A.EQ.0.AND.B.EQ.0.AND.C.EQ.1) 50,60
C
C     HOPE PARALLEL TO XY PLANE
C
   50 CALL FILXMAT9 (1,0,0,0,0,0,1,0,0,0,0,0,1,0,-D*C,XMAT9)
      RETURN
C
C
C     HOPE IS PARALLEL TO THE XZ PLANE IF-
C
   60 IF (A.EQ.0.AND.B.EQ.1.AND.C.EQ.0) 70,80
C
C     HOPE PARALLEL TO XZ PLANE
   70 CALL FILXMAT9 (1,0,0,0,0,0,0,1,0,0,0,-1,0,0,-D*B,XMAT9)
```

```
      RETURN
C
C
C     HOPE IS PARALLEL TO THE YZ PLANE IF=
C
   80 IF (A.EQ.1.AND.B.EQ.0.AND.C.EQ.0) 90,100
C
C     HOPE PARALLEL TO YZ PLANE
C
   90 CALL FILXMAT9 (0,1,0,0,0,0,0,1,0,0,1,0,0,0,D*A,XMAT9)
      RETURN
C
C
C     HOPE PARALLEL TO X AXIS AND INTERCEPTS Y AND Z AXES
C
  100 IF (A.EQ.0.AND.B.NE.0.AND.C.NE.0) 110,120
C
  110 CALL FILXMAT9 (1,0,0,0,0,0,C,-B,0,A,B,C,-(D*B*B+D*C*C),XMAT9)
C
      RETURN
C
C
C     IF D IS ZERO THEN HOPE PLANE INTERSECTS THE ORIGIN
C
  120 IF ((D.EQ.0).AND.(B.NE.0)) 130,140
C
C     IN SUCH A SITUATION
C     FIRST TRY TO LET THE X PRIME AXIS BE
C     THE INTERSECTION BETWEEN THE HOPE PLANE
C     AND THE XY PLANE
C
  130 PHI=ATAN((-A)/(B))
      T11=COS(PHI)
      T21=SIN(PHI)
      T31=0
      GO TO 180
C
C     SECOND TRY THE INTERSECTION BETWEEN
C     THE HOPE PLANE AND THE XZ PLANE
C
  140 IF ((D.EQ.0).AND.(C.NE.0)) 150,160
  150 PHI=ATAN((-A)/(C))
      T11=COS(PHI)
      T21=0
      T31=SIN(PHI)
      GO TO 180
C
C     THIRD TRY THE INTERSECTION BETWEEN
C     THE HOPE PLANE AND THE YZ PLANE
C
  160 IF ((D.EQ..0).AND.(A.NE.0)) 170,190
  170 PHI=ATAN((-B)/(C))
      T11=0
      T21=SIN(PHI)
      T31=COS(PHI)
      GO TO 180
```

```
C
C     THE DIRECTION COSINES FOR THE Z AXIS ARE A,B,C
C
  180 T13=A
      T23=B
      T33=C
C
C     THE DIRECTION COSINES FOR THE Y PRIME AXIS
C     ARE THE CROSS PRODUCTS OF ZPRIME CROSS X PRIME
C
      T12=T23*T31-T21*T33
      T22=T11*T33-T13*T31
      T32=T13*T21-T11*T23
      CALL FILXMAT9 (T11,T21,T31,0,T12,T22,T32,0,T13,T23,T33,0,XMAT9)
      RETURN
C
C     IF THIS POINT IS REACHED, THEN NO HOPE COORDINATE SYSTEM IS
C     POSSIBLE, AN ERROR MESSAGE IS PRINTED
C     AND THE PROGRAM STOPPED,
C
  190 PRINT 210, A,B,C,D
      STOP
C
C
  200 FORMAT (1X,*ALL COEFFICIENTS FOR HOPE PLANE TEST ZERO*,/,*NO CROSS
     1 SECTION POSSIBLE, STOP CALLED FOR*)
  210 FORMAT (1X,*FAILED  11 TESTS IN TESTHOPE, DECLARED INVALID*,/,1X,*
     1A = *,F10.5,*, B= *,F10.5,*, C = *,F10.5,*, D = *F10.5,/)
      END
```

```
TESTHOPE                                          11/02/70        ED      0

                                 IDENT     TESTHOPE
PROGRAM LENGTH              00527
ENTRY POINTS    TESTHOPE    00056
BLOCK NAMES
                CROSSEC     03037
EXTERNAL SYMBOLS
                THEND,
                Q8QSTOPS
                Q8QDICT,
                GENSOL
                FILXMAT9
                SINF
                COSF
                ATANF
                STH,
                QNSINGL,
   00162 SYMBOLS
```

```
                                                     11/02/70
      SUBROUTINE FILXMAT9 (U1,U2,U3,U4,U5,U6,U7,U8,U9,U10,U11,U12,XMAT9)
C
C     THE PURPOSE OF THIS SUBROUTINE IS TO FILL UP THE XMAT9 ARRAY
C     WITH THE 12 INPUT PARAMETERS.
C
C
C     THE PURPOSE OF XMAT9 IS EXPLAINED IN CROSEC COMMENTS
C
C
C     THIS SUBROUTINE IS CALLED FROM SUBROUTINE TESTHOPE
C
C
      DIMENSION XMAT9(1)
      XMAT9(1)=U1
      XMAT9(2)=U2
      XMAT9(3)=U3
      XMAT9(4)=U4
      XMAT9(5)=U5
      XMAT9(6)=U6
      XMAT9(7)=U7
      XMAT9(8)=U8
      XMAT9(9)=U9
      XMAT9(10)=U10
      XMAT9(11)=U11
      XMAT9(12)=U12
      RETURN
      END

FILXMAT9                                             11/02/70        ED      0

                                  IDENT     FILXMAT9
PROGRAM LENGTH              00200
ENTRY POINTS     FILXMAT9   00003
EXTERNAL SYMBOLS
                 Q8QDICT.
   00073 SYMBOLS
```

11/02/70

```
      SUBROUTINE GENSOL (L1,L2,L3,A,B,C,D,XMAT9)
C
C     THE PURPOSE OF THIS SUBROUTINE IS TO COMPUTE
C     THE DIRECTION COSINES FOR THE GENERAL CASE WHERE THE HOPE PLANE
C     IS NOT PARALLEL TO PLANES XY, XZ, OR YZ,
C
C     (L1,L2,L3) IS THE INTERSECTION OF THE X-AXIS AND THE HOPE PLANE
C
      REAL L1,L2,L3
C
C     A,B,C,D ARE THE HOPE PLANE CONSTANTS
C
C     THE XMAT9 ARRAY IS FOR OUTPUT AND IS DESCRIBED IN CROSEC COMMENTS
C
      DIMENSION XMAT9(1)
C
C     THIS SUBROUTINE IS CALLED FROM SUBROUTINE TESTHOPE
C
C     THE COORDINATES OF THE POINT FORMED BY THE INTERSECTION OF THE
C     NORMAL TO THE PLANE AND THE PLANE ARE (DA,DB,DC)
C
      DA=D*A
      DB=D*B
      DC=D*C
C
C     THE DELTA X, DELTA Y, AND DELTA Z DISTANCES BETWEEN THE TWO POINTS
C
      XX=L1-DA
      XY=L2-DB
      XZ=L3-DC
C
C     THE X≠ AXIS IS DIRECTED FROM (DA,DB,DC) TO (L1,L2,L3)
C
      DEN=SQRT(XX**2+XY**2+XZ**2)
      XAX=XX/DEN
      XAY=XY/DEN
      XAZ=XZ/DEN
C
C
C     THE Z≠ AXIS HAS DIRECTION COSINES A,B,C
C
C     THE Y≠ AXIS HAS ITS DIRECTION COSINES COMPUTED
C     FROM THE CROSS PRODUCT TERMS OF Z≠ CROSS X≠  ,
C
      YAX=B*XAZ-C*XAY
      YAY=C*XAX-A*XAZ
      YAZ=A*XAY-B*XAX
C
C     THE XMAT9 ARRAY IS FILLED WITH POSITIONS 4, 8, AND 12  BEING
C     FILLED BY THE COMPUTED CONSTANTS C1, C2, C3 AS DEFINED IN THE
C     CROSEC COMMENTS,
C
      XMAT9(1)=XAX
      XMAT9(2)=XAY
      XMAT9(3)=XAZ
      XMAT9(4)=XAX*(-DA)+XAY*(-DB)+XAZ*(-DC)
```

11/02/70

```
      XMAT9(5)=YAX                                                 GEN 570
      XMAT9(6)=YAY                                                 GEN 580
      XMAT9(7)=YAZ                                                 GEN 590
      XMAT9(8)=YAX*(-DA)+YAY*(-DB)+YAZ*(-DC)                       GEN 600
      XMAT9(9)=A                                                   GEN 610
      XMAT9(10)=B                                                  GEN 620
      XMAT9(11)=C                                                  GEN 630
      XMAT9(12)=A*(-DA)+B*(-DB)+C*(-DC)                            GEN 640
      RETURN                                                       GEN 650
      END                                                          GEN 660.
```

```
 GENSOL                                           11/02/70        ED      (

                              IDENT     GENSOL
PROGRAM LENGTH            00323
ENTRY POINTS    GENSOL    00003
FXTERNAL SYMBOLS
                Q8QDICT,
                SQRTF
   00111 SYMBOLS
```

11/02/70

```
      FUNCTION ISITOK (XR,YR,ZR,XI,YI,Z,RAWXMIN,RAWYMIN,RAWZMIN,RAWXMAX,
     1RAWYMAX,RAWZMAX,YES,NO,IONCE,KR,IP,SCALE)
      DIMENSION KR(1)
C
C
C     THE PURPOSE OF THIS  FUNCTION  IS TO PERFORM
C     TWO ACCEPTANCE TESTS
C     ON THE NEWLY COMPUTED POINT OF INTERSECTION AND TO PROVIDE A YES
C     OR NO ANSWER TO ITS ACCEPTABLENESS
C
C     (XR,YR,ZR) IS THE POINT IN THE X,Y,Z SYSTEM
C
C     (X,Y,Z)    IS THE POINT IN PLOT COORDINATES (Z INACTIVE)
C
C     RAWXMIN,... ARE THE X,Y,Z SYSTEM LIMITS
C
C     YES, NO     ARE THE TWO POSSIBLE ANSWERS
C
C     THIS FUNCTION IS CALLED FROM THE INNER LOOP IN CROSEC
C
      REAL KR
C
C     DOES THE POINT FALL WITHIN THE X,Y,Z LIMITS?
C
      IF (XR.LT.RAWXMIN.OR.XR.GT.RAWXMAX.OR.YR.LT.RAWYMIN.OR.YR.GT.RAWYM
     1AX.OR.ZR.LT.RAWZMIN.OR.ZR.GT.RAWZMAX) 30,10
C
C     ARE THE PLOT COORDINATES BETWEEN 0 AND 10.0?
C
   10 IF (XI.LT.0.0.OR.XI.GT.10.0.OR.YI.LT.0.0.OR.YI.GT.10.0) 30,20
C
C     IF THE POINT HAS PASSED THE FIRST TWO TESTS ITS (X,Y) IS
C     ENTERED IN THE KR ARRAY USED BY SUBROUTINE DLINE.
   20 KR(IP)=(XI-5.0)/SCALE
      KR(IP+1)=(YI-5.0)/SCALE
      IP=IP+2
C
C     OPEN THE GATE THAT CONTROLS THE DLINE CALL
C
      IONCE=0
C
C
C     THE YES RETURN
C
      ISITOK=YES
      RETURN
C
C     THE  NO RETURN
C
   30 ISITOK=NO
C
C     ZERO THE POINT VARIABLES SINCE THE ANSWER IS NO.
C
      XR=0
      YR=0
      ZR=0
```

```
      XI=0                                                                    IOK 570
      YI=0                                                                    IOK 580
      Z=0                                                                     IOK 590
      RETURN                                                                  IOK 600
      END                                                                     IOK 610-
```

```
  ISITOK                                                  11/02/70        ED      0

                                IDENT     ISITOK
PROGRAM LENGTH          00310
ENTRY POINTS    ISITOK  00003
EXTERNAL SYMBOLS
                01010100
                Q8QDICT,
   00134 SYMBOLS
```

11/02/70

```
      SUBROUTINE DLINE (NAME1,IP,KR,SCALE)
      DIMENSION KR(1)
C
C     THE PURPOSE OF THIS SUBROUTINE IS TO PLOT A CONTINUOUS LINE
C     BETWEEN THE POINTS IN THE KR ARRAY
C     WITHOUT ALLOWING ANY OVERDRAWING
C     AND TO COMPUTE AND PRINT THE SLOPE AND INTERCEPT OF THIS LINE.
C
C     AN ORDERING IN X IS PERFORMED
C     TO FIND A MINIMUM AND MAXIMUM.
C     IF A VERTICAL LINE IS PRESENT, HOWEVER,
C     AN ORDERING IN Y IS NECESSARY.
C
C     NAME 1 IS THE NAME OF THE PLANE ISOLATED IN THE OUTER LOOP
C     AND IS THE PLANE WHOSE LINE OF INTERSECTION
C     IS ABOUT TO BE PLOTTED.
C
C     THE KR ARRAY CONTAINS CONSECUTIVE X,Y PLOT COORDINATES
C     ((IP-1)/2 IS THE NUMBER OF POINTS)
C
      REAL KR
      REAL MINX,MAXX,SLOPE
      REAL MINY,MAXY
      S(W)=W*SCALE+5.0
C
C     ESTABLISH THE END COUNT FOR LOOP CONTROL
      IPEND=IP-1
C
C     THE FIRST POINT.
C
      X=KR(1)
      Y=KR(2)
C
C     IF THERE IS ONLY ONE POINT IN THE ARRAY A LINE CANNOT BE DRAWN
C
C     IF THERE ARE MORE THAN TWO POINTS A MINIMUM AND A MAXIMUM MUST
C     BE DETERMINED TO AVOID THE PEN DRAWING OVER A LINE
C     MORE THAN ONCE.
C
      IF (IP.EQ.3) 280,110
C
C     THERE ARE TWO OR MORE POINTS
C
C     INITIALLY THE FIRST POINT IN X IS BOTH THE MINIMUM AND MAXIMUM
  110 MINX=X
      YOFMINX=Y
      MAXX=X
      YOFMAXX=Y
C
      DO 150 I=3,IPEND,2
C     A NEW SET OF VARIABLES
C
      X=KR(I)
      Y=KR(I+1)
C
C     THE TEST FOR MINIMUM
```

```
C
      IF (X.LT.MINX) 130,120
C
C     THE TEST FOR MAXIMUM
C
  120 IF (X.GT.MAXX) 140,150
C
C     A NEW MINIMUM
  130 MINX=X
      YOFMINX=Y
      GO TO 150
C
C     A NEW MAXIMUM
C
  140 MAXX=X
      YOFMAXX=Y
      GO TO 150
  150 CONTINUE
C
C
C     THE SLOPE AND INTERCEPT OF THE LINE ARE TO BE CALCULATED
C     AND PRINTED OUT ALONG WITH THE NAME OF THE PLANE FOR
C     PURPOSES OF INFORMATION AND AS AN AID TO PLOT INTERPRETATION
C     THE EXAMINATION OF DELX AND DELY FOR ZERO IS TO AVOID AN ATTEMPT
C     TO DIVIDE BY ZERO WHEN COMPUTING THE SLOPE
C
C     IF DELX IS ZERO THAN AN ORDERING OF Y IS DONE
C     ON THE ASSUMPTION OF A VERTICAL LINE BEING PRESENT
C
C     IT IS ASSUMED THAT TWO EQUATIONS OF THE FORM Y=MX+B EXIST
C     WITH THE TWO POINTS THAT HAVE BEEN ISOLATED,
C          (MINX, Y OF MIN X), (MAX X, Y OF MAX X),
C     AND THE FOLLOWING COMPUTATIONS REPRESENT A SOLUTION FOR M AND B
C     UNDER VARIOUS CONDITIONS,
C     I.E.
C          YOFMAXX=SLOPE*MAXX+B
C          YOFMINX=SLOPE*MINX+B
C     PLEASE NOTE...
C     WE LET M BE REPRESENTED BY THE WORD SLOPE
C     TO AVOID CONFUSION WITH AN INDEX REGISTER
C     THEN,
C          SLOPE=(YOFMAXX-YOFMINX)/(MAXX-MINX)
C     OR      SLOPE=DELY/DELX
C     AND OF COURSE, B=YOFMAXX-SLOPE*MAXX
C
C
      DELX=MAXX-MINX
      DELY=YOFMAXX-YOFMINX
C
C     A BRANCH TO STATMENT 180 IS A VERTICAL SOLUTION
C
C     A BRANCH TO STATEMENT 240 IS A HORIZONTAL SOLUTION
C
C     A BRANCH TO STATEMENT 170 IS A NORMAL (Y=SLOPE*X+B) SOLUTION
C
      IF (ABSF(DELX).LT.0.00001) 180,160
```

11/02/70

```
  160 IF (ABSF(DELY).LT.0.00001) 240,170
C
C     NORMAL SOLUTION
C
  170 TA=DELY/DELX
      SLOPE=(ATANF(TA))*57.32
      B=YOFMAXX-TA*MAXX
      PRINT 290, NAME1,SLOPE,B
      GO TO 260
C
C     SINCE DELX IS ZERO IT IS NECESSARY TO TRY ORDERING IN Y
C
C     INITIALLY THE FIRST POINT IS BOTH THE MINIMUM AND MAXIMUM
C
C
C     VERTICAL SOLUTION
C
  180 X=KR(1)
      Y=KR(2)
      XOFMINY=X
      MINY=Y
      XOFMAXY=X
      MAXY=Y
      DO 220 I=3,IPEND,2
C     A NEW SET OF VARIABLES
      X=KR(I)
      Y=KR(I+1)
C     THE TEST FOR MINIMUM Y
      IF (Y.LT.MINY) 200,190
C     THE TEST FOR MAXIMUM Y
  190 IF (Y.GT.MAXY) 210,220
C     A NEW MINIMUM Y
  200 XOFMINY=X
      MINY=Y
      GO TO 220
C     A NEW MAXIMUM
  210 XOFMAXY=X
      MAXY=Y
      GO TO 220
  220 CONTINUE
C     CHECK FOR NO SOLUTION
      IF (MINY.NE.MAXY) 230,250
  230 CONTINUE
      B=XOFMAXY
      PRINT 300, NAME1,B
C
C     SETTING UP THE PLOT ON ORDERED Y VALUES
C
      FIRSTX=S(XOFMINY)
      FIRSTY=S(MINY)
      SECONDX=S(XOFMAXY)
      SECONDY=S(MAXY)
      GO TO 270
C
C     HORIZONTAL LINE SOLUTION
C
```

```
  240 SLOPE=0
      B=YOFMAXX
      PRINT 310, NAME1,B
      GO TO 260
  250 CONTINUE
C
C     NO SOLUTION
C
      PRINT 320, NAME1
      GO TO 280
  260 CONTINUE
C     SETTING UP THE PLOT ON ORDERED X VALUES
      FIRSTX=S(MINX)
      FIRSTY=S(YOFMINX)
      SECONDX=S(MAXX)
      SECONDY=S(YOFMAXX)
      GO TO 270
  270 CONTINUE
C
C
C     DRAW A LINE FROM THE MINIMUM TO THE MAXIMUM
C
C
C     MOVE WITH THE PEN UP TO THE MINIMUM
C
      CALL PLOT (FIRSTX,FIRSTY,3)
C
C     LOWER THE PEN
C
      CALL PLOT (FIRSTX,FIRSTY,2)
C
C     MOVE TO THE MAXIMUM POINT WITH THE PEN DOWN
C
      CALL PLOT (SECONDX,SECONDY,2)
C
C     RAISE THE PEN AT THE MAXIMUM POINT
C
      CALL PLOT (SECONDX,SECONDY,3)
C
  280 CONTINUE
C
C     INITIALIZE THE KR ARRAY COUNTER
C
      IP=1
C
C                                                   AN EXTRA LINE SPACE
C
      PRINT 330
C
  290 FORMAT (/,/,1X,*LINE OF INTERSECTION FOR PLANE *,A8,/,1X,*HAS A SL
     1OPE = *,F7.1,*  DEGREES*,2X,*AND A Y INTERCEPT = *,F6,2)
  300 FORMAT (1X,*LINE OF INTERSECTION FOR PLANE *,A8,/1X,*IS VERTICAL,I
     1,E,SLOPE = 90 DEGREES*,/1X,*AND INTERSECTS THE X PRIME AXIS AT *,F
     26,2)
  310 FORMAT (1X,*LINE OF INTERSECTION FOR PLANE *,A8,/,1X,*IS PARALLEL
     1TO THE X PRIME AXIS AND *,/,1X,*INTERSECTS THE Y PRIME AXIS AT *,F
```

11/02/70

```
   26,2)                                                                DLN2170
 320 FORMAT (1X,*FOR *,A8,* PLANE BOTH DELX AND DELY ARF ZERO*,/,1X,*THDLN2180
   1EREFORE NO LINEAR FORM OF LINE IS POSSIBLE*)                       DLN2190
 330 FORMAT (1X,/)                                                      DLN2200
     END                                                                DLN2210
```

```
  DLINE                                                  11/02/70       ED

                                        IDENT      DLINE
PROGRAM LENGTH                 00620
ENTRY POINTS       DLINE       00151
EXTERNAL SYMBOLS
                   THEND,
                   Q8QDICT,
                   PLOT
                   ATANF
                   STH,
                   QNSINGL,
  00146 SYMBOLS
```

11/09/70

```
      PROGRAM CYLINDER                                                  CYL   10
      COMMON /TIMETEST/ KSETIME1,KSETIME2,KSETIME,KSETADR               CYL   20
C                                                                       CYL   30
C     ...     COMMON AREA FOR CDC APT 3 SECTION 0                       CYL   40
C                                                                       CYL   50
C     *          ******          ******          ******          ****** CYL   60
      COMMON /SYSTEMZ/ SYSTEM(4),KAPTCN,KAPTTR,KAPTIO,KAPTID,KFLAGS(10),CYL   70
     1K0,K1,K2,K3,K4,K5,K6,K7,K8,K9,IFILL1,IFILL2,IFILL3,IFILL4,KFLAG0,KCYL   80
     2FLAG1,KFLAG2,KFLAG3,KFLAG4,IFILL5,IWAVEN,IPTNLY,NOPOST,IFILL6,KAUTCYL   90
     3OP,ICLPRT,INDEXX,IPLOTR,IFILL7,NOPLOT,KDYNFG,LOCJPT,LOCBEG,KSECIN,CYL  100
     4NCLREC,LOCMAC,KPOCKET,IFILL8,IFILL9,IFILL10,IPOSTP(1),NUMPST,IPOSTCYL  110
     5FL(18),TAPETB(1),CANTAP,CLTAPE,POCTAP,PLOTAP,SRFTAP,LIBTAP,CRDTAP,CYL  120
     6IFILL11,CORTAP,TAPES1,TAPES2,TAPES3,TAPES4,FORTIN,INTAPE,IOUTAP,PUCYL  130
     7NTAP,LSTFLG,LTVFLG,KONVTCL,KINTRUPT,PI,PI02,DGTRD,RDTDG,ONE,EXTRA0CYL  140
     8(20)                                                              CYL  150
      EQUIVALENCE (PROTAP,TAPETB)                                       CYL  160
C     *          ******          ******          ******          ****** CYL  170
C                                                                       CYL  180
C     ...     COMMON AREA FOR CDC APT 3 SECTION 1                       CYL  190
C                                                                       CYL  200
C     *          ******          ******          ******          ****** CYL  210
      COMMON /SECT1LOG/ ITAB1,ITAB2,ITAB3,ITAB4,ITAB5,ISNAM,ITAB11,ITAB1CYL  220
     12,ITAB13,JENDPTPP,JENDCAN,JENDSTOR,JSTRTCAN,JENDSYM,JCANTEMP,JRPTACYL  230
     2R,JLPTAB,MAXNST,JINWD,JCHAR,IWDERR,JBUFL,NUPERP,NUPUN,JSTYPE,JVARSCYL  240
     32,SCHERR,NMACV,MACASN(25),INDXPT,IPTP,IXPT,MODE,EOCFLG,LPNDFL,TRMFCYL  250
     4LG,INTRUPT,JUMPFL,ICDERR,DEBUG,MACMODE,NESTFL,NRESULT,IPTLIM,JEXECCYL  260
     5,KTYPE,MACTYP,IPARTERR,FINIS,IOFLG,MACDEL,JSUBER,NUMBERR,DEFSTO(85CYL  270
     6),DEFTAB(1000),ZSUR(30),XMAT4(16),XMAT3(16),XMAT2(16),XMAT1(16),TMCYL  280
     7ATX(16),ISTDMODE,ISTDLIT,ISTDTBL,ISTDINDX,ISTDTYPE,ISTDWD,JPTIND,KCYL  290
     8PTCODE,KPTNAME,KPTTYPE,KPTNUM,KPTINDX,KPTSUB,KOMFLG,KOMPOP,NOSUBS,CYL  300
     9KANFLG,KREFSYS,KANREC,KANCNT,INAME,KANSURF,KANINDX,JPRELEN,NEWCARD,CYL  310
     $JGORIT,NUMSTID,NUMCSEQ,IRECIX,IRECNO,JTLPOS,ITITLE(9),LSRECN,NNODECYL  320
     $FX,NNODEFI,NIDUM,ISLASH,IEQUAL,IBLANK,                            CYL  330
     T    IDUMMY,    N10000,    N7777,     MASKU,     MASKL,     IDIV,  CYL  331
     U    MACREL,    MACLOC,    MACBEGN,   MACLAST,   MACLEVEL,         CYL  332
     V    MACNAME(3), MACINDX(3), NMV,     JRESTOR,   MACPSH(3,25),     CYL  333
     W    JTEMP1,    JTEMP2,    JTEMP3,    JTEMP4,    JTEMP5,    JTEMP6, CYL  334
     X    JTEMP7,    JTEMP8,    JTEMP9,    EXTRA1(20)                     CYL  335
      EQUIVALENCE (DEFANS(1),IDEFSTO(4),DEFSTO(4)), (LSTYPE,KTYPE), (PTNCYL  340
     1UM,KPTNUM)                                                        CYL  350
      DIMENSION IDEFSTO(85), DEFANS(26), IDEFTAB(1000), ILPTAB(200), IRPCYL  360
     1TAB(200), ITNTAB(200), JPROTP(100), IRECSV(200), MACVAR(25), MACNOCYL  370
     2R(25), INWORD(10), IBUUP(2), PISTO(6), IPISTO(6), IDREC(4)        CYL  380
      EQUIVALENCE (INWORD(14),IBUUP(16),ILPTAB(990),IRPTAB(790),ITNTAB(5CYL  390
     190),PISTO(390),IPISTO(390),IDREC(384),JPROTP(380),IRECSV(280),MACVCYL  400
     2AR(75),MACNOR(50),IDEFTAB(1000),DEFTAB(1000))                     CYL  410
C                                                                       CYL  420
      COMMON /VOCABTBL/ KOM(1100)                                       CYL  430
C                                                                       CYL  440
      COMMON /2/ JTABNUM,JTABL(12120)                                   CYL  450
C     *          ******          ******          ***4***           **** CYL  460
C                                                                       CYL  470
C     CONSTANTS CARRIED OVER FROM CROSEC.                               CYL  480
C     SEE PROGRAM CROSEC FOR DETAILS AND CARD ≠≠CYL 820≠≠ BELOW.        CYL  490
C                                                                       CYL  500
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,UCYL  510
```

11/09/70

```
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,CYL 520
     2MINOR,PHI,CX,CY,RADILS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,ILCYL 530
     3OOP,IP,KR(200),IONCE                                             CYL 540
      EQUIVALENCE (INCHOPE,IFILL8)                                     CYL 550
      EQUIVALENCE (PLOTPLNE,IFILL9)                                    CYL 560
C                                                                      CYL 570
C                                                                      CYL 580
C     THE PURPOSE OF PROGRAM CYLINDER IS TO PROVIDE                    CYL 590
C     THE ENTRANCE POINT FOR SEGMENT 102 AND TO CONTROL ITS ACTIONS.   CYL 600
C                                                                      CYL 610
C                                                                      CYL 620
C     THE MAIN ACTIONS TAKEN ARE TO-                                   CYL 630
C                                                                      CYL 640
C     (1) FETCH AND IDENTIFY  CYLINDER/CIRCLE AND SPHERE FORMS         CYL 650
C     IN THE DEFINED SYMBOL TABLE.                                     CYL 660
C                                                                      CYL 670
C     (2) OBTAIN THE CANONICAL CONSTANTS                               CYL 680
C     FOR THE PROPERLY IDENTIFIED FORMS,                               CYL 681
C                                                                      CYL 690
C     (3) CONTROL INTERSECTION PROCESSING THRU THE 4 SUBROUTINES,      CYL 700
C     ALGOR A, ALGOR B, ALGORC AND ALGOR D.                            CYL 710
C                                                                      CYL 720
C     (4) PRINT OUT THE DESCRIPTION OF THE CIRCLE OR ELLIPSE,          CYL 721
C     AS THE CASE MAY BE, AND                                          CYL 730
C                                                                      CYL 740
C     (5) CALL IN SEGMENT 103 IF MORE PROCESSING IS NEEDED, OR         CYL 750
C     RECYCLE ITSELF,OR                                                CYL 760
C     OR GO TO ISHOPE FOR ANOTHER HOPE PLANE,OR                        CYL 770
C     PREPARE FOR CROSEC TERMINATION IF ALL POSSIBLE                   CYL 780
C     CYLINDERS/CIRCLES AND SHPERES HAVE BEEN PROCESSED AND            CYL 790
C     NO MORE HOPE PLANES ARE PRESENT.                                 CYL 800
C                                                                      CYL 810
C                                                                      CYL 820
C     COMMENTS ON THOSE PORTIONS OF CROSSEC COMMON                     CYL 830
C     PERTINENT TO SEGMENT 102                                         CYL 840
C                                                                      CYL 850
C                                                                      CYL 860
C     THE CANONICAL FORM FOR THE CYLINDER CONSISTS OF A                CYL 870
C        1) A POINT IN SPACE,                                          CYL 880
C        2) A VECTOR, AND                                              CYL 890
C        3) A RADIUS.                                                  CYL 900
C                                                                      CYL 910
C                                                                      CYL 920
C     LET THE POINT BE DESIGNATED BY XC, YC, ZC,                       CYL 930
C         THE DIRECTION  COSINES OF THE VECTOR BY UX, UY, UZ, AND      CYL 940
C         THE RADIUS BY R,                                             CYL 950
C                                                                      CYL 960
C     THE DCOSCYL ARRAY HOLDS THE DIRECTION COSINES                    CYL 970
C     OF THE LOCAL CYLINDER COORDINATE SYSTEM                          CYL 980
C                                                                      CYL1000
C      THE CXYZ ARRAY HOLDS THE COEFFICIENTS OF THE  CYLINDER EQUATION CYL1010
C      IN THE X,Y,Z  SYSTEM                                            CYL1020
C                                                                      CYL1030
C     THE EXYZ ARRAY HOLDS THE COEFFICIENTS                            CYL1040
C     OF THE CURVE OF INTERSECTION                                     CLY1041
C     EXPRESSED IN THE XYZ COORDINATE SYSTEM                           CYL1050
```

```
C
C     THE EH ARRAY HOLDS THE COEFFICIENTS OF THE CURVE OF INTERSECTION
C     EXPRESSED IN THE HOPE PLANE COORDINATE SYSTEM
C
C     THE HC ARRAY HOLDS THE COEFFICIENTS FOR THE CURVE INTERSECTION
C     IN THE HOPE PLANE WHEN IT IS REDUCED TO TWO COORDINATES,
C     XPRIME AND YPRIME.
C
C     THE X,Y ARRAYS ARE USED TO BUILD UP
C     ONE QUADRANT SET OF POINTS FOR EITHER
C     THE ELLIPSE OR THE CIRCLE IN BUILDING
C     THE DISCRETE SET OF POINTS.
C
C     ≠≠MAJOR≠≠ AND ≠≠MINOR≠≠ ARE THE SEMI-AXES OF THE ELLIPSE.
C
C     CX AND CY ARE THE HOPE PLANE COORDINATES OF THE
C     CENTER OF THE ELLIPSE OR CIRCLE OF INTERSECTION.
C
C     ≠≠RADIUS≠≠ IS THE RADIUS OF A CIRCLE OF INTERSECTION.
C
C     ≠≠ISET≠≠, IS THE NUMBER OF COMPUTED DISCRETE POINTS FOR THE
C     ELLIPSE OR CIRCLE (200 FOR A CIRCLE AND 400 FOR AN ELLIPSE).
C
C     ≠≠TL≠≠, IS THE LENGTH OF THE FINITE CYLINDER.
C     UNTIL REFINED BY FUTURE CHANGES ALL CYLINDERS ARE CONSIDERED TO
C     BE 5 INCHES LONG.
C
C     ≠≠ILOOP≠≠, USED IN CONNECTION WITH THE CYLINDER/SPHERE SEARCH
C     LOOP BETWEEN SEGMENTS 102 AND 103.
C
C     ≠≠IONCE≠≠, FLAG USED TO INDICATE A SINGLE HOPE PLANE OR THE
C     LAST HOPE PLANE OF A SEQUENCE.
C
C     THE XS,YS ARRAYS HOLDS THE POINTS OF THE PERIMETER OF THE CURVE
C     OF INTERSECTION IN THE HOPE PLANE COORDINATE SYSTEM
C
C
C     THE SEMI-AXES OF THE ELLIPSE
C
      REAL MAJOR,MINOR
C
C
C     SEARCHING THE DEFINED SYMBOL TABLE FOR A CYLINDER
C     SEE CROSEC COMMENTS FOR DETAILS
C
      TL=5
   10 I=ILOOP
      ISTDTBL=0
      ISAM=0
      J=ITAB11+I-1
      IONE=JTABL(J)
      IX=(IONE.AND.77000000000000008)
C
C     REJECTION IF CONTAINS ≠≠X≠≠ PREFIX.
C
      IF (IX.EQ.8HX0000000) 230,20
```

11/09/70

```
   20 CONTINUE                                                            CYL1620
      ITWO=JTABL(J+1)                                                     CYL1630
      IF (ITWO.EQ.NNODEFX.OR.ITWO.EQ.NNODEFI) 230,30                      CYL1640
   30 ISTDWD=ITWO                                                         CYL1650
      CALL STDUNPK                                                        CYL1660
      IF (ISTDTBL.EQ.4) 40,230                                            CYL1670
   40 KANSLRF=ITWO                                                        CYL1680
      CALL CANGET                                                         CYL1690
      ISAM=(DEFSTO(1).AND.7777B)                                          CYL1700
      IF ((ISAM.EQ.4).OR.(ISAM.EQ.5)) 60,50                               CYL1710
   50 IF (ISAM.EQ.13) 100,230                                             CYL1720
   60 NAMECYL=IONE                                                        CYL1730
      WHICH=8HCYLINDER                                                    CYL1740
C                                                                         CYL1750
C     CYLINDER VALUES PASSED THRU S32T048 TO AVOID -1 TROUBLES.           CYL1760
C                                                                         CYL1770
      XC=S32T048(DEFSTO(4))                                               CYL1780
      YC=S32T048(DEFSTO(5))                                               CYL1790
      ZC=S32T048(DEFSTO(6))                                               CYL1800
      UX=S32T048(DEFSTO(7))                                               CYL1810
      UY=S32T048(DEFSTO(8))                                               CYL1820
      UZ=S32T048(DEFSTO(9))                                               CYL1830
      R=S32T048(DEFSTO(10))                                               CYL1840
C                                                                         CYL1850
C                                                                         CYL1860
C     IF THE SUM OF SQUARES OF THE VECTOR≠S COMPONENTS IS NOT WITHIN      CYL1870
C     0.0001 OF UNITY THEN FURTHER CYLINDER PROCESSING FOR CROSEC IS      CYL1880
C     NOT POSSIBLE.                                                       CLY1881
C                                                                         CYL1900
C                                                                         CYL1910
      IF (ABS(1-(UX**2+UY**2+UZ**2)).LE.0.0001) 80,70                     CYL1920
   70 PRINT 270,NAMECYL,UX,UY,UZ                                          CYL1930
       GO TO 230                                                          CYL1931
   80 CONTINUE                                                            CYL1940
C                                                                         CYL1950
C                                                                         CYL1960
C     THE PARAMETERS IN THE CALL TO TESTCYL ARE THE SEVEN ELEMENTS OF     CYL1970
C     CYLINDER≠S CANONICAL FORM AND THE NAME OF THE ARRAY                 CYL1980
C       THAT IS TO BE FILLED                                              CYL1990
C     WITH THE DIRECTION COSINES OF THE CYLINDER≠S COORDINATE SYSTEM.     CYL2000
C                                                                         CYL2010
C                                                                         CYL2020
      IANSWER=999                                                         CYL2030
      CALL TESTCYL (XC,YC,ZC,UX,UY,UZ,R,DCOSCYL,IANSWER)                  CYL2040
C                                                                         CYL2050
C                                                                         CYL2060
C     THE QUESTION IS ASKED                                               CYL2070
C          ≠≠WAS A CYLINDER SET OF COORDINATES ESTABLISHED?≠≠             CYL2080
C                                                                         CYL2090
C       IANSWER=0 MEANS YES,IANSWER=1 MEANS NO                            CYL2100
C                                                                         CYL2110
C                                                                         CYL2120
      IF (IANSWER.EQ.1) 230,90                                            CYL2130
C                                                                         CYL2140
   90 GO TO 160                                                           CYL2150
  100 NAMESPH=IONE                                                        CYL2160
```

```
      WHICH=8HSPHERE
      XC=S32T048(DEFST0(4))
      YC=S32T048(DEFST0(5))
      ZC=S32T048(DEFST0(6))
      R=S32T048(DEFST0(7))
C     THE DISTANCE FROM THE CENTER OF THE SPHERE TO THE HOPE PLANE
C     IS THIS DISTANCE IS GREATER THAN R, THE HOPE PLANE DOES
C     NOT INTERSECT THE SPHERE
C         IF THIS DISTANCE IS EQUAL TO R, THE INTERSECTION IS A POINT
C     IF DS IS LESS THAN R, GO AHEAD A CIRCLE IS PRESENT
      DS=ABSF(((XC*A)+(YC*B)+(ZC*C)-D)/SQRTF(A**2+B**2+C**2))
      SPHXS=DS-R
      IF (ABSF(SPHXS).LT.0.0001) 130,110
  110 IF (SPHXS.LT.0) 150,120
  120 IF (SPHXS.GT.0) 140,230
  130 PRINT 280, NAMESPH
C     DS HAS DIRECTION COSINES A,B,C
C     BECAUSE IT IS NORMAL TO THE HOPE PLANE
C     THE COMPONENTS OF DS IN TERMS OF THE XYZ SYSTEM
C     ARE THEREFORE DS*A,DS*B,AND DS*C
C     THE TANGENT POINT (XT,YT,ZT) IN THE XYZ SYSTEM
      XT=XC+DS*A
      YT=YC+DS*B
      ZT=ZC+DS*C
C     THE TANGENT POINT CONVERTED TO THE HOPE SYSTEM
      CALL MM (XP,YP,ZP,XT,YT,ZT,XMAT9)
      X=XP*SCALE+5.0
      Y=YP*SCALE+5.0
      Z=ZP
C     THE TANGENT POINT PLOTTED AS TRIANGLE SYMBOL
      CALL SYMBOL (X,Y,0.20,2,0,-1)
      GO TO 230
  140 PRINT 290, NAMESPH
      GO TO 230
  150 CONTINUE
      CALL FDCOSCYL (1.0,0.0,0.0,1.0,0.0,0.0,1.0,DCOSCYL)
  160 CALL ALGORA (WHICH)
C
C     CREATE A SET OF COEFFICIENTS OF THE CURVE
C     OF INTERSECTION BY TRANSFORMING
C     THE CYLINDER EQUATION INTO THE XYZ SYSTEM AND AND SOLVING IT WITH
C     HOPE PLANE COEFFICIENTS
C
      CALL ALGORB
C
C     THE CURVE OF INTERSECTION IS TRANSFORMED
C     INTO THE HOPE PLANE SYSTEM.
C
      CALL ALGORC
C
C     THE CURVE OF INTERSECTION IS IDENTIFIED AS TO TYPE.
C
      CX=0
      CY=0
      MAJOR=0
      MINOR=0
```

11/09/70

```
      RADILS=0                                                          CYL2700
      PHI=0                                                             CYL2710
      IOKTEST=0                                                         CYL2720
      JSUBER=0                                                          CYL2730
      CALL ALGORD (JSUBER,NAMECYL)                                      CYL2740
      IF (IOKTEST.EQ.0) 170,230                                         CYL2750
  170 CONTINUE                                                          CYL2760
C                                                                       CYL2770
C     COMPUTE THE ECCENTRICITY                                          CYL2780
C                                                                       CYL2790
      EC=SQRTF(1-((MINOR**2)/(MAJOR**2)))                               CYL2800
      DPHI=PHI*RDTDG                                                    CYL2810
      IF (RADIUS.EQ.99999) 180,190                                      CYL2820
C                                                                       CYL2830
C     PRINT OUT INFO ON EITHER ELLIPSE OR CIRCLE                        CYL2840
C                                                                       CYL2850
  180 PRINT 300, NAMECYL,CX,CY,MAJOR,MINOR,EC,DPHI                      CYL2860
      GO TO 220                                                         CYL2870
  190 CONTINUE                                                          CYL2880
      IF ((WHICH.EQ.8HCYLINDER).AND.(MAJOR.EQ.99999)) 200,210           CYL2890
  200 PRINT 310, NAMECYL,CX,CY,RADIUS                                   CYL2900
      GO TO 220                                                         CYL2910
  210 PRINT 310, NAMESPH,CX,CY,RADIUS                                   CYL2920
      GO TO 220                                                         CYL2930
  220 CONTINUE                                                          CYL2940
C                                                                       CYL2941
C     THE CURRENT CYLINDER HAS BEEN PROCESSED THRU THE                  CYL2950
C     DETERMINATION OF ITS CURVE OF INTERSECTION                        CYL2960
C     (EITHER CIRCLE OR ELLIPSE)                                        CYL2970
C     NOW CALL SEGMENT 3 AND PROCESS FURTHER FOR                        CYL2980
C     DISCRETE POINT DETERMINATIONS                                     CYL2990
      IFILL7=103                                                        CYL3000
      CALL PRCNTL                                                       CYL3010
C                                                                       CYL3020
C     END OF CYLINDER SEARCH LOOP                                       CYL3030
C                                                                       CYL3040
C     TURN OFF THE PLOTTER                                              CYL3050
C                                                                       CYL3060
      GO TO 230                                                         CYL3070
  230 CONTINUE                                                          CYL3080
      ILOOP=ILOOP+1                                                     CYL3090
      IF (ILOOP.EQ.IDSEND) 240,10                                       CYL3100
  240 CONTINUE                                                          CYL3110
C     CYLINDER/SPHERE PROCESSING TO BE DISCONTINUED                     CYL3120
C     IS THERE ONLY ONE HOPE PLANE                                      CYL3130
      IF (INCHOPE.EQ.4HONCE) 250,260                                    CYL3140
  250 CONTINUE                                                          CYL3150
C     THE CYLINDER AND SPHERE SEARCH IS COMPLETE                        CYL3160
C     TERMINATE CROSS SECTIONAL PLOTTING                                CYL3170
C     AND GO ON TO OVERLAY 2                                            CYL3180
      IFILL7=9999                                                       CYL3190
      CALL PLOT (0,0,3)                                                 CYL3200
      CALL PLOTS (0,0)                                                  CYL3210
      IDSEND=IDSEND-1                                                   CYL3220
      CALL PRCNTL                                                       CYL3230
C     IF NOT, SEARCH FOR ANOTHER.                                       CYL3240
```

UNCLASSIFIED

11/09/70

```
  260 CONTINUE
      CALL PLOT (0,0,3)
      CALL PLOTS (0,0)
      ILOOP=0
      CALL ISHOPE
      CALL PRCNTL
C
  270 FORMAT (1X,A8,* DOES NOT HAVE A VALID VECTOR*,1X,3(E27.20,1X))
  280 FORMAT (1X,A8,* TANGENT TO PRESENT HOPE PLANE*,/)
  290 FORMAT (1X,A8,* NOT INTERSECTED BY PRESENT HOPE PLANE*,/)
  300 FORMAT (1X,A8,*,ELLIPSE*,*, CENTER AT *,F7.2,*, *,F7.2,*, SEMI-MAJ
     1OR AXIS = *,F7.2,*, SEMI-MINOR AXIS = *,F7.2,*, ECC = *,F7.5,*, PH
     2I = *,F7.2,/)
  310 FORMAT (1X,A8,*CIRCULAR INTERSECTION WITH*,*, CENTER AT *,F10.2,*,
     1 *,F10.2,*, RADIUS = *,F10.2,/)
      END
```

```
CYLINDER                                                 11/09/70          ED      0

                                        IDENT     CYLINDER
PROGRAM LENGTH                 00750
ENTRY POINTS      CYLINDER     00136
BLOCK NAMES
                  TIMETEST     00004
                  SYSTEMZ      00175
                  SECT1LOG     02642
                  VOCABTBL     02114
                  2            27531
                  CROSSEC      03037
EXTERNAL SYMBOLS
                  Q8QENTRY
                  THEND.
                  Q1Q03100
                  Q8QDICT.
                  STDUNPK
                  CANGET
                  S32TO48
                  TESTCYL
                  MM
                  SYMBOL
                  FDCOSCYL
                  ALGORA
                  ALGORB
                  ALGORC
                  ALGORD
                  PRCNTL
                  PLOT
                  PLOTS
                  ISHOPE
                  SQRTF
                  STH.
                  QNSINGL.
   00553 SYMBOLS
```

```
      SUBROUTINE TESTCYL (XC,YC,ZC,UX,UY,UZ,R,DCOSCYL,IANSWER)
      DIMENSION DCOSCYL(1)
C
C
C     THE PURPOSE OF SUBROUTINE TESTCYL IS TO ESTABLISH A LOCAL
C     COORDINATE SYSTEM FOR A CYLINDER OR A CIRCLE.
C
C     IF IT IS DETERMINED IN STATMENTS NO. 110 THRU 160 THAT A
C     VECTOR IS PARALLEL TO EITHER THE + OR - DIRECTIONS OF ONE OF THE
C     AXES THEN A STRAIGHTFORWARD ASSIGNMENT OF DIRECTION COSINES CAN
C     BE MADE IN THE APPROPRIATE STATEMENT OF THE SERIES
C     170,180,190,200,210,220.
C
C
C     AFTER THIS, THERE ARE SIX POSSIBILITIES FOR A SKEWED VECTOR
C
C     THREE OF THESE ARE CALLED INTERSECTION SOLUTIONS
C     A SERIES OF TESTS FOR THEM BEGINS AT STATEMENT 230.
C
C     THE LAST THREE POSSIBILITIES ARE CALLED INTERCEPT SOLUTIONS
C     A SERIES OF TESTS FOR THEM BEGINS AT STATEMENT 290.
C
C     THE BASE PLANE
C
C
C     THE CYLINDER≠S BASE PLANE IS DEFINED AS THAT PLANE WHICH CONTAINS
C     THE POINT (XC,YC,ZC) AND HAS THE CYLINDER≠S VECTOR AS ITS NORMAL.
C
C     THE EQUATION FOR THE BASE PLANE IS
C
C         (UX)*X + (UY)*Y + (UZ)*Z = UX*XC + UY*YC + UZ*ZC
C
C      THE X AND Y AXIS FOR THE LOCAL CYLINDER COORDINATE
C      SYSTEM ARE ASSIGNED IN THE BASE PLANE.
C      THE Z AXIS IS THE VECTOR.
C
C     THE Y AXIS IS DEFINED AS Z CROSS X.
C
C
C     SOME GENERAL COMMENTS ON INTERSECTION SOLUTIONS
C
C
C     THE POINT FROM THE CYLINDER DEFINITION MUST BE ON AN AXIS.
C     THE LOCAL X AXIS TO BE CREATED IS THE INTERSECTION OF THE BASE
C     PLANE WITH ONE OF THE PRINCIPLE PLANES (X=0, Y=0, Z=0).
C     THIS INTERSECTION MUST BE PERPENDICULAR TO THE CYLINDER≠S AXIS.
C     AND THE COMPUTATION OF ITS DIRECTION COSINES MAKES USE OF THIS
C     FACT, SINCE A NEGATIVE OF THE QUOTIENT OF TWO AXIS VECTOR
C     COMPONENTS IS INVOLVED.
C
C
C     SOME GENERAL COMMENTS ON INTERCEPT SOLUTIONS
C
C     THE X AXIS OF THE CYLINDER SYSTEM IS DEFINED BY THE DIRECTED LINE
C     SEGMENT FROM (XC,YC,ZC) TO THE BASE PLANE INTERCEPT
C      WITH EITHER THE
```

```
C     MAJOR X AXIS, OR THE MAJOR Y AXIS, OR THE MAJOR Z AXIS DEFINED BY
C     THE PROGRAMMER.
C
C     THE START OF THE STANDARD TESTS
C
  110 IF (UX.EQ.1.0.AND.UY.EQ.0.AND.UZ.EQ.0) 170,120
C
C
C     DOES U POINT IN DIRECTION OF -X
C
  120 IF (UX.EQ.-1.0.AND.UY.EQ.0.AND.UZ.EQ.0) 180,130
C
C
C     DOES U POINT IN DIRECTION OF +Y
C
  130 IF (UX.EQ.0.AND.UY.EQ.1.0.AND.UZ.EQ.0) 190,140
C
C
C     DOES U POINT IN DIRECTION OF -Y
C
  140 IF (UX.EQ.0.AND.UY.EQ.-1.0.AND.UZ.EQ.0) 200,150
C
C
C     DOES U POINT IN DIRECTION OF +Z
C
  150 IF (UX.EQ.0.AND.UY.EQ.0.AND.UZ.EQ.1.0) 210,160
C
C
C     DOES U POINT IN DIRECTION OF -Z
C
  160 IF (UX.EQ.0.AND.UY.EQ.0.AND.UZ.EQ.-1.0) 220,230
C
C     THE DIRECTION COSINE SET FOR +X VECTOR ORIENTATION.
C
  170 CALL FDCOSCYL (0.,1.0,0.0,0.1.0,1.0,0.0,DCOSCYL)
      GO TO 370
C
C
C     THE DIRECTION COSINE SET FOR -X VECTOR ORIENTATION.
C
  180 CALL FDCOSCYL (0.,0.1.0,0.1.0,0.-1.0,0.0,DCOSCYL)
      GO TO 370
C
C
C     THE DIRECTION COSINE SET FOR +Y VECTOR ORIENTATION.
C
  190 CALL FDCOSCYL (0.,0.1.0,1.0,0.0,0.1.0,0.DCOSCYL)
      GO TO 370
C
C
C     THE DIRECTION COSINE SET FOR -Y VECTOR ORIENTATION.
C
  200 CALL FDCOSCYL (1.0,0.0,0.0,1.0,0.-1.0,0.DCOSCYL)
      GO TO 370
C
C
```

```
C     THE DIRECTION COSINE SET FOR +Z VECTOR ORIENTATION
C
C
  210 CALL FDCOSCYL (1,0,0,0,0,1,0,0,0,0,1,0,DCOSCYL)
      GO TO 370
C
C
C     THE DIRECTION COSINE SET FOR -Z VECTOR ORIENTATION.
C
  220 CALL FDCOSCYL (0,1,0,0,1,0,0,0,0,0,-1,0,DCOSCYL)
      GO TO 370
C
C
C      START OF THE SIX SKEWED SOLUTIONS
C
C
C      THE PROJECTION SOLUTIONS
C
C
C     IS THE POINT AT THE ORIGIN OR ALONG THE X AXIS
C
C     IF SO, AN X PROJECTION SOLUTION IS APPROPRIATE.
C
  230 IF (((XC.EQ.0.AND.YC.EQ.0.AND.ZC.EQ.0).OR.(XC.NE.0.AND.YC.EQ.0.AND
     1.ZC.EQ.0)).AND.(UY.NE.0.)) 240,250
C
C     X INTERSECTION SOLUTION
C
  240 PHI=ATAN((-UX)/(UY))
      T11=COS(PHI)
      T21=SIN(PHI)
      T31=0
      GO TO 360
C
C     IS THE POINT ALONG THE Y AXIS
C
C     IF SO, THEN A Y PROJECTION SOLUTION IS APPROPRIATE.
C
  250 IF ((YC.NE.0.AND.XC.EQ.0.AND.ZC.EQ.0).AND.(UZ.NE.0.)) 260,270
C
C     Y INTERSECTION SOLUTION
C
  260 PHI=ATAN((-UY)/(UZ))
      T11=0
      T21=COS(PHI)
      T31=SIN(PHI)
      GO TO 360
C
C
C     IS THE POINT ALONG THE Z AXIS
C
C     IF SO, THEN A Z PROJECTION SOLUTION IS APPROPRIATE
C
  270 IF ((ZC.NE.0.AND.XC.EQ.0.AND.YC.EQ.0).AND.(UX.NE.0.)) 280,290
C
C     Z INTERSECTION SOLUTION
```

```
C
  280 PHI=ATAN((-UZ)/(UX))
      T11=SIN(PHI)
      T21=0
      T31=COS(PHI)
      GO TO 360
C
C
C     THE INTERCEPT SOLUTIONS
C
C
  290 IF (XC.NE.0.AND.(YC.NE.0.OR.ZC.NE.0).AND.UX.NE.0) 300,310
C
C     X INTERCEPT SOLUTION
C
C     IF UX IS NOT EQUAL TO ZERO THEN AN INTERCEPT
C     BETWEEN THE BASE PLANE AND THE X MAJOR AXIS IS POSSIBLE
C
C
C     THE X INTERCEPT (XI) IS OBTAINED
C     BY SETTING Y AND Z TO ZERO IN THE
C     EQUATION FOR THE BASE PLANE, (1), AND SOLVING FOR X.
C
  300 XI=(UX*XC+UY*YC+UZ*ZC)/UX
C     LET
C
      DELX=XI-XC
C
C     XL,THE LENGTH OF A LINE FROM (XC,YC,ZC) TO (XI,0,0)
C
      XL=SQRT(DELX**2+YC**2+ZC**2)
C
C
C     THE DIRECTION COSINES FOR THE CYLINDER'S X AXIS (IF XI IS USED)
C
      T11=DELX/XL
      T21=-YC/XL
      T31=-ZC/XL
      GO TO 360
C
C
  310 IF (YC.NE.0.AND.(XC.NE.0.OR.ZC.NE.0).AND.UY.NE.0) 320,330
C
C     Y INTERCEPT SOLUTION
C
C     IF UY ≠ 0  THEN AN INTERCEPT BETWEEN THE BASE PLANE OF THE
C     CYLINDER AND THE Y-MAJOR AXIS IS POSSIBLE,
C
C
C     THE Y INTERCEPT (YI) IS OBTAINED
C     BY SETTING X AND Z TO ZERO IN THE
C     EQUATION FOR THE BASE PLANE, (1), AND SOLVING FOR Y
C
  320 YI=(UX*XC+UY*YC+UZ*ZC)/UY
C
C     LET
```

11/02/70

```
C                                                                        TCY2220
      DELY=YI-YC                                                         TCY2230
C                                                                        TCY2240
C                                                                        TCY2250
C     YL,THE LENGTH OF A LINE FROM (XC,YC,ZC) TO (0,YI,0)                TCY2260
C                                                                        TCY2270
      YL=SQRT(XC**2+DELY**2+ZC**2)                                       TCY2280
C                                                                        TCY2290
C                                                                        TCY2300
C     THE DIRECTION COSINES FOR THE CYLINDER≠S X AXIS (IF YI IS USED)    TCY2310
C                                                                        TCY2320
      T11=-XC/YL                                                         TCY2330
      T21=DELY/YL                                                        TCY2340
      T31=-ZC/YL                                                         TCY2350
      GO TO 360                                                          TCY2360
C                                                                        TCY2370
  330 IF (ZC.NE.0.AND.(XC.NE.0.OR.YC.NE.0).AND.UZ.NE.0) 340,350          TCY2380
C                                                                        TCY2390
C     Z INTERCEPT SOLUTION                                               TCY2400
C                                                                        TCY2410
C     IF UZ ≠ 0 THEN AN INTERCEPT BETWEEN THE BASE PLANE                 TCY2420
C     OF THE CYLINDER AND THE Z MAJOR AXIS IS POSSIBLE.                  TCY2430
C                                                                        TCY2440
C     THE Z INTERCEPT (ZI) IS OBTAINED                                   TCY2450
C     BY SETTING X AND Y TO ZERO IN THE                                  TCY2451
C     EQUATION FOR THE BASE PLANE, (1), AND SOLVING FOR Z.               TCY2460
C                                                                        TCY2470
  340 ZI=(UX*XC+UY*YC+UZ*ZC)/UZ                                          TCY2480
C                                                                        TCY2490
C                                                                        TCY2500
C     LET                                                                TCY2510
C                                                                        TCY2520
      DELZ=ZI-ZC                                                         TCY2530
C     ZL,THE LENGTH OF A LINE FROM (XC,YC,ZC) TO (0,0,ZI)                TCY2540
C                                                                        TCY2550
      ZL=SQRT(XC**2+YC**2+DELZ**2)                                       TCY2560
C                                                                        TCY2570
C                                                                        TCY2580
C     THE DIRECTION COSINES FOR THE CYLINDER≠S X AXIS (IF ZI IS USED)    TCY2590
C                                                                        TCY2600
      T11=-XC/ZL                                                         TCY2610
      T21=-YC/ZL                                                         TCY2620
      T31=DELZ/ZL                                                        TCY2630
      GO TO 360                                                          TCY2640
  350 PRINT 390                                                          TCY2650
      GO TO 380                                                          TCY2660
C                                                                        TCY2670
C                                                                        TCY2680
C     THE COMMON FOLLOW UP FOR ALL SIX COMPUTED SOLUTIONS                TCY2690
C                                                                        TCY2700
C     THE DIRECTION COSINES FOR THE CYLINDER≠S Z AXIS                    TCY2710
C     ARE THOSE OF U, THE AXIS VECTOR.                                   TCY2711
C                                                                        TCY2720
  360 T13=UX                                                             TCY2730
      T23=UY                                                             TCY2740
      T33=UZ                                                             TCY2750
```

```
C
C
C     THE CROSS PRODUCT TERMS FOR CYLINDER Z CROSS CYLINDER X
C     TO GET THE DIRECTION COSINES FOR CYLINDER Y.
C
      T12=T23*T31-T21*T33
      T22=T11*T33-T13*T31
      T32=T13*T21-T11*T23
C
C
      CALL FDCOSCYL (T11,T21,T31,T12,T22,T32,T13,T23,T33,DCOSCYL)
  370 CONTINUE
C
C     IANSWER=0 MEANS YES, AN AFFIRMATIVE ANSWER
C      YES, A SET OF DIRECTION COSINES FOR THE CYLINDER WAS FOUND
C
      IANSWER=0
      RETURN
C
C
C      IANSWER=1 MEANS NO, A NEGATIVE ANSWER
C      NO, A SET OF DIRECTION COSINES FOR THE CYLINDER WAS NOT FOUND
C
  380 IANSWER=1
      RETURN
C
  390 FORMAT (1X,*NO LOCAL COORDINATE SYSTEM FOUND*)
      END
```

```
 TESTCYL                                               11/02/70      ED      0

                                   IDENT     TESTCYL
PROGRAM LENGTH                01211
ENTRY POINTS     TESTCYL      00013
EXTERNAL SYMBOLS
                 THEND,
                 Q8QDICT,
                 FDCOSCYL
                 SQRTF
                 SINF
                 COSF
                 ATANF
                 STH,
   00340 SYMBOLS
```

```
                                                          11/02/70

      FUNCTION S32T048 (AR)                                               STR  10
C                                                                         STR  20
C     THE PURPOSE OF THIS FUNCTION   IS TO CORRECT A DEFICIENCY           STR  30
C     IN THE APT SYSTEM WITH RESPECT TO GIVING CYLINDER PARAMETERS        STR  40
C     FULL 48 BIT WORD REPRESENTATION                                     STR  50
C     THE FUNCTION GIVES THE FLOATING POINT MINUS ONE REPRESENTATION      STR  60
C                                                                         STR  70
C     AS 5776377777777777                                                 STR  80
C                                                                         STR  90
C     INSTEAD OF                                                          STR 100
C                                                                         STR 101
C        5776377777777774                                                 STR 120
C                                                                         STR 130
C     WHICH THE CDC 3800 SYSTEM SEEMS TO CREATE UNDER                     STR 131
C     DRUM SCOPE 2.0 RESIDENT 2R.0                                        STR 132
C                                                                         STR 133
      IC=AR                                                               STR 140
       IF((AR.EQ.5776377777777774B).OR.(ABSF(AR+1.0).LT.(0.00001)))10,20  STR 150
   10 S32T048=IC                                                          STR 160
      RETURN                                                              STR 170
   20 S32T048=AR                                                          STR 180
      END                                                                 STR 190-

 S32T048                                                  11/02/70        ED       0
                                      IDENT      S32T048
PROGRAM LENGTH                 00106
ENTRY POINTS    S32T048        00003
EXTERNAL SYMBOLS
                Q1Q10100
                Q8QDICT,
   00026 SYMBOLS
```

11/02/70

```
      SUBROUTINE FDCOSCYL (U1,U2,U3,U4,U5,U6,U7,U8,U9,DCOSCYL)
      DIMENSION DCOSCYL(1)
C
C
C     THE PURPOSE OF THIS SUBROUTINE
C     IS TO FILL THE DCOSCYL ARRAY
C     WITH A SET OF DIRECTION COSINES.
C     THE SUBROUTINE IS CALLED FROM SUBROUTINE TESTCYL.
C
      DCOSCYL(1)=U1
      DCOSCYL(2)=U2
      DCOSCYL(3)=U3
      DCOSCYL(4)=U4
      DCOSCYL(5)=U5
      DCOSCYL(6)=U6
      DCOSCYL(7)=U7
      DCOSCYL(8)=U8
      DCOSCYL(9)=U9
C
      RETURN
      END
```

```
FDCOSCYL                                          11/02/70        ED      0

                                  IDENT     FDCOSCYL
PROGRAM LENGTH              00150
ENTRY POINTS     FDCOSCYL   00003
EXTERNAL SYMBOLS
                 Q8QDICT.
   00057 SYMBOLS
```

11/02/70

```
      SUBROUTINE ALGORA (WHICH)
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP
      REAL K1,K2,K3
C
C
C
C     ALGORITHM A COMPUTES THE COEFFICIENTS
C     NEEDED TO EXPRESS THE CYLINDER AND/OR SPHERE
C     IN TERMS OF XYZ (THE MAJOR COORDINATE SYSTEM) , THAT IS :
C     FOR A CYLINDER=
C       X(CYLSYSTEM)**2+Y(CYLSYSTEM)**2=R**2
C     AND FOR A SPHERE=
C       X(SPHSYSTEM)**2+Y(SPHSYSTEM)**2+Z(SPHSYSTEM)**2=R**2
C     GET CONVERTED IN THIS SUBROUTINE.
C
C     THE CXYZ ARRAY CONTAINS THE ≠≠A≠≠ COEFFICIENT SET
C     RESULTING FROM SUCH CONVERSIONS=
C
C      (A11 X + A12 Y + A13 Z + A14) X
C     +(A21 X + A22 Y + A23 Z + A24) Y
C     +(A31 X + A32 Y + A33 Z + A34) Z
C     +(A41   + A42   + A43   + A44) = 0
C
C     R IS THE RADIUS OF THE CYLINDER,(XC,YC,ZC) IS THE AXIS POINT
C
C     THE DCOXCYL ARRAY CONTAINS THE DIRECTION COSINES THAT
C     DEFINE THE CYLINDER COORDINATE SYSTEM
C
C
C     RETURN TO THE ≠≠T≠≠ SUBSCRIPTS
C
      T11=S32TO48(DCOSCYL(1))
      T21=S32TO48(DCOSCYL(2))
      T31=S32TO48(DCOSCYL(3))
      T12=S32TO48(DCOSCYL(4))
      T22=S32TO48(DCOSCYL(5))
      T32=S32TO48(DCOSCYL(6))
      T13=S32TO48(DCOSCYL(7))
      T23=S32TO48(DCOSCYL(8))
      T33=S32TO48(DCOSCYL(9))
C
C
C     DEFINE THREE CONSTANTS  K1,K2,K3
C
      K1=T11*XC+T21*YC+T31*ZC
      K2=T12*XC+T22*YC+T32*ZC
      K3=T13*XC+T23*YC+T33*ZC
C
C     THEN =
      IF (WHICH.EQ.8HCYLINDER) 10,20
   10 CONTINUE
C
C     A CYLINDER
```

```
C
      A11=(T11**2+T12**2)
      A12=(T11*T21+T12*T22)
      A13=(T11*T31+T12*T32)
      A14=-(K1*T11+K2*T12)
      A21=A12
      A22=(T21**2+T22**2)
      A23=(T21*T31+T22*T32)
      A24=-(K1*T21+K2*T22)
      A31=A13
      A32=A23
      A33=(T31**2+T32**2)
      A34=-(K1*T31+K2*T32)
      A41=A14
      A42=A24
      A43=A34
      A44=(K1**2+K2**2-R**2)
      GO TO 30
C
C     A SPHERE
C
   20 CONTINUE
      A11=(T11**2+T12**2+T13**2)
      A12=(T11*T21+T12*T22+T13*T23)
      A13=(T11*T31+T12*T32+T13*T33)
      A14=-(K1*T11+K2*T12+K3*T13)
      A21=A12
      A22=(T21**2+T22**2+T23**2)
      A23=(T21*T31+T22*T32+T23*T33)
      A24=-(K1*T21+K2*T22+K3*T23)
      A31=A13
      A32=A23
      A33=(T31**2+T32**2+T33**2)
      A34=-(K1*T31+K2*T32+K3*T33)
      A41=A14
      A42=A24
      A43=A34
      A44=(K1**2+K2**2+K3**2-R**2)
C
C     LOADING UP THE CXYZ ARRAY
   30 CONTINUE
C
      CXYZ(1)=A11
      CXYZ(2)=A12
      CXYZ(3)=A13
      CXYZ(4)=A14
      CXYZ(5)=A21
      CXYZ(6)=A22
      CXYZ(7)=A23
      CXYZ(8)=A24
      CXYZ(9)=A31
      CXYZ(10)=A32
      CXYZ(11)=A33
      CXYZ(12)=A34
      CXYZ(13)=A41
      CXYZ(14)=A42
```

11/02/70

```
      CXYZ(15)=A43                                                    ALA 960
      CXYZ(16)=A44                                                    ALA 970
      RETURN                                                          ALA 980
      END                                                             ALA 990-
```

```
   ALGORA                                                   11/02/70       ED      0

                                      IDENT      ALGORA
 PROGRAM LENGTH                 00423
 ENTRY POINTS     ALGORA        00004
 BLOCK NAMES
                  CROSSEC       02525
 EXTERNAL SYMBOLS
                  Q1Q03100
                  Q8QDICT,
                  S32T048
    00122 SYMBOLS
```

11/02/70

```
      SUBROUTINE ALGORB
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),H0(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),FH(16),IOKTEST,TL,IL
     3OOP
C
C     THE SIMULTANEOUS SOLUTION OF THE HOPE PLANE EQUATION AND THE
C     CYLINDER EQUATION IS ACCOMPLISHED BY ADDING THE FIRST DEGREE
C     COEFFICIENTS AND THE CONSTANT TERMS,  THAT IS.
C      2*THE NEW A14 = (2*THE OLD A14 +  A)  FOR THE X COEFFICIENT
C      2* THE NEW A24 = (2*THE OLD A24 +  B)  FOR THE Y COEFFICIENT
C      2*THE NEW A34 = (2*THE OLD A34 +  C)  FOR THE Z COEFFICIENT
C        THE NEW A44 = (  THE OLD A44 -  D)  FOR THE CONSTANT
C
C     WHERE A,B,C, D ARE FROM THE PLANE EQUATION (AX+BY+CZ=D)
C
C     THE CXYZ ARRAY CONTAINS THE COEFFICIENTS
C     FOR THE CYLINDER EQUATION OR THE SPHERE EQUATION
C     IN THE XYZ MAJOR COORDINANT SYSTEM
C
C     THE EXYZ ARRAY CONTAINS THE COEFFICIENTS FOR THE ELLIPSE(CIRCLE)
C     OF INTERSECTION IN THE XYZ MAJOR COORDINANT SYSTEM
C
C
C     FOR A11             THE X**2    COEFFICIENT
      EXYZ(1)=CXYZ(1)
C
C     FOR A12             1/2 THE XY COEFFICIENT
      EXYZ(2)=CXYZ(2)
C
C     FOR A13             1/2 THE XZ COEFFICIENT
      EXYZ(3)=CXYZ(3)
C
C     FOR A14             1/2 THE X  COEFFICIENT
      EXYZ(4)=CXYZ(4)+A/2
C
C     FOR A21             1/2 THE XY COEFFICIENT
      EXYZ(5)=CXYZ(5)
C
C     FOR A22             THE Y**2    COEFFICIENT
      EXYZ(6)=CXYZ(6)
C
C     FOR A23             1/2 THE YZ COEFFICIENT
      EXYZ(7)=CXYZ(7)
C
C     FOR A24             1/2 THE  Y COEFFICIENT
      EXYZ(8)=CXYZ(8)+B/2
C
C     FOR A31             1/2 THE XZ COEFFICIENT
      EXYZ(9)=CXYZ(9)
C
C     FOR A32             1/2 THE YZ COEFFICIENT
      EXYZ(10)=CXYZ(10)
C
C     FOR A33             THE Z**2    COEFFICIENT
      EXYZ(11)=CXYZ(11)
```

11/02/70

```
C                                                                      ALB 530
C     FOR A34            1/2 THE  Z COEFFICIENT                        ALB 540
      EXYZ(12)=CXYZ(12)+C/2                                            ALB 550
C                                                                      ALB 560
C     FOR A41            1/2 THE  X COEFFICIENT                        ALB 570
      EXYZ(13)=EXYZ(4)                                                 ALB 580
C                                                                      ALB 590
C     FOR A42            1/2 THE  Y COEFFICIENT                        ALB 600
      EXYZ(14)=EXYZ(8)                                                 ALB 610
C                                                                      ALB 620
C     FOR A43            1/2 THE  Z COEFFICIENT                        ALB 630
      EXYZ(15)=EXYZ(12)                                                ALB 640
C                                                                      ALB 650
C     FOR A44            THE CONSTANT TERM                             ALB 660
      EXYZ(16)=CXYZ(16)-D                                              ALB 670
C                                                                      ALB 680
      RETURN                                                           ALB 690
      END                                                              ALB 700-
```

```
  ALGORB                                                  11/02/70         ED      0
                                    IDENT       ALGORB
PROGRAM LENGTH                00042
ENTRY POINTS      ALGORB      00003
BLOCK NAMES
                  CROSSEC     02525
EXTERNAL SYMBOLS
                  Q8QDICT,
   00051 SYMBOLS
```

11/02/70

```
      SUBROUTINE ALGORC
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),H0(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP
C
C
C     ALGORITHM C OBTAINS THE ELLIPSE/CIRCLE EQUATION
C     IN HOPE PLANE COORDINATES,
C     I.E. IN TERMS OF THE ≠PRIME≠ COORDINATE SYSTEM WHOSE ORIGIN IS AT
C     THE  INTERSECTION OF THE HOPE PLANE AND ITS NORMAL.
C
C     THE EXYZ ARRAY CONTAINS THE COEFFICIENT ARRAY IN TERMS OF THE
C     XYZ*MAJOR COORDINATE SYSTEM
C
C     THE XMAT9 ARRAY CONTAINS THE CONVERSION CONSTANTS BETWEEN THE
C     XYZ*MAJOR AND THE ≠PRIME≠ COORDINATE SYSTEMS, A,B,C,D ARE THE
C     HOPE PLANE CONSTANTS,
C
C     THE EH    ARRAY WILL RECEIVE THE COMPUTED COEFFICIENTS  FOR THE
C     ≠PRIME≠ COORDINATE SYSTEM IN THE HOPE PLANE,
C
C
C     USING ≠≠H≠≠ INSTEAD OF THE USUAL ≠≠T≠≠, THE DIRECTION COSINES
C     THAT DEFINE THE PRIME AXES ARE IDENTIFIED FROM THE XMAT9 ARRAY
C
      H11=XMAT9(1)
      H21=XMAT9(2)
      H31=XMAT9(3)
      H12=XMAT9(5)
      H22=XMAT9(6)
      H32=XMAT9(7)
      H13=XMAT9(9)
      H23=XMAT9(10)
      H33=XMAT9(11)
C
C
C     THE COORDINATES OF THE ≠≠PRIME≠≠ ORIGIN (X0,Y0,Z0)
C
      X0=D*A
      Y0=D*B
      Z0=D*C
C
C
C     THE COEFFICIENTS W,R,T, XYZ MAJOR ARE IDENTIFIED WITH ≠≠A≠≠
C     AND ITS SUBSCRIPTS,
C
      A11=EXYZ(1)
      A12=EXYZ(2)
      A13=EXYZ(3)
      A14=EXYZ(4)
      A21=EXYZ(5)
      A22=EXYZ(6)
      A23=EXYZ(7)
      A24=EXYZ(8)
      A31=EXYZ(9)
```

11/02/70

```
      A32=EXYZ(10)
      A33=EXYZ(11)
      A34=EXYZ(12)
      A41=EXYZ(13)
      A42=EXYZ(14)
      A43=EXYZ(15)
      A44=EXYZ(16)
C
C
C     THE COMPUTED COEFFICIENTS FOR THE ≠≠PRIME≠≠ SYSTEM ARE
C     IDENTIFIED BY THE LETTER ≠≠P≠≠ AND THE CORRESPONDING SUBSCRIPTS
C
C     P11 FOR THE COEFFICIENT OF X PRIME SQUARED
C
C
      P11=(A11*(H11**2)+A22*(H21**2)+A33*(H31**2)+2*A12*H11*H21+2*A13*H1
     11*H31+2*A23*H21*H31)
C
C
C     P12 FOR 1/2 THE COEFFICIENT OF X≠Y≠ (WHERE ≠ IS THE PRIME SYMBOL)
C
      P12=(A11*H11*H12+A22*H21*H22+A33*H31*H32+A12*(H11*H22+H21*H12)+A13
     1*(H11*H32+H31*H12)+A23*(H21*H32+H22*H31))
C
C
C     P13 FOR 1/2 THE COEFFICIENT OF X≠Z≠
C
      P13=(A11*H11*H13+A22*H21*H23+A33*H31*H33+A12*(H11*H23+H21*H13)+A13
     1*(H11*H33+H31*H13)+A23*(H21*H33+H23*H31))
C
C
C     P14 FOR 1/2 THE COEFFICIENT OF X≠
C
      P14=(A11*H11*X0+A22*H21*Y0+A33*H31*Z0+A12*(H21*X0+H11*Y0)+A13*(H11
     1*Z0+H31*X0)+A23*(H21*Z0+H31*Y0)+A14*H11+A24*H21+A34*H31)
C
C
C     P21 FOR 1/2 THE COEFFICIENT OF X≠Y≠
C
      P21=P12
C
C
C     P22 FOR THE COEFFICIENT OF Y≠**2
C
      P22=(A11*(H12**2)+A22*(H22**2)+A33*(H32**2)+2*(A12*H12*H22)+2*(A13
     1*H12*H32)+2*(A23*H22*H32))
C
C
C     P23 FOR 1/2 THE COEFFICIENT OF Y≠Z≠
C
      P23=(A11*H12*H13+A22*H22*H23+A33*H32*H33+A12*(H12*H23+H22*H13)+A13
     1*(H12*H33+H13*H32)+A23*(H22*H33+H32*H23))
C
C
C     P24 FOR 1/2 THE COEFFICIENT OF Y≠
C
```

```
      P24=(A11*H12*X0+A22*H22*Y0+A33*H32*Z0+A12*(H22*X0+H12*Y0)+A13*(H12
     1*Z0+H32*X0)+A23*(H22*Z0+H32*Y0)+A14*H12+A24*H22+A34*H32)
C
C
C     P31 FØR 1/2 THE CØEFFICIENT ØF X'Z'
C
      P31=P13
C
C
C     P32 FØR 1/2 THE CØEFFICIENT ØF Y'Z'
C
      P32=P23
C
C
C     P33 FØR THE CØEFFICIENT ØF Z'**2
C
      P33=(A11*(H13**2)+A22*(H23**2)+A33*(H33**2)+2*A12*(H13*H23)+2*A13*
     1(H13*H33)+2*A23*(H23*H33))
C
C
C     P34 FØR 1/2 THE CØEFFICIENT ØF Z'
C
      P34=(A11*H13*X0+A22*H23*Y0+A33*H33*Z0+A12*(H23*X0+H13*Y0)+A13*(H13
     1*Z0+H33*X0)+A23*(H23*Z0+H33*Y0)+A14*H13+A24*H23+A34*H33)
C
C
C     P41 FØR 1/2 THE CØEFFICIENT ØF X'
C
      P41=P14
C
C
C     P42 FØR 1/2 THE CØEFFICIENT ØF Y'
C
      P42=P24
C
C
C     P43 FØR 1/2 THE CØEFFICIENT ØF Z'
C
      P43=P34
C
C
C     P44 FØR THE CONSTANT TERM
C
      P44=(A11*(X0**2)+A22*(Y0**2)+A33*(Z0**2)+2*A12*X0*Y0+2*A13*X0*Z0+2
     1*A23*Y0*Z0+2*A14*X0+2*A24*Y0+2*A34*Z0+A44)
C
C
C     FILLING THE EH ARRAY
C
      EH(1)=P11
      EH(2)=P12
      EH(3)=P13
      EH(4)=P14
      EH(5)=P21
      EH(6)=P22
      EH(7)=P23
```

```
      EH(8)=P24                                              ALC1670
      EH(9)=P31                                              ALC1680
      EH(10)=P32                                             ALC1690
      EH(11)=P33                                             ALC1700
      EH(12)=P34                                             ALC1710
      EH(13)=P41                                             ALC1720
      EH(14)=P42                                             ALC1730
      EH(15)=P43                                             ALC1740
      EH(16)=P44                                             ALC1750
C                                                            ALC1760
      RETURN                                                 ALC1770
      END                                                    ALC1780-
```

```
  ALGORC                                        11/02/70       ED      0

                               IDENT     ALGORC
PROGRAM LENGTH           00557
ENTRY POINTS     ALGORC  00003
BLOCK NAMES
                 CROSSEC 02525
EXTERNAL SYMBOLS
                 Q8QDICT.
   00126 SYMBOLS
```

11/02/70

```
      SUBROUTINE ALGOR D(JSUBER,NAMECYL)
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP,IP,KR(200),IONCE
      REAL MAJOR,MINOR
      REAL L1,L2
C
C
C
C     THE PURPOSE OF ALGORITHM D IS TO-
C         1) LOCATE THE CENTER OF THE ELLIPSE/CIRCLE IN THE HOPE PLANE
C            (TO BE EXPRESSED AS CX AND CY),
C
C         2) COMPUTE IT≠S SEMI-MAJOR AND SEMI-MINOR AXES
C            (EXPRESSED AS MAJOR OR MINOR)
C            OR ITS RADIUS IF A CIRCLE  (RADIUS)
C
C         3) ITS ROTATION ANGLE (PHI), AND TO
C
C         4) PROVIDE  A YES OR NO FLAG ON THE VALIDITY
C            OF THE ELLIPSE/CIRCLE
C            (IOKTEST SERVES AS THIS FLAG)
C
C     THE EH ARRAY CONTAINS THE ELLIPSE/CIRCLE COEFFICIENTS
C     IN THE HOPE PLANE SYSTEM.
C
C     ONLY COEFFICIENTS IN XPRIME AND YPRIME ARE USED SINCE Z≠ = 0.
C
C
C     IDENTIFYING AND FETCHING THE COEFFICIENTS (TWO DIMENSIONAL SET)
C
      PI=3.14159265
      PIOV2=1.570796325
      RDTDG=57.29578
      MAJOR=0
      MINOR=0
      RADIUS=0
C     E11            THE COEFFICIENT FOR X≠**2
C
      E11=EH(1)
C
C
C     E12        1/2 THE COEFFICIENT FOR X≠Y≠
C
      E12=EH(2)
C
C
C     E13        1/2 THE COEFFICIENT FOR X≠
C
      E13=EH(4)
C
C
C     E22            THE COEFFICIENT FOR Y≠**2
C
      E22=EH(6)
```

11/02/70

```
C
C
C     E23        1/2 THE COEFFICIENT FOR Y≠
C
      E23=EH(8)
C
C
C     E33              THE CONSTANT TERM
C
      E33=EH(16)
C
C
C     THESE EQUALITIES EXIST
C
      E21=E12
      E31=E13
      E32=E23
      H0(1)=E11
      H0(2)=E12
      H0(3)=E13
      H0(4)=E21
      H0(5)=E22
      H0(6)=E23
      H0(7)=E31
      H0(8)=E32
      H0(9)=E33
C
C
C     COMPUTE THE INVARIANTS OF A PLANE SECOND ORDER CURVE: I, D, A
C     REPRESENTED BY ≠≠TI≠≠, ≠≠DI≠≠, AND ≠≠AI≠≠, RESPECTIVELY,
C
      TI=E11+E22
C
      DI=E11*E22-E21*E12
C
      AI=E11*(E22*E33-E32*E23)-E12*(E21*E33-E31*E23)+E13*(E21*E32-E31*E2
     12)
C
C
C     TEST FOR PARALLEL POSSIBILITIES
C
      IF ((ABSF(AI).LT.0.0001).AND.(ABSF(DI).LT.0.0001)) 10,20
C
   10 CONTINUE
      AP=(E22*E33-E23*E32)+(E11*E33-E13*E31)
      LINE=8H
      CALL H0PARCYL (LINE,H0,TL,AP,SCALE,A,B,C,D,XMAT9,XC,YC,ZC,UX,UY,UZ
     1,R,NAMECYL)
      IF (LINE.EQ.7HYESLINE) G0 T0 350
      PRINT 360, NAMECYL
      G0 T0 350
C
C       TEST FOR PRESENCE OF ELLIPSE OR CIRCLE
C
C
   20 CONTINUE
```

11/02/70

```
      IF ((AI.NE.0).AND.(DI.GT.0).AND.((AI/TI).LT.0)) 40,30
   30 IF ((AI.NE.0).AND.(DI.GT.0).AND.(ABSF(E11-E22).LT.0.0001).AND.(E12
     1.EQ.0)) 50,350
C
   40 CØNTINUE
      GØ TØ 60
C
C
   50 CØNTINUE
      GØ TØ 60
C
   60 CØNTINUE
C
C
C     THE CENTER IS AT=
C
      CX=-(E13*E22-E12*E23)/DI
      CY=-(E11*E23-E13*E21)/DI
C
C
C     SØLVING THE CHARACTERISTIC EQUATIØN
C
C     L**2-IL+D=0
C
C     WHERE ≠≠L≠≠ STANDS FØR LAMBDA
C     AND I=TI
C     AND D=DI
C
C     CØEFFICIENT FØR L**2
C
      AA=1
C
C     CØEFFICIENT FØR L
C
      BB=(-TI/2)
C
C     CØNSTANT
C
      CC=DI
C
C     A QUADRATIC SØLUTIØN
C
      CALL QUAD (AA,BB,CC,R1,R2)
C
      IF (JSUBER.EQ.1050) 70,80
   70 PRINT 370
      GØ TØ 350
   80 IF (JSUBER.EQ.1051) 90,100
   90 PRINT 380
      GØ TØ 350
C
  100 IF (R1.EQ.R2) 110,120
C
  110 CØNTINUE
      L1=R1
      L2=R2
```

11/02/70

```
      GO TO 340
C
  120 IF (R1.GT.R2) 130,140
C
  130 L1=R1
      L2=R2
      GO TO 150
C
  140 L1=R2
      L2=R1
C
  150 MAJOR=SQRT(-(AI)/(L1*(L2**2)))
      MINOR=SQRT(-(AI)/((L1**2)*L2))
C
       IF((E12.EQ.0).OR.(ABSF(E12).LT.0.0001))160,210
  160 CONTINUE
      JSUBER=0
      AA=E22
      BB=E23
      CC=E33+2*E13*CX+E11*CX*CX
      CALL QUAD (AA,BB,CC,R1,R2)
      IF ((JSUBER.NE.1050).AND.(JSUBER.NE.1051).AND.(ABSF(ABSF(R1-CY)-AB
     1SF(R2-CY)).LT.0.0001)) 180,170
  170 PRINT 390, E22,E23,E33,CX,CY,MAJOR,MINOR
      GO TO 350
  180 IF (ABSF(ABSF(R1-R2)-2*MINOR).LT.0.0001) 190,200
C
C                                              NO ROTATION NEEDED
  190 PHI=0
      GO TO 330
C
C                                                 90 DEG ROTATION  NEEDED
  200 PHI=PIOV2
      GO TO 330
C
C
C                                               THERE IS AN XY TERM
C
C                                               IS THERE A 45 DEG SOLUTION
  210 CONTINUE
      IF ((ABSF(E11-E22).LT.0.0001).AND.(E12.NE.0)) 220,230
C
C                                              45 DEG SOLUTION
C
  220 PHI1=PIOV2/2
      GO TO 240
C
C     DETERMINING THE PROPER VALUE TO ASSIGN TO PHI
C
  230 PHI1=0.5*ATAN((2*E12)/(E11-E22))
  240 PHI2=PHI1+PIOV2
      IF ((E12.LT.0).AND.(PHI1.GT.0)) 250,260
  250 PHI=PHI1
      GO TO 330
  260 IF ((E12.LT.0).AND.(PHI2.LT.PIOV2)) 270,280
  270 PHI=PHI2
```

```
      GO TO 330
  280 IF ((E12.GT.0).AND.(PHI1.LT.0)) 290,300
  290 PHI=PHI1
      GO TO 330
  300 IF ((E12.GT.0).AND.(PHI2.GT.PIOV2)) 310,320
  310 PHI=PHI2
      GO TO 330
  320 CONTINUE
      DPHI2=PHI2*RDTDG
      DPHI1=PHI1*RDTDG
      PRINT 400, E12,E11,E22,DPHI1,DPHI2
      GO TO 350
  330 CONTINUE
      PHI=-PHI
      JSUBER=0
C
C     MESSAGE THAT CIRCLE NOT FOUND
C
      RADIUS=99999
C
      IOKTEST=0
      RETURN
C
C
C     SOLUTION FOR A CIRCLE
C
  340 RADIUS=SQRT(-(AI)/(L2*(L2**2)))
C
C     MESSAGE THAT ELLIPSE NOT FOUND
C
      MAJOR=99999
C
C
      IOKTEST=0
C
      RETURN
C
C
C     IOKTEST=0 MEANS YES, IOKTEST=1 MEANS NO
C
  350 IOKTEST=1
      RETURN
C
  360 FORMAT (30X,* PARALLEL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS
     1 OF CYLINDER.  THUS, NO INTERSECTION*,/)
  370 FORMAT (1X,*NO SOLUTION FOR LAMBDA. BOTH A AND B ARE ZERO*)
  380 FORMAT (1X,*NO REAL ROOTS FOR LAMBDA*)
  390 FORMAT (1X,*PHI COMPUTATION TROUBLE WITH XY TERM = 0.NO SOLUTION W
     1AS FOUND FOR Y*,/,F10.5,*YSQUARE +*,F10.5,*Y +*,F10.5,* = 0*,/,1X,
     2*CX, CY*,1X,F10.5,1X,F10.5,* A,B *,F10.5,1X,F10.5)
  400 FORMAT (1X,*UNABLE TO DETERMINE PHI*,/,1X,*E12 = *,F10.7,/,1X,*E11
     1 = *,F10.7,/,1X,*E22 = *,F10.7,/,1X,*PHI 1 = *,F10.2,/,1X,*PHI 2 =
     2 *,F10.2)
      END
```

```
                                        IDENT     ALGORD
PROGRAM LENGTH              01041
ENTRY POINTS    ALGORD      00154
BLOCK NAMES
                CROSSEC     03037
EXTERNAL SYMBOLS
                THEND,
                Q8QDICT,
                HOPARCYL
                QUAD
                SQRTF
                ATANF
                STH,
                QNSINGL,
   00240 SYMBOLS
```

UNCLASSIFIED

11/09/70

```
      SUBROUTINE HOPARCYL (LINE,HO,TL,AP,SCALE,A,B,C,D,XMAT9,XC,YC,ZC,UX
     1,UY,UZ,R,NAMECYL)
      DIMENSION HO(1)
      DIMENSION XMAT9(1)
      REAL K
      REAL M12,M34
C
C
C
C     THIS SUBROUTINE IS CALLED FROM ALGORD IF IT HAS BEEN
C     DETERMINED THAT THE HOPE PLANE IS PARALLEL TO THE AXIS OF THE
C     CYLINDER.  ITS PURPOSE IS TO PLOT THE DEGENERATE LINE SOLUTION AS
C     A SINGLE LINE, OR A PAIR OF PARALLEL LINES EITHER HORIZONTALLY,
C     VERTICALLY OR SKEWED.
C     IF THE SOLUTION IS TWO PARALLEL LINES, THEY
C     ARE JOINED ON THEIR ENDS TO FORM A RECTANGLE TO COMPLETE THE
C     INTERSECTION OF A FINITE CYLINDER WITH END PLAINS FORMING A
C     RIGHT CIRCULAR CYLINDER.
C
C
C     THE PARAMETERS ARE
C
C     ≠≠LINE≠≠, A HOLLARITH MESSAGE 8HYESLINE OR 8HNOLINE
C     IS RETURNED TO INDICATE SUCCESS OR FAILURE OF THE CALL.
C
C     ≠≠HO≠≠, THE ARRAY CONTAINING THE HOPE PLANE INTERSECTION
C     COEFFICIENTS.
C
C     ≠≠TL≠≠, THE LENGTH OF THE CYLINDER.
C
C     ≠≠AP≠≠, THE INVARIANT WHICH WHEN =0 INDICATES A SINGLE DEGENERATE
C     LINE, AND WHEN LESS THAN ZERO, A PAIR OF PARALLEL LINES.
C
C     ≠≠SCALE≠≠, THE PLOTTING SCALE WHICH WAS COMPUTED IN CROSEC.
C
C     ≠≠A,B,C,D≠≠, THE HOPE PLANE CONSTANTS.
C
C     ≠≠XMAT9≠≠, THE HOPE PLANE CONVERSION MATRIX.
C
C     ≠≠XC,YC,ZC,UX,UY,UZ,R≠≠, THE CYLINDER CONSTANTS.
C
C     ≠≠NAMECYL≠≠, THE NAME OF THE CYLINDER IN HOLLARITH.
C
C
C     LOCAL SCALING FUNCTION
      S(W)=(W*SCALE+5.0)
C
C     START OF OUTPUT MESSAGE
C
      PRINT 410, NAMECYL
C
C     INTERSECTION COEFFICIENTS PICKED UP FROM HO ARRAY.
C
C                                                  X SQUARED
      A11=HO(1)
C                                                  XY
```

11/09/70

```
      A12=HO(2)                                                          HOP 550
C                                                X                       HOP 560
      A13=HO(3)                                                          HOP 570
C                                                XY                      HOP 580
      A21=HO(4)                                                          HOP 590
C                                                 YSQUARED               HOP 600
      A22=HO(5)                                                          HOP 610
C                                                Y                       HOP 620
      A23=HO(6)                                                          HOP 630
C                                                X                       HOP 640
      A31=HO(7)                                                          HOP 650
C                                                Y                       HOP 660
      A32=HO(8)                                                          HOP 670
C                                                CONSTANT                HOP 680
      A33=HO(9)                                                          HOP 690
C                                                                        HOP 700
C     DD IS DISTANCE FROM CENTER POINT TO HOPE PLANE,                    HOP 710
C     IF ≠≠DD≠≠ IS GREATER THAN R NO LINE SOLUTIONS ARE POSSIBLE.        HOP 720
C                                                                        HOP 730
      DD=(A*XC+B*YC+C*ZC-D)/SQRT(A**2+B**2+C**2)                         HOP 740
       IF (ABSF(DD).GT.R)  GO TO 400                                     HOP 750
C                                                                        HOP 760
C     THE ≠≠LO≠≠ AND ≠≠HI≠≠ POINTS ARE COMPUTED,                         HOP 770
C     THEY ARE THE PROJECTION OF THE END POINTS OF THE CYLINDER          HOP 780
C     VECTOR ONTO THE HOPE PLANE.                                        HOP 790
C                                                                        HOP 800
C                                                                        HOP 810
      XL=XC-DD*A                                                         HOP 820
      YL=YC-DD*B                                                         HOP 830
      ZL=ZC-DD*C                                                         HOP 840
      XH=XL+TL*UX                                                        HOP 850
      YH=YL+TL*UY                                                        HOP 860
      ZH=ZL+TL*UZ                                                        HOP 870
C                                                                        HOP 880
C     THE HOPE PLANE COORDINATES ARE OBTAINED,                           HOP 890
C                                                                        HOP 900
      CALL MM (XLO,YLO,ZLO,XL,YL,ZL,XMAT9)                               HOP 910
      CALL MM (XHI,YHI,ZHI,XH,YH,ZH,XMAT9)                               HOP 920
C                                                                        HOP 930
C     TESTING FOR PARALLEL LINE SOLUTION,                                HOP 940
C                                                                        HOP 950
      IF (AP.LT.0) 110,190                                               HOP 960
C                                                                        HOP 961
C                                     FOR VERTICAL PAIR                  HOP 970
C                         X**2 AND CONSTANT TERM MUST BE NON ZERO        HOP 990
C                         AND XY, Y**2 AND Y TERMS ZERO                  HOP1000
C                                                                        HOP1001
  110 IF ((A11.NE.0).AND.(A33.NE.0).AND.(A21.EQ.0).AND.(A22.EQ.0).AND.(AHOP1010
     123.EQ.0)) 120,150                                                  HOP1020
C                                                                        HOP1021
C                           IS  THE PAIR SYMMETRICAL ABOUT ORIGIN        HOP1030
C                           I.E.IS THERE AN X TERM                       HOP1040
C                                                                        HOP1041
  120 IF (A13.EQ.0) 130,140                                              HOP1050
C                                                                        HOP1051
C                            X=+ OR - SQRT(A33) ABOUT (0,0)              HOP1060
```

```
C
  130 Q=SQRT(ABSF(A33))
      CALL ENDDRAW (S(-Q),S(YLO),S(Q),S(YHI))
      W1=-Q
      W2=Q
      PRINT 420, W1,W2
      GO TO 300
C
C                              X = + OR - SQRT (A33) ABOUT (H,0)
C
  140 H=-A13
      Q=ABSF(SQRT(H**2-A33))
      CALL ENDDRAW (S(-Q+H),S(YLO),S(Q+H),S(YHI))
      W1=-Q+H
      W2=Q+H
      PRINT 420, W1,W2
      GO TO 300
C
C                              FOR HORIZONTAL PAIR
C                              Y**2 AND CONSTANT TERMS MUST BE NON-ZERO
C                              AND X**2,XY,AND X TERMS MUST BE ZERO
C
  150 IF ((A22.NE.0).AND.(A33.NE.0).AND.(A11.EQ.0).AND.(A12.EQ.0).AND.(A
     113.EQ.0)) 160,310
C
C                              IS PAIR SYMMETRICAL ABOUT ORIGIN
C                             -DEPENDS ON PRESENCE OF Y TERM
C
  160 IF (A23.EQ.0) 170,180
C
C                              Y = + OR - SQRT(A33) ABOUT (0,0)
C
  170 Q=SQRT(ABSF(A33))
      CALL ENDDRAW (S(XLO),S(-Q),S(XHI),S(Q))
      W1=-Q
      W2=Q
      PRINT 430, W1,W2
      GO TO 300
C
C                              Y = + OR - SQRT(A33) ABOUT (0,K)
C
  180 K=-A23
      Q=ABSF(SQRT(K**2-A33))
      CALL ENDDRAW (S(XLO),S(-Q+K),S(XHI),S(Q+K))
      W1=-Q+K
      W2=Q+K
      PRINT 430, W1,W2
      GO TO 300
C
C                              IS THERE A SINGLE LINE
C                              WITH HOPE TANGENT TO CYLINDER
C
  190 IF (AP.EQ.0) 200,310
C
C                              A VERTICAL SITUATION IF
C                               X**2 TERM NON ZERO
```

```
C                              AND XY,Y**2,AND Y TERMS ZERO
C
  200 IF ((A11.NE.0).AND.(A21.EQ.0).AND.(A22.EQ.0).AND.(A23.EQ.0)) 210,2
     150
C
C                                          IS IT THRU THE ORIGIN
C
  210 IF ((A13.EQ.0).AND.(A33.EQ.0)) 220,230
C
C                               X =0 OR THE Y AXIS IS THE LINE
C
  220 CALL DRAW (S(0),S(YLO),S(0),S(YHI))
      IW1=1HX
      W1=0
      PRINT 440, IW1,W1
      GO TO 300
C
C                              IS IT THRU ANOTHER POINT
C
  230 IF ((A13**2).EQ.(ABSF(A33))) 240,310
C
C                              X=-A13
C
  240 CALL DRAW (S(-A13),S(YLO),S(-A13),S(YHI))
      IW1=1HX
      W1=-A13
      PRINT 440, IW1,W1
      GO TO 300
C
C                              A HORIZONTAL SITUATION IF
C                               Y**2 TERM NON ZERO
C                              X**2,XY,AND X TERMS EQUAL TO ZERO
C
  250 IF ((A22.NE.0).AND.(A11.EQ.0).AND.(A12.EQ.0).AND.(A13.EQ.0)) 260,3
     110
C
C                              ALONG X AXIS , SYMMETRIC ABOUT ORIGIN. IF-
C                              Y TERM AND CONSTANT BOTH = 0
C
  260 IF ((A23.EQ.0).AND.(A33.EQ.0)) 270,280
C
C                                          Y=0
C
  270 CALL DRAW (S(XLO),S(0),S(XHI),S(0))
      IW1=1HY
      W1=0
      PRINT 440, IW1,W1
      GO TO 300
C
C                              NOT SYMMETRIC ABOUT ORIGIN
C
  280 IF ((A23**2).EQ.(ABSF(A33))) 290,310
C
C                                          Y=-A23
C
  290 CALL DRAW (S(XLO),S(-A23),S(XHI),S(-A23))
```

```
      IW1=1HY
      W1=-A23
      PRINT 440, IW1,W1
      GO TO 300
  300 LINE=8HYESLINE
      RETURN
  310 CONTINUE
C
C
C     IS A SINGLE, SKEWED LINE PRESENT.
C
C
      IF (((DD-R).LT.0.0001).AND.((A23**2)-(A22*A33).LT.0.0001)) 320,330
  320 CALL DRAW (S(XLO),S(YLO),S(XHI),S(YHI))
      PRINT 450, XLO,YLO,XHI,YHI
      GO TO 300
  330 CONTINUE
C
C     A SKEWED PAIR OF PARALLEL LINES IS NEEDED.
C
C     THE SLOPE OF THE LINES.
C
      PHI=ATANF((YHI-YLO)/(XHI-XLO))
      JSUBER=0
C
C     A GENERAL SOLUTION FOR THE INTERSECTIONS ON THE Y PRIME AXIS.
C
      CALL QUAD (A22,A23,A33,B1,B2)
      IF (JSUBER.EQ.1050) 340,350
  340 PRINT 460, A22,A23,A33
      PRINT 470
      GO TO 400
  350 IF (JSUBER.EQ.1051) 360,370
  360 PRINT 460, A22,A23,A33
      PRINT 480
      GO TO 400
  370 CONTINUE
      IF (B1.GT.B2) 380,390
  380 BBI=B1
      B1=B2
      B2=BBI
  390 CONTINUE
C
C     SLOPE, INTERCEPT SOLUTIONS FOR THE RECTANGLE SIDES.
C
      M12=TANF(PHI)
      M34=PHI+1.570796325
      M34=TANF(M34)
      B3=YHI-M34*XHI
      B4=YLO-M34*XLO
      DEN=M12-M34
      X1=(B3-B1)/DEN
      Y1=M12*X1+B1
      X2=(B3-B2)/DEN
      Y2=M12*X2+B2
      X3=(B4-B2)/DEN
```

```
      Y3=M12*X3+B2
      X4=(B4-B1)/DEN
      Y4=M12*X4+B1
C
C     THE SKEWED RECTANGLE IS PLOTTED,
C
      CALL PLOT (S(X1),S(Y1),3)
      CALL PLOT (S(X1),S(Y1),2)
      CALL PLOT (S(X2),S(Y2),2)
      CALL PLOT (S(X3),S(Y3),2)
      CALL PLOT (S(X4),S(Y4),2)
      CALL PLOT (S(X1),S(Y1),2)
      CALL PLOT (S(X1),S(Y1),3)
      PPHI=PHI*57.29478
      PRINT 490, PPHI,X1,Y1,X2,Y2,X3,Y3,X4,Y4
      GO TO 300
  400 LINE=8HNOLINE
      RETURN
C
  410 FORMAT (1H+,A8,*, INTERSECTION IS1*)
  420 FORMAT (30X,*VERTICAL PARALLEL LINES, X = *,F10.2,*, X = ,*F10.2,/
     1)
  430 FORMAT (30X,*HORIZONTAL PARALLEL LINES, Y = *,F10.2,*, Y = *,F10.2
     1,/)
  440 FORMAT (30X,*SINGLE LINE, *,A1,* = *,F10.2,/)
  450 FORMAT (30X,*SINGLE LINE, FROM( *,F10,2,*, *,F10,2,*) TO (*,F10,2,
     1*, *,F10,2,*)*)
  460 FORMAT (1X,*IN HOPARCYL, ROTATED QUADRATIC SOLUTION FOR Y*/,1X,*A
     1= *,F10,5,*, B = *,F10,5,*, C = *,F10,5)
  470 FORMAT (1X,*BOTH A AND B TERMS ZERO*)
  480 FORMAT (1X,*NO REAL ROOTS*)
  490 FORMAT (30X,*RECTANGLE AT *,F9.2,* DEGREES*,/,30X,*FOUR CORNERS AS
     1 XY PAIRS, *,4(*( *,F9.2,*, *,F9.2,*) *))
      END
```

```
HOPARCYL                                         11/09/70        ED      0

                                  IDENT      HOPARCYL
PROGRAM LENGTH               02036
ENTRY POINTS     HOPARCYL    00206
EXTERNAL SYMBOLS
                 THEND.
                 Q8QDICT.
                 MM
                 ENDDRAW
                 DRAW
                 QUAD
                 PLOT
                 TANF
                 SQRTF
                 ATANF
                 STH.
                 QNSINGL.
   00424 SYMBOLS
```

11/02/70

```
      SUBROUTINE DRAW (X1,Y1,X2,Y2)
C
C
C     THIS SUBROUTINE IS CALLED FROM ≠≠HOPARCYL≠≠.
C     IT IT USED TO DRAW EITHER A SINGLE LINE  OR A RECTANGLE.
C     THE ORIENTATION OF THESE FIGURES IS EITHER HORIZONTAL OR VERTICAL.
C
C
      CALL PLOT (X1,Y1,3)
      CALL PLOT (X1,Y1,2)
      CALL PLOT (X2,Y2,2)
      CALL PLOT (X2,Y2,3)
      RETURN
      ENTRY ENDDRAW
      CALL PLOT (X1,Y1,3)
      CALL PLOT (X1,Y1,2)
      CALL PLOT (X1,Y2,2)
      CALL PLOT (X2,Y2,2)
      CALL PLOT (X2,Y1,2)
      CALL PLOT (X1,Y1,2)
      CALL PLOT (X1,Y1,3)
      RETURN
      END
```

```
    DRAW                                          11/02/70       ED       0

                                IDENT     DRAW
PROGRAM LENGTH              00157
ENTRY POINTS     DRAW       00003
                 ENDDRAW    00030
EXTERNAL SYMBOLS
                 Q8QDICT,
                 PLOT
   00036 SYMBOLS
```

11/02/70

```
      PROGRAM DISCRETE
      COMMON /TIMETEST/        KSETIME1, KSETIME2, KSETIMF,  KSETADR
C
C     ...     COMMON AREA FOR CDC APT 3 SECTION 0
C
C     *          ******         ******          ******          ******
      COMMON /SYSTEMZ/         SYSTEM(4),KAPTCN,  KAPTTR,   KAPTIO,
     1    KAPTID,   KFLAGS(10),KO,      K1,       K2,       K3,
     2    K4,       K5,        K6,      K7,       K8,       K9,
     3    IFILL1,   IFILL2,    IFILL3,  IFILL4,
     4    KFLAG0,   KFLAG1,    KFLAG2,  KFLAG3,   KFLAG4,
     5    IFILL5,   IWAVEN,    IPTNLY,  NOPOST,   IFILL6,   KAUTOP,
     6    ICLPRT,   INDEXX,    IPLOTR,  IFILL7,   NOPLOT,   KDYNFG,
     7    LOCJPT,   LOCBEG,    KSECIN,  NCLREC,   LOCMAC,   KPOCKET,
     8    IFILL8,   IFILL9,    IFILL10, IPOSTP(1),NUMPST,   IPOSTFL(18),
     9    TAPETB(1),CANTAP,    CLTAPE,  POCTAP,   PLOTAP,   SRFTAP,
     $    LIBTAP,   CRDTAP,    IFILL11, CORTAP,   TAPES1,   TAPES2,
     $    TAPES3,   TAPES4,    FORTIN,  INTAPE,   IOUTAP,   PUNTAP,
     $    LSTFLG,   LTVFLG,    KONVTCL, KINTRUPT,
     $    PI,       PI02,      DGTRD,   RDTDG,    ONE,
     $    EXTRA0(20)
      EQUIVALENCE    (PROTAP,  TAPETB)
C     *          ******         ******          ******          ******
C
C     ...     COMMON AREA FOR CDC APT 3 SECTION 1
C
C     *          ******         ******          ******          ******
      COMMON /SECT1LOG/        ITAB1,   ITAB2,    ITAB3,
     1    ITAB4,    ITAB5,     ISNAM,   ITAB11,   ITAB12,   ITAB13,
     2    JENDPTPP, JENDCAN,   JENDSTOR, JSTRTCAN, JENDSYM,
     3    JCANTEMP, JRPTAB,    JLPTAB,  MAXNST,
     4    JINWD,    JCHAR,     IWDERR,  JBUFL,    NUPERP,   NUPUN,
     5    JSTYPE,   JVARS2,    SCHERR,  NMACV,    MACASN(25),
     6    INDXPT,   IPTP,      IXPT,    MODE,     EOCFLG,   LPNDFL,
     7    TRMFLG,   INTRUPT,   JUMPFL,  ICDERR,   DEBUG,
     8    MACMODE,  NESTFL,    NRESULT, IPTLIM,   JEXEC,
     9    KTYPE,    MACTYP,
     $    IPARTERR, FINIS,     IOFLG,   MACDEL,   JSUBER,   NUMBERR,
     $    DEFSTO(85),     DEFTAB(1000),   ZSUR(30),
     $    XMAT4(16),XMAT3(16),XMAT2(16),XMAT1(16),TMATX(16),
     $    ISTDMODE, ISTDLIT,   ISTDTBL, ISTDINDX, ISTDTYPE, ISTDWD,
     $    JPTIND,   KPTCODE,   KPTNAME, KPTTYPE,  KPTNUM,   KPTINDX,
     $    KPTSUB,   KOMFLG,    KOMPOP,  NOSUBS,   KANFLG,   KRFSYS,
     $    KANREC,   KANCNT,    INAME,   KANSURF,  KANINDX,
     $    JRRELEN,  NEWCARD,   JGORIT,  NUMSTID,  NUMCSEQ,
     $    IRECIX,   IRECNO,    JTLPOS,  ITITLE(9),LSRECN,
     $    NNODEFX,  NNODEFI,   NIDUM,   ISLASH,   IEQUAL,   IBLANK,
     T    IDUMMY,   N10000,    N7777,   MASKU,    MASKL,    IDIV,
     U    MACREL,   MACLOC,    MACBEGN, MACLAST,  MACLEVEL,
     V    MACNAME(3), MACINDX(3), NMV,  JRESTOR,  MACPSH(3,25),
     W    JTEMP1,   JTEMP2,    JTEMP3,  JTEMP4,   JTEMP5,   JTEMP6,
     X    JTEMP7,   JTEMP8,    JTEMP9,  EXTRA1(20)
      EQUIVALENCE (DEFANS(1),IDEFSTO(4),DEFSTO(4)),(LSTYPE, KTYPE),
     1    (PTNUM, KPTNUM)
      DIMENSION IDEFSTO(85),  DEFANS(26),  IDEFTAB(1000)  ,
     1    ILPTAB(200), IRPTAB(200), ITNTAB(200), JPROTP(100),
```

```
      2    IRECSV(200), MACVAR(25),  MACNOR(25),  INWORD(10), IBUUP(2),
      3    PISTO(6), IPISTO(6), IDREC(4)
       EQUIVALENCE (INWORD(14), IBUUP(16), ILPTAB(990), IRPTAB(790),
      1    ITNTAB(590), PISTO(390), IPISTO(390), IDREC(384),
      2    JPROTP(380), IRECSV(280),  MACVAR(75), MACNOR(50),
      3    IDEFTAB(1000), DEFTAB(1000))
C
       COMMON/VOCABTBL/ KOM(1100)
C
       COMMON /2/ JTABNUM,JTABL(12120)
C      *          ******        ******          ***4***           ****
C
C      CONSTANTS CARRIED OVER FROM CROSEC.
C      SEE CROSEC AND CYLINDER COMMENTS FOR DETAILS
C
C
        COMMON/CROSSEC/
      1  A,B,C,D,IDSEND,XMAT9(16),
      2  ARRAY(254),
      3   XC,YC,ZC,UX,UY,UZ,R,
      4 SCALE,
      5  DCOSCYL(9),
      6  HO(9),
      7      XS(400),YS(400),
      8  X(103),Y(103),
      9  MAJOR,MINOR,PHI,CX,CY,RADIUS,ISET,
      $  CXYZ(16),EXYZ(16),EH(16),
      $  IOKTEST,
      $  TL,
      $     ILOOP,IP,KR(200),IONCE
        EQUIVALENCE (INCHOPE,IFILL8)
        EQUIVALENCE (PLOTPLNO,IFILL9)
        REAL MAJOR,MINOR
        REAL LIM
C
C
C      THIS≠≠PROGRAM≠≠ PROVIDES THE ENTRANCE POINT FOR SEGMENT 103,
C
C      ITS PURPOSE IS TO-
C
C       (1) CALL FOR THE COMPUTATION OF THE DISCRETE SET OF POINTS
C      THAT DEFINES THE PERIMETER OF THE ELLIPSE OR THE CIRCLE,
C
C      (2) IN THE CASE OF THE CIRCLE THE POINTS ARE IMMEDIATELY PLOTTED,
C      BUT FOR THE ELLIPSE IT MUST BE DETERMINED IF TRUNCATION IS
C      NECESSARY.  TO MAKE THIS DETERMINATION TEST ARE MADE ON ≠≠D1Z≠≠
C      AND ≠≠D2Z≠≠,  D1Z AND D2Z ARE THE DISTANCES FROM THE MAJOR-AXIS
C      END POINTS TO THE BASE PLANE OF THE CYLINDER MEASURED IN
C      THE Z DIRECTION OF THE CYLINDER≠S COORDINATE SYSTEM.
C      THE CALLS TO ELBOUNDS HANDLE THE ACTUAL TRUNCATION.
C
C      (3) TESTS ON D1Z AND D2Z NOT ONLY DETERMINE THE EXTENT OF
C      TRUNCATION, BUT ALSO DETERMINE THE SETTING FOR ≠≠IWHCPART≠≠,
C      WHICH PART OF THE ELLIPSE IS LOWER, THE POSITIVE END OR THE
C      NEGATIVE END OF THE MAJOR AXIS,
C
```

```
C     (4) PROVIDES FOR PROPER TERMINATION, EITHER,
C          (A) RETURN TO SEGMENT 102 VIA PRCNTL TO LOOK FOR MORE
C          CYLINDERS, OR
C          (B) RETURN TO ISHOPE FOR ANOTHER HOPE PLANE, OR
C          (C) GO BACK TO PRCNTL FOR APT CONTINUATION,
C
C
      TEST9=99999
      IF (IOKTEST.EQ.0.AND.MAJOR.EQ.TEST9) 10,30
C
C
C     COMPUTE SET OF ORIGIN ORIENTED POINTS FOR CIRCLE
C
   10 CALL POINTSC
C
C     ADD CENTER OFFSETS, CX AND CY,
C
      DO 20 I=1,200
      XS(I)=XS(I)+CX
      YS(I)=YS(I)+CY
   20 CONTINUE
      GO TO 130
C
   30 IF (IOKTEST.EQ.0.AND.RADIUS.EQ.TEST9) 40,120
C
C     COMPUTE SET OF ORIGIN ORIENTED POINTS FOR THE ELLIPSE,
C
   40 CALL POINTSE
C
C     INCLUDE ROTATION AND TRANSLATION
C
      DO 50 I=1,400
      XXS=XS(I)*COS(PHI)+YS(I)*SIN(PHI)
      YYS=-XS(I)*SIN(PHI)+YS(I)*COS(PHI)
      XS(I)=XXS+CX
      YS(I)=YYS+CY
   50 CONTINUE
C
C     CONVERT MAJOR AXIS END POINTS TO XYZ SYSTEM
C
      XPART=MAJOR*COS(-PHI)
      YPART=MAJOR*SIN(-PHI)
      P1=CX+XPART
      Q1=CY+YPART
      R1=0
      X1=0
      Y1=0
      Z1=0
C
C     CONVERT FROM HOPE COORDINATES TO XYZ SYSTEM,
C
      CALL CONVERT (P1,Q1,R1,X1,Y1,Z1)
C
C     CONVERT FROM XYZ SYSTEM TO CYLINDER SYSTEM.
C
      CALL CONVERT2 (PC,QC,RC,X1,Y1,Z1)
```

11/02/70

```
      D1Z=RC
      P2=CX*XPART
      Q2=CY*YPART
      R2=0
      X2=0
      Y2=0
      Z2=0
      CALL CONVERT (P2,Q2,R2,X2,Y2,Z2)
      CALL CONVERT2 (PC,QC,RC,X2,Y2,Z2)
      D2Z=RC
C
C     USE MAJOR AXIS END POINT DISTANCES
C     TO DETERMINE ≠≠IWHCPART≠≠ ORIENTATIONS.
C
      IF (((D1Z.GT.0).AND.(D2Z.GT.0).AND.(D1Z.LT.D2Z)).OR.((D1Z.LT.0).AN
     1D.(D2Z.GT.0)).OR.((D1Z.LT.0).AND.(D2Z.LT.0).AND.(ABSF(D1Z).GT.ABSF
     2(D2Z)))) 60,70
   60 IWHIPART=7HPOSDOWN
      GO TO 80
   70 IWHIPART=7HNEGDOWN
   80 CONTINUE
C
C
C     USE ≠≠D1Z≠≠ AND ≠≠D2Z≠≠ TO DETERMINE EXTENT OF
C     NECESSARY TRUNCATION, EITHER,
C          (A) NEITHER PLANE, OR
C          (B) BASE PLANE ONLY, OR
C          (C) TOP PLANE ONLY,OR
C          (D) BOTH PLANES.
C
C
      XTL=XC+UX*TL
      YTL=YC+UY*TL
      ZTL=ZC+UZ*TL
      IF (((D1Z.LE.0).AND.(D2Z.LE.0)).OR.((D1Z.GE.TL).AND.(D2Z.GE.TL)).O
     1R.(((D1Z.GT.0).AND.(D1Z.LE.TL)).AND.((D2Z.GT.0).AND.(D2Z.LE.TL))))
     1GO TO 130
      IF (((D1Z.LT.0).AND.((D2Z.GT.0).AND.(D2Z.LE.TL))).OR.
     1((D2Z.LT.0).AND.((D1Z.GT.0).AND.(D1Z.LE.TL))))GO TO 90
      IF (((D2Z.GT.TL).AND.((D1Z.GT.0).AND.(D1Z.LE.TL))).OR.
     1((D1Z.GT.TL).AND.((D2Z.GT.0).AND.(D2Z.LE.TL))))GO TO 100
      IF (((D1Z.LT.0).AND.(D2Z.GT.TL)).OR.((D2Z.LT.0).AND.(D1Z.GT.TL)))
     1GO TO 110
       PRINT 200,TL,D1Z,D2Z
   90 CONTINUE
      JE=0
      CALL ELBOUNDS (XC,YC,ZC,JSUBER,JE,IWHIPART)
      GO TO 130
  100 CONTINUE
      JE=1
      CALL ELBOUNDS (XTL,YTL,ZTL,JSUBER,JE,IWHIPART)
      GO TO 130
  110 CONTINUE
      JE=0
      CALL ELBOUNDS (XC,YC,ZC,JSUBER,JE,IWHIPART)
      JE=1
```

```
      CALL ELBOUNDS (XTL,YTL,ZTL,JSUBER,JE,IWHIPART)
      GO TO 130
C     NO CYLINDER INTERSECTION
C
  120 CONTINUE
C
C     THE RETURN FROM ALGOR D (BACK IN SEGMENT 102)
C     WAS NOT RECOGNIZABLE AS EITHER CIRCLE OR ELLIPSE.
C
      PRINT 190, IOKTEST
      GO TO 140
C
  130 CONTINUE
      IF (JE.EQ.2HNO) GO TO 140
C
C     THE SET OF XS,YS POINTS IS PLOTTED.
C
      CALL PLOTSET
C
C     THE CURRENT CYLINDER HAS BEEN PROCESSED THRU
C     THE DISCRETE POINT DETERMINATIONS
C     NOW GO BACK TO SEGMENT 2 AND LOOK FOR
C     MORE CYLINDERS OR SPHERES THAT HAVE NOT BEEN PROCESSED
C
  140 CONTINUE
      ILOOP=ILOOP+1
      IF (ILOOP.EQ.IDSEND) 150,180
  150 IF (INCHOPE.EQ.4HONCE) 160,170
  160 CONTINUE
      IFILL7=9999
      CALL PLOTS (0,0)
      CALL PLOT (0,0,3)
      CALL PRCNTL
  170 CONTINUE
      CALL PLOT (0,0,3)
      CALL PLOTS (0,0)
      CALL ISHOPE
      ILOOP=0
      IDSEND=IDSEND-1
      CALL PRCNTL
  180 CONTINUE
      IFILL7=102
      CALL PRCNTL
C
  190 FORMAT (1X,*IOKTEST FAILURE*,/,1X,*UNABLE TO IDENTIFY EITHER CIRCL
     1E OR ELLIPSE*)
  200  FORMAT(1X,*TRUNCATION DETERMINATION TROUBLE*,
     11X,*TL = *,F10.7,
     21X,*D1Z = *,F10.7,
     31X,*D2Z = *,F10.7)
       END
```

```
                                       IDENT     DISCRETE
PROGRAM LENGTH             00613
ENTRY POINTS    DISCRETE   00052
BLOCK NAMES
                TIMETEST   00004
                SYSTEMZ    00175
                SECT1LOG   02642
                VOCABTBL   02114
                2          27531
                CROSSEC    03037
EXTERNAL SYMBOLS
                Q8QENTRY
                THEND,
                Q8QDICT,
                POINTSC
                POINTSE
                CONVERT
                CONVERT2
                ELBOUNDS
                PLOTSET
                PLOTS
                PLOT
                PRCNTL
                ISHOPE
                SINF
                COSF
                STH,
                QNSINGL,
   00603 SYMBOLS
```

11/02/70

```
      SUBROUTINE POINTSE
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),H0(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP
C
      REAL NUM
      REAL MAJOR,MINOR,MA,MI
C
C
C     THIS SUBROUTINE IS CALLED FROM ≠≠PROGRAM DISCRETE≠≠.
C
C     THE PURPOSE OF THIS SUBROUTINE IS TO COMPUTE ≠≠M≠≠ SETS,
C          (XS(J),YS(J)) WHERE J RUNS FROM 1 TO ≠≠M≠≠,
C     OF POINTS OF AN ELLIPSE WHOSE CENTER IS AT CX,CY
C     AND WITH THE DESIGNATED SEMI-MAJOR AND SEMI-MAJOR AXES
C
C     THESE ≠≠M≠≠ POINTS ARE IN THE FIRST QUADRANT,
C     TAKEN FROM 12:0≠CLOCK TO 3:0≠CLOCK.
C     THEY WILL SERVE AS A FEEDER SET
C     FROM WHICH A FULL SET OF 400 PAIRS WILL BE COMPUTED
C     IN SUBROUTINE EXPAND.
C
C     VARIABLES USED:
C     ≠≠INC≠≠, THE NUMBER OF INCREMENTS IN PI RADIANS
C     ≠≠DTH≠≠, THE RADIAN INCREMENT,
C     ≠≠TH≠≠, THE VALUE OF THE ANGLE THETA IN RADIANS,
C     ≠≠EC≠≠, THE ECCENTRICITY OF THE ELLIPSE,
C     ≠≠MAJOR≠≠, THE SEMI-MAJOR AXIS,
C     ≠≠MINOR≠≠, THE SEMI-MINOR AXIS,
C     ≠≠RA≠≠, A VECTOR FROM CENTER TO ITH POINT ON PERIMETER.
C
C
      ISET=400
      MA=MAJOR
      MI=MINOR
      M=(ISET-4)/4
      INC=(M-1)*2
      DTH=3.141592653589793/INC
      TH=(3.141592653589793/2)+DTH
      EC=SQRTF(1-((MINOR**2)/(MAJOR**2)))
      N=M+2
      NUM=(MINOR**2)
      A3=EC**2
      DO 10 J=1,N
      TH=TH-DTH
      A0=TH
      A1=COS(A0)
      A2=A1**2
      A4=A2*A3
      DEN=1-A4
      A6=(NUM/DEN)
      RA=SQRTF(A6)
      X(J)=RA*COSF(TH)
      Y(J)=RA*SINF(TH)
   10 CONTINUE
```

11/02/70

```
C                                                                POE 620
      CALL EXPAND (M)                                            POE 630
C                                                                POE 640
      RETURN                                                     POE 650
C                                                                POE 660
      END                                                        POE 670*
```

```
 POINTSE                                          11/02/70      ED      0
                                    IDENT      POINTSE
PROGRAM LENGTH                  00145
ENTRY POINTS    POINTSE         00003
BLOCK NAMES
                CROSSEC         02525
EXTERNAL SYMBOLS
                Q1Q05100
                Q8QDICT,
                EXPAND
                SQRTF
                SINF
                COSF
   00105 SYMBOLS
```

```
      SUBROUTINE POINTSC
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,ILOOP
C
C
C     THIS SUBROUTINE IS CALLED FROM ≠≠PROGRAM DISCRETE≠≠,
C
C
C     THE PURPOSE OF THIS SUBROUTINE IS TO COMPUTE ≠≠M≠≠ SETS,
C     (XS(J),YS(J)), J=1-≠≠M≠≠,
C     OF POINTS OF A CIRCLE IN THE HOPE PLANE CENTERED AT (0,0)
C     WITH THE DESIGNATED RADIUS
C
C
C     THESE ≠≠M≠≠ POINTS ARE IN THE FIRST QUADRANT,
C     TAKEN FROM 12:0≠CLOCK TO 3:0≠CLOC<.
C     THEY WILL SERVE AS A FEEDER SET
C     FROM WHICH A FULL SET OF 200 PAIRS WILL BE COMPUTED
C     IN SUBROUTINE EXPAND,
C
C
C     SHORTHAND FORM FOR RADIUS
C
      R=RADIUS
C
C
      ISET=200
      TH=3.1415927/2
      M=(ISET-4)/4
      INC=(M+1)*2
      DTH=3.1415927/INC
C
C
C     INITIAL VALUE FOR Y
C
C
C     FIRST VALUES OF FEEDER SET
C
      X(1)=0
      Y(1)=R
C
      N=M+1
      DO 10 J=2,N
      TH=TH-DTH
C
      X(J)=R*COSF(TH)
C
      Y(J)=R*SINF(TH)
C
C
   10 CONTINUE
      K=M+2
      X(K)=R
      Y(K)=0.0
C
```

UNCLASSIFIED

```
C                                                                    POC 570
      CALL EXPAND (M)                                                POC 580
C                                                                    POC 590
      RETURN                                                         POC 600
C                                                                    POC 610
      END                                                            POC 620-
```



```
                                     IDENT     POINTSC
PROGRAM LENGTH                 00105
ENTRY POINTS       POINTSC     00003
BLOCK NAMES
                   CROSSEC     02524
EXTERNAL SYMBOLS
                   Q1Q05100
                   Q8QDICT,
                   EXPAND
                   SINF
                   COSF
   00070 SYMBOLS
```

11/02/70

```
      SUBROUTINE EXPAND (M)
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP
C
C
C     THIS SUBROUTINE IS CALLED FROM EITHER SUBROUTINE POINTSC
C     OR FROM SUBROUTINE POINTSE,
C
C
C     THE PURPOSE OF THIS SUBROUTINE
C     IS TO TAKE THE BASIC 90 DEGREE, FIRST QUADRANT, SET OF POINTS
C     CONTAINED IN THE X AND Y ARRAYS
C     AND EXPAND THEM INTO A FULL SET OF ISET POINTS
C     IT WORKS FOR EITHER A CIRCLE FEEDER SET (PREPARED BY POINTSC)
C     OR FOR AN ELLIPSE FEEDER SET (PREPARED BY POINTSE).
C
C
C     SETTING UPPER AND LOWER QUADRANT LIMITS,
C
C
      IQ1U=M+2
      IQ2L=IQ1U+1
      IQ2U=IQ2L+M
      IQ3L=IQ2U+1
      IQ3U=IQ3L+M
      IQ4L=IQ3U+1
      IQ4U=IQ4L+(M-1)
C
C     FIRST QUADRANT,12:00 TO 3:00
C
      DO 10 J=1,IQ1U
      XS(J)=X(J)
      YS(J)=Y(J)
   10 CONTINUE
C
C
C     SECOND QUADRANT,3:00 TO 6:00
C
      N=2
C
      DO 20 J=IQ2L,IQ2U
C
      XS(J)=X(J-N)
C
      YS(J)=-Y(J-N)
C
      N=N+2
C
   20 CONTINUE
C
C
C     THIRD QUADRANT,6:00 TO 9:00
C
C
```

```
      N=IQ2U-1
      DO 30 J=IQ3L,IQ3U
C
      XS(J)=-X(J-N)
C
      YS(J)=-Y(J-N)
C
   30 CONTINUE
C
C
C     FOURTH QUADRANT,9:00 TO 12:00
      N=IQ3L
C
      DO 40 J=IQ4L,IQ4U
C
      XS(J)=-X(J-N)
C
      YS(J)=Y(J-N)
C
      N=N+2
C
   40 CONTINUE
C
      RETURN
      END
```



```
                                        IDENT     EXPAND
PROGRAM LENGTH                00201
ENTRY POINTS     EXPAND       00003
BLOCK NAMES
                 CROSSEC      02525
EXTERNAL SYMBOLS
                 Q8QDICT,
   00111 SYMBOLS
```

01/20/71

```
      SUBROUTINE PLOTSET
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(154),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TI,IL
     3OOP
C
C
C     THIS ROUTINE IS CALLED FROM  ##PROGRAM DISCRETE##.
C
C     THE PURPOSE OF THIS SUBROUTINE IS TO PLOT THE SET OF POINTS
C     IN THE XS AND YS ARRAYS WHICH ARE THE CURVE OF INTERSECTION
C     BETWEEN THE HOPE PLANE AND A CYLINDER.
C     THIS CURVE IS EITHER A CIRCLE, AN ELLIPSE, OR A TRUNCATED ELLIPSE.
C
C     MOVE THE PEN IN THE UP POSITION TO THE 1ST POINT
      PX=(XS(1)*SCALE)+5
      PY=(YS(1)*SCALE)+5
      IF ((PX.LT.0).OR.(PX.GT.10.0).OR.(PY.LT.0).OR.(PY.GT.10.0)) 10,20
   10 PRINT 60, PX,PY
      RETURN
   20 CONTINUE
       PX1=PX
       PY1=PY
      CALL PLOT (PX,PY,3)
C     LOWER THE PEN
      CALL PLOT (PX,PY,2)
      J=1
      DO 50 J=2,ISET
      PX=(XS(J)*SCALE)+5
      PY=(YS(J)*SCALE)+5
      IF ((PX.LT.0).OR.(PX.GT.10.0).OR.(PY.LT.0).OR.(PY.GT.10.0)) 30,40
C
C
C     PLEASE NOTE...
C     THERE IS AN INTENTIONAL INCONSISTENCY HERE.
C     THE ELLIPSE AND/OR CIRCLE IS NOT BEING SUBJECTED TO
C     HOXYMIN AND HOXYMAX LIMITS IN CROSEC (MOD 2.0),
C     AS ARE PLANE INTERSECTIONS.
C     ONLY TO A 10 INCH SQUARE CHECK.
C     THE REASONS FOR THIS CHOICE WERE:
C     (1) TO ALLOW THE FULL ELLIPSES TO BE PLOTTED IF POSSIBLE FOR
C         REASONS OF SHOWMANSHIP.
C         (##THEY REALLY CAN BE COMPUTED AND PLOTTED--
C         SEE, HERE THEY ARE.##)
C     (2) SINCE THERE WILL PROBABLY BE FEWER CIRCULAR AND ELLIPTICAL
C     INTERSECTIONS THAN PLANE INTERSECTIONS, ALLOWING THIS EXCEPTION,
C     AT LEAST INITIALLY, SHOULD NOT DESTROY THE SIMPLICITY
C     AND INTERPRETABILITY OF THE PLOT.
C     (3) THE ELBOUNDS EFFECTS ARE PERHAPS RESTRICTION ENOUGH
C     ON THE ELLIPSES SINCE THEY ARE POTENTIALLY CONTROLABLE BY
C     THE SPECIFICATION OF THE CYLINDER LENGTH.
C     A FURTHER BOUNDARY CHANGE IS LEFT TO LATER
C     CROSEC MODIFICATIONS.
C
C
   30 CALL PLOT (PX,PY,3)
```

01/20/71

```
      GO TO 50                                                          PLS 570
   40 CALL PLOT (PX,PY,2)                                               PLS 580
   50 CONTINUE                                                          PLS 590
        CALL PLOT(PX1,PY1,2)                                            PLS 591
        CALL PLOT(PX1,PY1,3)                                            PLS 592
C                                                                       PLS 600
C     RAISE THE PEN AT THE LAST POSITION                                PLS 610
      CALL PLOT (PX,PY,3)                                               PLS 620
      RETURN                                                            PLS 630
C                                                                       PLS 640
   60 FORMAT (1X,*FIRST POINT PLOT OVERFLOW, X , Y *,F10.5,1X,F10.5)    PLS 650
      END                                                               PLS 660
```

```
 PLOTSET                                          01/20/71       ED      0

                                 IDENT       PLOTSET
PROGRAM LENGTH             00165
ENTRY POINTS     PLOTSET   00017
BLOCK NAMES
                 CROSSEC   02525
EXTERNAL SYMBOLS
                 THEND.
                 QBQDICT.
                 PLOT
                 STH.
                 QNSINGL.
   00104 SYMBOLS
```

12/23/70

```
      SUBROUTINE ELBOUNDS (F1,F2,F3,JSUBER,JE,IWHIPART)
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),H0(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP,IP,KR(200),IONCE
C
C
C     THIS SUBROUTINE IS CALLED FROM ≠≠DISCRETE≠≠,
C     ITS PURPOSE IS TO COMPUTE THE COORDINATES OF THE CLIP LINE
C     AND HAVING DONE THAT TO CALL ON ≠≠CLIP≠≠ TO PERFORM THE CLIPPING,
C
C     THE CLIP LINE IS COMPUTED BY,
C          (1) GETTING THE LINE OF INTERSECTION BETWEEN A BASE PLANE
C          AND THE HOPE PLANE,
C          (2) INTERSECTING THIS LINE WITH THE ELLIPSE
C          TO GET TWO POINTS (X1,Y1) AND (X2,Y2),
C          (WITH END PLAINS THAT ARE PERPENDICULAR TO THE CYLINDER≠S
C          AXIS IT FOLLOWS THAT THE CLIP LINE IS PARALLEL TO THE MINOR
C          AXIS OF THE ELLIPSE),
C
C     THE PARAMETERS ARE
C     ≠≠F1,F2,F3≠≠, AN END PLANE AXIS POINT,
C     ≠≠JSUBER≠≠, AN ERROR DETECTION FLAG FOR SUBROUTINE≠≠QUAD≠≠,
C     ≠≠JE≠≠, ENTERS WITH END PLANE IDENTIFICATION-
C                                              KEY AS TO WHICH BASE PLANE IS
C                                             BEING PROCESSED
C                                              JE=0 IS BOTTOM BASE PLANE
C                                               JE=1  IS TOP BASE PLANE
C               RETURNS WITH HOLLARITH YES OR NO MESSAGE
C               REGARDING SUSSESSFUL CLIPPING,
C     ≠≠IWHIPART≠≠, INFORMATION NEEDED BY ≠≠CLIP≠≠,
C
C
      IPLANE=JE
      X0=D*A
      Y0=D*B
      Z0=D*C
      E11=H0(1)
      E12=H0(2)
      E13=H0(3)
      E21=H0(4)
      E22=H0(5)
      E23=H0(6)
      E31=H0(7)
      E32=H0(8)
      E33=H0(9)
C
C
C     ASSUME CYLINDER BOUND PLANE HAS EQUATION IN X,Y,Z SYSTEM
C
C     UX * X + UY * Y + UZ * Z = UX * F1 + UY * F2 + UZ * F3
C
C     WHERE
C          UX,UY,UZ  ARE THE CYLINDER AXIS DIRECTION COSINES
C
C     AND  F1,F2,F3  IS A POINT ON THE AXIS
```

UNCLASSIFIED

```
C
C
C     OBTAINING LINE OF INTERSECTION BETWEEN HOPE PLANE AND END PLAIN,
C
      P=A=LX
      Q=B=LY
      R=C=LZ
      S=D=(LX*F1+LY*F2+LZ*F3)
      T11=XMAT9(1)
      T21=XMAT9(2)
      T31=XMAT9(3)
      T12=XMAT9(5)
      T22=XMAT9(6)
      T32=XMAT9(7)
      T13=XMAT9(9)
      T23=XMAT9(10)
      T33=XMAT9(11)
      P1=P*T11+Q*T21+R*T31
      Q1=P*T12+Q*T22+R*T32
      R1=P*T13+Q*T23+R*T33
      S1=S-(X0*P+Y0*Q+Z0*R)
      P=P1
      Q=Q1
      R=R1
      S=S1
C
C
C     TEST TO DETERMINE IF TRUNCATION LINE IS HORIZONTAL
C
      IF ((P.LT.0.00001).OR.(ABSF(PHI+1.57079633).LT.0.00001).OR.(ABSF(P
     1HI-1.57079633).LT.0.00001)) 110,140
C
C     TRUNCATION LINE HORIZONTAL
C
  110 Y1=S/Q
      Y2=Y1
      AA=E11
      BB=E12*Y1+E13
      CC=E22*Y1*Y1+2*E23*Y1+E33
      CALL QUAD (AA,BB,CC,X1,X2)
      IF (JSUEER.EQ.1050) 150,120
  120 IF (JSUEER.EQ.1051) 170,130
  130 IF (X1.EQ.X2) 210,200
C
C
C     TURUNCATION LINE NOT HORIZONTAL
C
  140 CONTINUE
C     OBTAINING INTERSECTION OF ≠≠LINE≠≠ WITH ELLIPSE
C     TO GET CLIP POINTS.
C
      SL=(-Q/P)
      TE=S/P
      AA=(E11*SL*SL+2*E12*SL+E22)
      BB=(E11*SL*TE)+(E12*TE)+(E13*SL)+E23
      CC=(E11*TE*TE+E33+2*E13*TE)
```

```
      JSUBERR=0
      CALL QUAD (AA,BB,CC,Y1,Y2)
      IF (JSUBERR.EQ.1050) 150,160
  150 CONTINUE
      JE=2HNO
      RETURN
  160 IF (JSUBERR.EQ.1051) 170,180
  170 CONTINUE
      JE=2HNO
      RETURN
  180 IF (Y1.EQ.Y2) 210,190
  190 CONTINUE
      X1=SL*Y1+TE
      X2=SL*Y2+TE
  200 CALL CLIP (X1,Y1,X2,Y2,IPLANE,JE,IWHIPART)
      RETURN
C     SINGLE SOLUTION
  210 CONTINUE
      JE=3HYES
      RETURN
      END
```



```
                                         IDENT     ELBOUNDS
PROGRAM LENGTH                  00456
ENTRY POINTS      ELBOUNDS      00006
BLOCK NAMES
                  CROSSEC       03037
EXTERNAL SYMBOLS
                  Q8QDICT,
                  QUAD
                  CLIP
   00202 SYMBOLS
```

01/07/71

```
      SUBROUTINE CLIP (X1,Y1,X2,Y2,IPLANE,JE,IWHCPART)
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),HO(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP,IP,KR(200),IONCE
C
C
C     THE SUBROUTINE IS CALLED FROM  ≠≠ELBOUNDS≠≠
C     IT≠S PURPOSE IS TO FORM A SUB=SET OF POINTS,
C     THAT DEFINE A TRUNCATED ELLIPSE,
C     FROM THE SET OF POINTS DEFINING THE ENTIRE ELLISPE
C     CONTAINED IN THE XS, YS ARRAYS.
C
C     THE TRUNCATION LINE RUNS FROM (X1,Y1) TO (X2,Y2),
C     BOTH OF THESE POINTS ARE ON THE PERIMETER OF THE ELLIPSE,
C     BUT MAY NOT BE IN THE XS, YS SET OF POINTS,
C     THE RELATIVE POSITION OF (X1,Y1),AND (X2,Y2) BETWEEN POINTS OF
C     THE XS,YS SET WILL BE DETERMINED BY TESTING,
C
C     ≠≠IPLANE≠≠, IS THE KEY AS TO WHICH END PLANE IS BEING PROCESSED,
C     WHEN IPLANE=1 THE TOP PLANE IS BEING PROCESSED, AND
C     WHEN IPLANE=0 THE BASE PLANE IS BEING PROCESSED,
C
C     ≠≠IWHCPART≠≠ BRINGS IN A HOLLERITH MESSAGE
C     WHICH DESCRIBES THE ORIENTATION OF THE ELLIPSE.
C     IF IWHCPART = 7HPOSDOWN
C     IT MEANS THAT THE ELLIPSE IS SLANTED WITH THE POSITIVE END OF THE
C     MAJOR AXIS DIRECTED DOWNWARD W.R.T. THE CYLINDER≠S Z AXIS,
C     IF IWHCPART=7HNEGDOWN THE OPPOSITE IS TRUE, THE NEGATIVE
C     END OF THE MAJOR AXIS IS LOWER THAN THE POSITIVE END W.R.T.
C     THE CYLINDER≠S AXIS,
C
C     ≠≠JE≠≠, RETURNS A HOLLARITH YES OR NO MESSAGE,
C     A ≠≠NO≠≠ MESSAGE MEANS CLIPPING
C     HAS NOT BEEN PERFORMED BECAUSE EITHER
C           (X1,Y1) AND (X2,Y2) DID NOT MEET THE PROPER CIRTERIA, OR
C           IWHCPART WAS NOT PROPERLY IDENTIFABLE.
C
C     LOCAL ARRAYS FOR INTERMEDIATE PURPOSES
C
      DIMENSION XSS(400), YSS(400)
C
C     LOCAL FUNCTIONS FOR ROTATION
C
      XNEW(P,Q,PPHI)=(P*COS(PPHI)+Q*SIN(PPHI))
      YNEW(P,Q,PPHI)=(-P*SIN(PPHI)+Q*COS(PPHI))
C
C
C     IF THE ANGLE OF ROTATION IS NOT ZERO THEN
C     THE XS,YS SET OF POINTS MUST BE TRANSLATED AND ROTATED TO AN
C     ORIGIN CENTERED, ZERO DEGREE POSITION,
C
C     IF PHI IS ZERO TRANSLATION IS NOT NEEDED.
C
C
      IF (PHI.NE.0) 110,130
```

01/07/71

```
  110 CONTINUE                                                          CLP 530
C                  ROTATE SET OF POINTS TO ZERO DEGREES TEMPORARILY     CLP 540
C                                                                       CLP 541
      DO 120 I=1,ISET                                                   CLP 550
      XI=XS(I)-CX                                                       CLP 560
      YI=YS(I)-CY                                                       CLP 570
      XIN=XNEW(XI,YI,-PHI)                                              CLP 580
      YIN=YNEW(XI,YI,-PHI)                                              CLP 590
      XS(I)=XIN                                                         CLP 600
      YS(I)=YIN                                                         CLP 610
  120 CONTINUE                                                          CLP 620
C                                                                       CLP 630
C                                                                       CLP 640
C     THE (X1,Y1) AND (X2,Y2) POINTS MUST BE HANDLED IN THE             CLP 650
C     SAME MANNER,                                                      CLP 660
C                                                                       CLP 670
C                                                                       CLP 680
       XX1=X1                                                           CLP 681
       YY1=Y1                                                           CLP 682
       XX2=X2                                                           CLP 683
       YY2=Y2                                                           CLP 684
      XI=X1-CX                                                          CLP 690
      YI=Y1-CY                                                          CLP 700
      XIN=XNEW(XI,YI,-PHI)                                              CLP 710
      YIN=YNEW(XI,YI,-PHI)                                              CLP 720
      X1=XIN                                                            CLP 730
      Y1=YIN                                                            CLP 740
      XI=X2-CX                                                          CLP 750
      YI=Y2-CY                                                          CLP 760
      XIN=XNEW(XI,YI,-PHI)                                              CLP 770
      YIN=YNEW(XI,YI,-PHI)                                              CLP 780
      X2=XIN                                                            CLP 790
      Y2=YIN                                                            CLP 800
  130 CONTINUE                                                          CLP 810
      M=(ISET-4)/4                                                      CLP 820
      ISUBSET=1                                                         CLP 830
C                                                                       CLP 840
C                                                                       CLP 850
C     THE TWO POINTS MUST BE SO ORIENTED THAT THE FIRST ONE IS          CLP 860
C     MORE POSITIVE THAN THE SECOND ONE,  THE TWO POINTS ARE            CLP 870
C     SWAPPED WITH EACH OTHER IF NECESSARY TO FULLFILL THIS CONDITION,  CLP 880
C                                                                       CLP 890
C                                                                       CLP 900
      IF (Y1,LT,Y2) 140,150                                             CLP 910
  140 CALL REVERSE (X1,Y1,X2,Y2)                                        CLP 920
  150 CONTINUE                                                          CLP 930
C                                                                       CLP 940
C                                                                       CLP 950
C     A MAJOR BRANCHING TAKES PLANE BASED ON THE SIGN OF X1 AND X2,     CLP 960
C     IF THEY ARE NOT OF THE SAME SIGN SOMFTHING IS WRONG,              CLP 970
C     AND A ≠≠NO≠≠ MESSAGE IS RETURNED IN ≠≠JE≠≠,                       CLP 980
C                                                                       CLP 990
C                                                                       CLP1000
      IF ((X1,GE,0),AND,(X2,GE,0)) 200,160                              CLP1010
  160 IF ((X1,LT,0),AND,(X2,LT,0)) 410,170                              CLP1020
  170 IF (((ABSF(X1),LT,0,0001),AND,(ABSF(Y1-1,0),LT,0,0001)),AND,((ABSFCLP1030
```

```
     1(X2).LT.0.0001).AND.(ABSF(Y2+1.0).LT.0.0001))        .OR.
     1    ((ABSF(X1).LT.0.0001).AND.(ABSF(X2).LT.0.0001)) )180,190
  180 X1=0.0
      X2=0.0
      GO TO 200
  190 PRINT 680, X1,Y1,X2,Y2
      GO TO 670
  200 CONTINUE
C
C
C     THE SWITCH **ISW** IS SET TO CONTROL THE TRANSFER
C     OR NON-TRANSFER OF POINTS FROM THE SET TO THE SUBSET.
C
C     IF ISW=1 THE TRANSFER IS TO TAKE PLACE.
C
C     THREE VARIABLES ARE INVOLVED IN THE DETERMINATION OF THE
C     INITIAL SETTING OF ISW.  THESE ARE THE SIGN OF X1 AND X2,
C     THE VALUE OF IPLANE
C     AND THE IWHCPART ORIENTATION.
C           (SEE THE TABLE GIVEN IN THE SUBROUTINE WRITEUP)
C
C
      IF (IWHCPART.EQ.7HNEGDOWN) 210,240
  210 IF (IPLANE) 220,230
  220 ISW=1
      GO TO 290
  230 ISW=0
      GO TO 300
  240 IF (IWHCPART.EQ.7HPOSDOWN) 250,280
  250 IF (IPLANE) 260,270
  260 ISW=0
      GO TO 290
  270 ISW=1
      GO TO 300
C
C
C     A CLIP MESSAGE IS PRINTED IDENTIFYING THE PLANE
C     AND THE CLIP POINTS.
C
C
  280 PRINT 690
      GO TO 670
  290  PRINT 700,XX1,YY1,XX2,YY2
      GO TO 310
  300  PRINT 710,XX1,YY1,XX2,YY2
  310 CONTINUE
C
C
C     IN THIS LOOP THE RELATIVE POSITION OF THE FIRST POINT IS
C     INITIALLY SEARCHED FOR SINCE THE X1 SIGN IS POSITIVE OR ZERO.
C
C     WHEN (X1,Y1) IS FOUND RELATIVE TO THE XS,YS SET
C     IT IS TRANSFERRED TO THE SUBSET AND ISW IS CHANGED.
C     AND X1 FLAGGED WITH 99.99.
C
C     THEN THE SEARCH FOR THE RELATIVE POSITION OF (X2,Y2) COMMENCES.
```

01/07/71

```
C                                                                       CLP1590
C     WHEN THE SECOND POINT IS FOUND RELATIVE TO THE XS,YS SET A TEST ISCLP1600
C     NECESSARY TO DETERMINE IF THE PROCESS IS COMPLETE OR IF THE       CLP1610
C     REMAINDER OF XS,YS MUST BE PASSED.                                CLP1620
C                                                                       CLP1630
C                                                                       CLP1640
      DO 400 I=1,ISET                                                   CLP1650
      IF ((X1.EQ.99.99).AND.(X2.EQ.99.99).AND.(ISW.EQ.1)) GO TO 390     CLP1660
      IF ((((ABSF(XS(I)-X1).LE.0.0001).OR.(XS(I).GT.X1)).AND.((ABSF(Y1-YCLP1670
     1S(I)).LE.0.0001).OR.(YS(I).LT.Y1))).OR.(((ABSF(XS(I)-X2).LE.0.0001CLP1680
     2).OR.(XS(I).LT.X2)).AND.((ABSF(YS(I)-Y2).LE.0.0001).OR.(YS(I).LT.YCLP1690
     32)))) 320,380                                                     CLP1700
  320 CONTINUE                                                          CLP1710
      IF (X1.EQ.99.99) 360,330                                          CLP1720
  330 CONTINUE                                                          CLP1730
      XSS(ISUBSET)=X1                                                   CLP1740
      YSS(ISUBSET)=Y1                                                   CLP1750
      X1=99.99                                                          CLP1760
      ISUBSET=ISUBSET+1                                                 CLP1770
      IF (ISW) 350,340                                                  CLP1780
  340 ISW=1                                                             CLP1790
      GO TO 380                                                         CLP1800
  350 ISW=0                                                             CLP1810
      GO TO 380                                                         CLP1820
  360 CONTINUE                                                          CLP1830
      XSS(ISUBSET)=X2                                                   CLP1840
      YSS(ISUBSET)=Y2                                                   CLP1850
      ISUBSET=ISUBSET+1                                                 CLP1860
      X2=99.99                                                          CLP1870
      IF (((IWHCPART.EQ.7HNEGDOWN).AND.(IPLANE.EQ.1)).OR.((IWHCPART.EQ.7CLP1880
     1HPOSDOWN).AND.(IPLANE.EQ.0))) 370,610                             CLP1890
  370 ISW=1                                                             CLP1900
      GO TO 380                                                         CLP1910
  380 CONTINUE                                                          CLP1920
      IF (ISW.EQ.1) 390,400                                             CLP1930
  390 CONTINUE                                                          CLP1940
C                                                                       CLP1941
C     PASS INCLUSION AREA POINT TO SUBSET                               CLP1950
C                                                                       CLP1951
      XSS(ISUBSET)=XS(I)                                                CLP1960
      YSS(ISUBSET)=YS(I)                                                CLP1970
      ISUBSET=ISUBSET+1                                                 CLP1980
  400 CONTINUE                                                          CLP1990
      ISUBSET=ISUBSET-1                                                 CLP2000
      GO TO 610                                                         CLP2010
  410 CONTINUE                                                          CLP2020
      IF (IWHCPART.EQ.7HNEGDOWN) 420,450                                CLP2030
  420 IF (IPLANE) 430,440                                               CLP2040
  430 ISW=0                                                             CLP2050
      GO TO 490                                                         CLP2060
  440 ISW=1                                                             CLP2070
      GO TO 500                                                         CLP2080
  450 IF (IWHCPART.EQ.7HPOSDOWN) 460,280                                CLP2090
  460 IF (IPLANE) 470,480                                               CLP2100
  470 ISW=1                                                             CLP2110
      GO TO 490                                                         CLP2120
```

```
  480 ISW=0
      GO TO 500
  490 PRINT 700, X1,Y1,X2,Y2
      GO TO 510
  500 PRINT 710, X1,Y1,X2,Y2
      GO TO 510
  510 CONTINUE
C
C
C     IN THIS LOOP X1 AND X2 ARE NEGATIVE AND THE RELATIVE POSITION OF
C     (X2,Y2) IN THE XS,YS SET IS SEARCHED FOR FIRST,
C
C     WHEN (X2,Y2)'S POSITION IS FOUND IT IS TRANSFERRED TO THE
C     SUBSET, THE ISW SWITCH IS CHANGED AND X2 FLAGGED WITH -99,99,
C
C     THE SEARCH FOR THE RELATIVE POSITION OF (X1,Y1) THEN COMMENCES,
C
C     WHEN IT IS FOUND IT IS PASSED TO THE SUBSET AND A TEST MADE TO
C     DETERMINE IF THE REST OF THE XS,YS ARRAY SHOULD BE PASSED,
C
C
      DO 600 I=1,ISET
      IF ((X2,EQ,-99,99),AND,(X1,EQ,99,99),AND,(ISW,EQ,1)) GO TO 590
      IF ((((ABSF(XS(I)-X2),LE,0,0001),OR,(XS(I),LT,X2)),AND,((ABSF(YS(I
     1)-Y2),LE,0,0001),OR,(YS(I),GT,Y2))),OR,((X2,EQ,-99,99),AND,((ABSF(
     2XS(I)-X1),LE,0,0001),OR,(XS(I),GT,X1)),AND,((ABSF(YS(I)-Y1),LE,0,0
     3001),OR,(YS(I),GT,Y1)))) 520,580
  520 IF (X2,EQ,-99,99) 530,550
  530 XSS(ISUBSET)=X1
      YSS(ISUBSET)=Y1
      X1=99,99
      ISUBSET=ISUBSET+1
      IF (((IWHCPART,EQ,7HNEGDOWN),AND,(IPLANE,EQ,0)),OR,((IWHCPART,EQ,7
     1HPOSDOWN),AND,(IPLANE,EQ,1))) 540,610
  540 ISW=1
      GO TO 580
  550 CONTINUE
      XSS(ISUBSET)=X2
      YSS(ISUBSET)=Y2
      X2=-99,99
      ISUBSET=ISUBSET+1
      IF (ISW) 560,570
  560 ISW=0
      GO TO 580
  570 ISW=1
      GO TO 580
  580 IF (ISW) 590,600
  590 CONTINUE
      XSS(ISUBSET)=XS(I)
      YSS(ISUBSET)=YS(I)
      ISUBSET=ISUBSET+1
  600 CONTINUE
      ISUBSET=ISUBSET-1
  610 CONTINUE
      XSS(ISUBSET)=XSS(1)
      YSS(ISUBSET)=YSS(1)
```

```
C      SWAP                                                              CLP2690
C                                                                        CLP2700
C                                                                        CLP2710
C      THE TERMINATION PROCESS COMMENCES BY ZEROING THE XS,YS ARRAY.     CLP2720
C                                                                        CLP2730
C                                                                        CLP2740
       DO 620 J=1,ISET                                                   CLP2750
       XS(J)=0                                                           CLP2760
       YS(J)=0                                                           CLP2770
  620 CONTINUE                                                           CLP2780
C                                                                        CLP2790
C                                                                        CLP2800
C      THE SUBSET VALUES ARE TRANSFERRED TO THE XS,YS ARRAY AND THE      CLP2810
C      ISET COUNTER RESET TO THE NEW VALUE OF ISUBSET.                   CLP2820
C                                                                        CLP2830
C                                                                        CLP2840
       DO 630 K=1,ISUBSET                                                CLP2850
       XS(K)=XSS(K)                                                      CLP2860
       YS(K)=YSS(K)                                                      CLP2870
       XSS(K)=0                                                          CLP2880
       YSS(K)=0                                                          CLP2890
  630 CONTINUE                                                           CLP2900
       ISET=ISUBSET                                                      CLP2910
       JE=3HYES                                                          CLP2920
C                                                                        CLP2930
C                                                                        CLP2940
C      THE ROTATION AND TRANSLATION IS PUT BACK IN IF NECESSARY.         CLP2950
C                                                                        CLP2960
C                                                                        CLP2970
       IF (PHI.NE.0) 640,660                                             CLP2980
  640 CONTINUE                                                           CLP2990
C                                                                        CLP2991
C                        PUT PHI ROTATION BACK INTO NEW SET OF POINTS    CLP3000
C                                                                        CLP3001
       DO 650 I=1,ISUBSET                                                CLP3010
       XIN=XNEW(XS(I),YS(I),PHI)                                         CLP3020
       YIN=YNEW(XS(I),YS(I),PHI)                                         CLP3030
       XS(I)=XIN+CX                                                      CLP3040
       YS(I)=YIN+CY                                                      CLP3050
  650 CONTINUE                                                           CLP3060
  660 CONTINUE                                                           CLP3070
       RETURN                                                            CLP3080
  670 CONTINUE                                                           CLP3090
       JE=2HNO                                                           CLP3100
       RETURN                                                            CLP3110
C                                                                        CLP3120
  680 FORMAT (1X,*CLIP REJECTION, ADJUSTED X VALUES W.R.T. ORIGIN*,/,1X,CLP3130
     1*MUST BOTH BE POSITIVE OR BOTH BE NEGATIVE*,/, 1X,*X1,Y1,X2,Y2 *,4CLP3140
     2(F10.5,1X))                                                        CLP3150
  690 FORMAT (1X,*NO CLIPPING PERFORMED, IWHCPART TROUBLE*)              CLP3160
  700 FORMAT (10X,*TOP PLANE CLIP*,1X,*FROM X,Y *,F7.2,*, *,F7.2,* TO X,CLP3170
     1Y *,F7.2,*, *,F7.2)                                                CLP3180
  710 FORMAT (10X,*BASE PLANE CLIP*,1X,*FROM X,Y *,F7.2,*, *,F7.2,* TO XCLP3190
     1,Y *,F7.2,*, *,F7.2)                                               CLP3200
       END                                                               CLP3210
```

```
    CLIP                                                                       01/07/71        ED

                                             IDENT      CLIP
PROGRAM LENGTH                    03247
ENTRY POINTS        CLIP          01572
BLOCK NAMES
                    CROSSEC       03037
EXTERNAL SYMBOLS
                    THEND,
                    Q8QDICT,
                    REVERSF
                    SINF
                    COSF
                    STH,
                    QNSINGL,
   00462 SYMBOLS
```

```
      SUBROUTINE CONVERT (P,Q,RR,XX,YY,ZZ)
      COMMON /CROSSEC/ A,B,C,D,IDSEND,XMAT9(16),ARRAY(254),XC,YC,ZC,UX,U
     1Y,UZ,R,SCALE,DCOSCYL(9),H0(9),XS(400),YS(400),X(103),Y(103),MAJOR,
     2MINOR,PHI,CX,CY,RADIUS,ISET,CXYZ(16),EXYZ(16),EH(16),IOKTEST,TL,IL
     3OOP,IP,KR(200),IONCE
C
C
C     THE PURPOSE OF THIS SUBROUTINE IS
C     PERFORM MATRIX CONVERSION FROM
C           (1) HOPE SYSTEM TO XYZ SYSTEM,
C           (2) XYZ SYSTEM TO CYLINDER SYSTEM
C               (HANDLED BY ENTRY CONVERT2)
C     THIS CONVERSION IS DONE FOR SINGLE POINTS WHERE
C     THE INPUT POINT IS (P,Q,RR), AND
C     THE OUTPUT POINT IS (XX,YY,ZZ),
C
C
C     PICKING UP DIRECTION COSINES,
C
      T11=XMAT9(1)
      T21=XMAT9(2)
      T31=XMAT9(3)
      T12=XMAT9(5)
      T22=XMAT9(6)
      T32=XMAT9(7)
      T13=XMAT9(9)
      T23=XMAT9(10)
      T33=XMAT9(11)
C
C     COMPUTING THE ORIGIN,
C
      XZERO=D*A
      YZERO=D*B
      ZZERO=D*C
C
C     THE CONVERSION EQUATIONS FOR ROTATION AND TRANSLATION,
C
      XX=T11*P+T12*Q+T13*RR+XZERO
      YY=T21*P+T22*Q+T23*RR+YZERO
      ZZ=T31*P+T32*Q+T33*RR+ZZERO
      RETURN
C
      ENTRY CONVERT2
C
C     PICKING UP DIRECTION COSINES,
C
      T11=DCOSCYL(1)
      T21=DCOSCYL(2)
      T31=DCOSCYL(3)
      T12=DCOSCYL(4)
      T22=DCOSCYL(5)
      T23=DCOSCYL(6)
      T13=DCOSCYL(7)
      T23=DCOSCYL(8)
      T33=DCOSCYL(9)
C
```

```
C     TRANSLATION,
C
      DX=XX-XC
      DY=YY-YC
      DZ=ZZ-ZC
C
C     THE CONVERSION EQUATIONS FOR ROTATION ON THE TRANSLATED POINTS.
C
      P=T11*DX+T21*DY+T31*DZ
      Q=T12*DX+T22*DY+T32*DZ
      RR=T13*DX+T23*DY+T33*DZ
      RETURN
      END
```

```
 CONVERT                                             11/02/70       ED      0

                                      IDENT       CONVERT
PROGRAM LENGTH                  00310
ENTRY POINTS      CONVERT       00003
                  CONVERT2      00077
BLOCK NAMES
                  CROSSEC       03037
EXTERNAL SYMBOLS
                  Q8QDICT,
   00135 SYMBOLS
```

11/02/70

```
      SUBROUTINE REVERSE (M1,N1,M2,N2)
      HOLD1=M2
      HOLD2=M1
      M1=HOLD1
      M2=HOLD2
      HOLD1=N2
      HOLD2=N1
      N1=HOLD1
      N2=HOLD2
      RETURN
      END
```

```
 REVERSE                                              11/02/70        ED       0

                                   IDENT       REVERSE
PROGRAM LENGTH                00131
ENTRY POINTS      REVERSE     00003
EXTERNAL SYMBOLS
                  Q1Q10100
                  Q8QDICT.
   00034 SYMBOLS
```

Appendix B

COMPUTER OUTPUT RESULTING FROM PART PROGRAM RUN WITH CROSEC (MOD 2.0)

This appendix contains the line printer output from the CalComp and the Gerber plotter* output of a computer run for four CROSEC plots on a production piece at NRL, a lower cover of a film magazine container.

A regular, non-computer-related drawing of the piece is shown in Fig. B1. The program deck contained definitions only: no tool motions. The part-programmer's origin was chosen in the center of the XY plane.

Figures B2 through B5 show the four plots to which typed labels have been added. Ordinarily CROSEC plots contain no lettering. The scale is shown in the lower left-hand corner. The distance between the vertical marks represents one unit. The HOPE origin, barely visible as a + symbol on the originals, has been touched up with a pen to show its location, since intersection lines were drawn through it by CROSEC.

Figures B2, B3 and B4 correspond to coordinate plane cuts through the piece and relationships to views of the drawing in Fig. B1 can be readily observed. Because of perpendicularity in these three views, projected lines correspond to intersected lines and recognition comes easily.

Figure B5 requires interpretation. The intersection lines do not necessarily correspond directly with projected lines. Planes that represent true surfaces in the part and have been cut by the HOPE plane have produced a true view of their intersection. It is beyond the present capabilities, of CROSEC to distinguish between these two situations.

A similar situation exists with cylinder intersections, they may or may not be a part of the cross section. In any event the ellipses help in locating the neighboring lines of intersection from certain planes. This subject of interpretation of CROSEC patterns is discussed in detail in the text of the report.

The elapsed time for the CalComp version shown on pp. 183-236 was 4 min and 9 sec and was the actual time spent by the computer doing its CROSEC job, which involved Section I interpretation of the data cards and treatment of four HOPE planes with a quick bypass of Sections II and III. There was of course, no post processor involved either. The "part program execution time" of 5 min and 5 sec shown on p. 235 represents the total elapsed time including interrupts, etc., and can be confusing. (The elapsed time for the Gerber version was 10 min 14 sec, and the Part Program execution time was 10 min 20 sec. Paper tape punching accounts for the increased time.)

The overlays and segment fills were done from magnetic tape and are obviously time consuming, especially when processing circle/cylinder intersections. Utilization of the drum would probably provide a time saving. The plot took 10 min off line.

This part program was made available through the kindness of Jay Williams, part programmer par excellence in the Engineering Services Division.

---

*A version of CROSEC (MOD 2.0) with paper tape output for the flatbed Gerber plotter.

UNCLASSIFIED

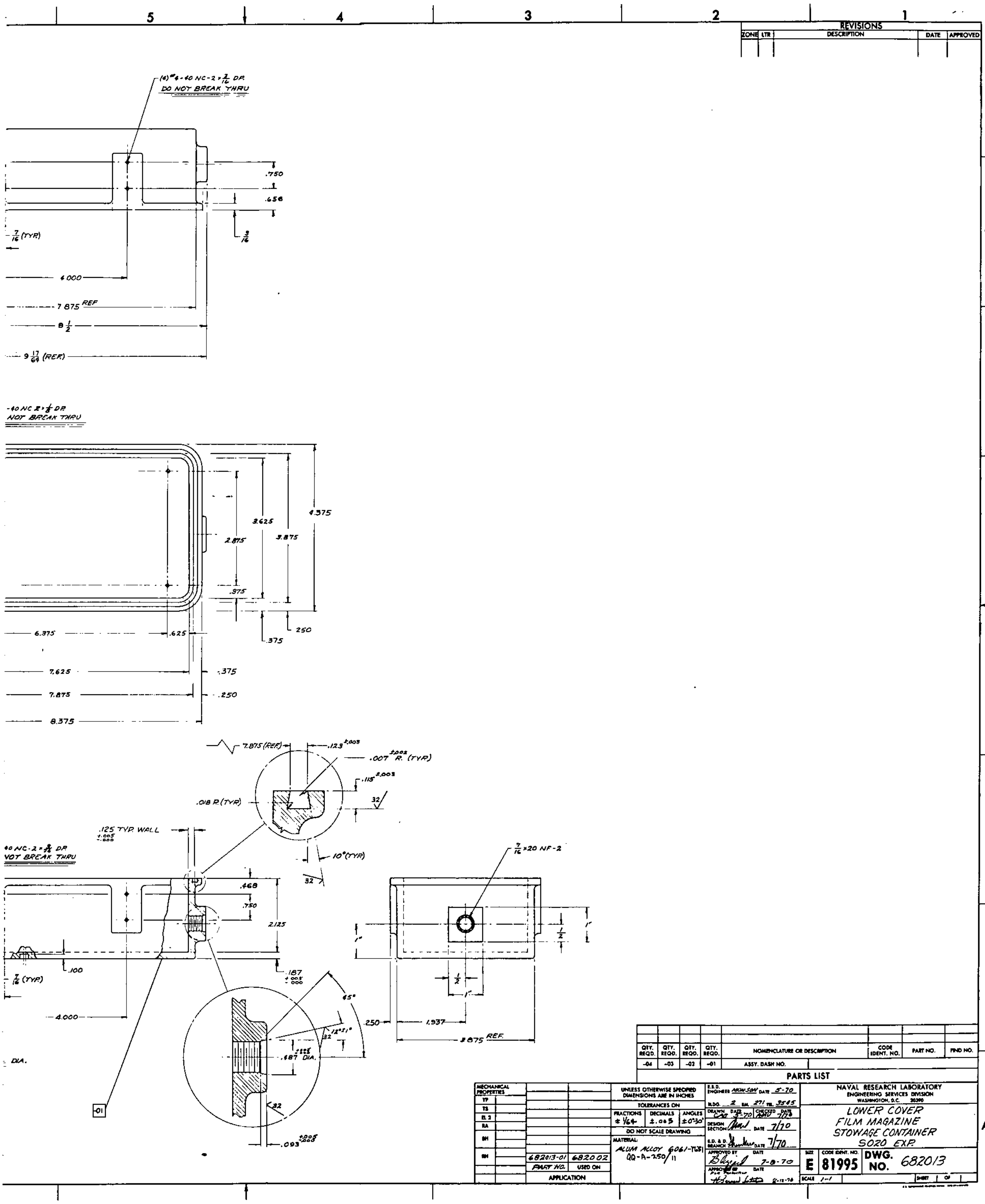
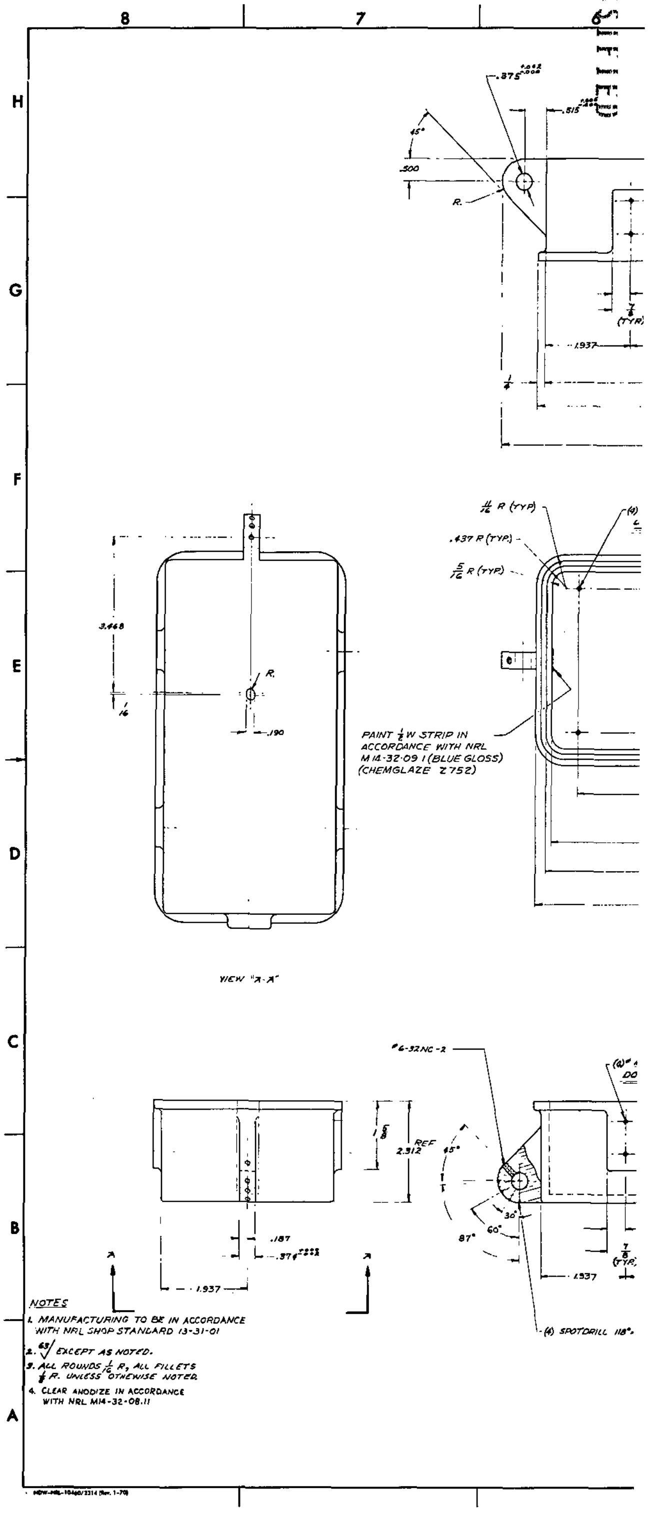


Figure B1


8
7
6
H
G
F
E
D
C
B
A
.875
.815
45°
.500
R.
1.937
3.468
R.
.190
VIEW "A-A"
PAINT ¼W STRIP IN ACCORDANCE WITH NRL M14-32-09.1 (BLUE GLOSS) (CHEMGLAZE Z752)
.437 R (TYP.)
#6-32NC-2
REF 2.312
45°
30°
60°
87°
.187
1.937
1.537
(4) SPOTDRILL 118°
A
A
NOTES
1. MANUFACTURING TO BE IN ACCORDANCE WITH NRL SHOP STANDARD 13-31-01
2. 63/ EXCEPT AS NOTED.
3. ALL ROUNDS 1/16 R, ALL FILLETS 1/8 R. UNLESS OTHERWISE NOTED.
4. CLEAR ANODIZE IN ACCORDANCE WITH NRL M14-32-08.11

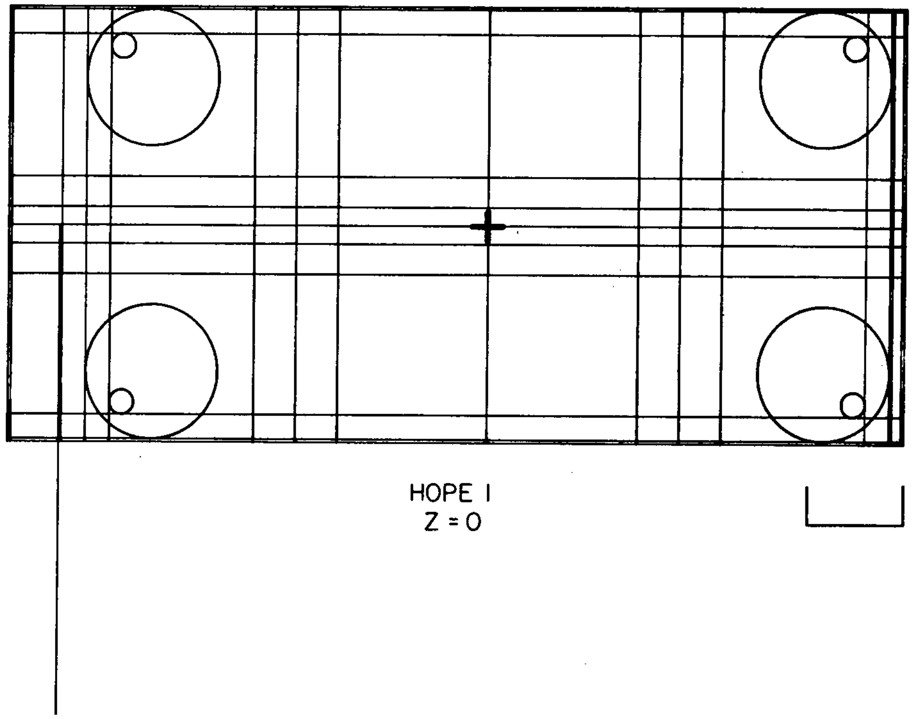


Figure B2

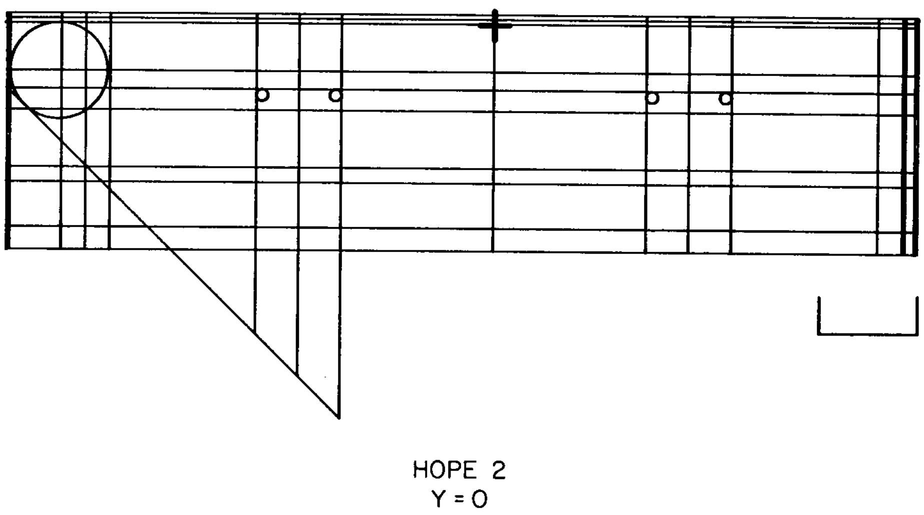


Figure B3

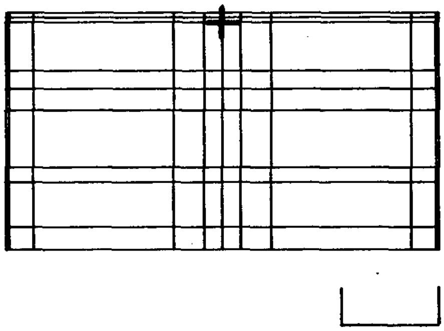


Figure B4

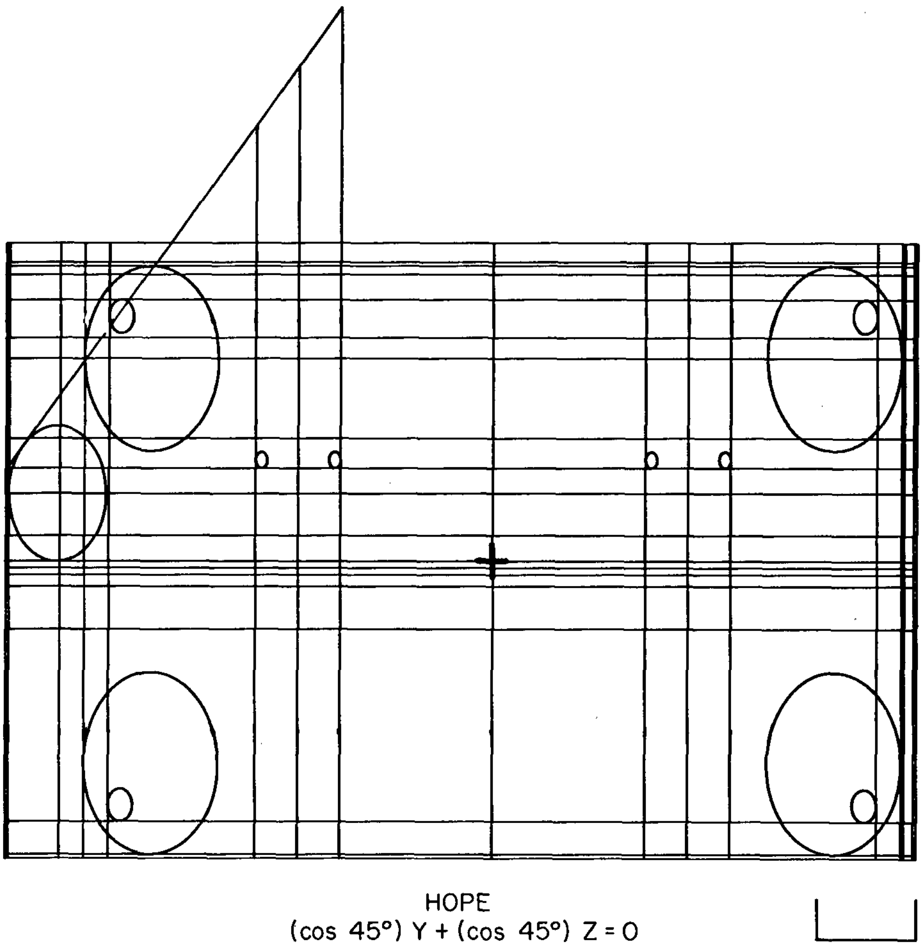


Figure B5

```
SEQUENCE 03775 STARTED  PRINTING 11/03/70 AT 200242 ON LP00
DRUM SCOPE 2.0 COMPUTER ONE,  MAX, DEMAND IS 54000B,  RESIDENT 2T.1    07/20/70
SEQUENCE NUMBER 403775 STARTED AT TIME  200219  DATED  11/03/70
JOB,23Z0001,573KPT,10
DEMAND,53730B
EQUIP,10=PL
EQUIP,49=(CROSEC CALCOM),HI,RO,DA
LOADMAIN,49,10,3500

EXECUTION STARTED AT   2002 .25
```

```
PARTNO     TAPE S-170  DWG 682013 LOWER COVER     J.A.W
         NOPOST
         MULTAX
 HOXYMIN=POINT/-05,-05,-05
  HOXYMAX=POINT/05,05,05
     SYN/P,POINT,L,LINE,E,PLANE,C,CIRCLE,GL,GOLFT,GR,GORGT,GF,GOFWD,GB,$
        GOBACK,TN,TLON,TL,TLLFT,TR,TLRGT,XS,XSMALL,XL,XLARGE,PA,PARLEL,$
        IT,INTOF,YL,YLARGE,YS,YSMALL   ,ZS,ZSMALL,ZL,ZLARGE,RA,RADIUS,$
        CE,CENTER,TA,TANTO,AA,ATANGL,PE,PERPTO,SD,GODLTA
  PL0    = E/0,0,1,,1
  PL1    = E/0,0,1,0
  PL2    = E/0,0,1,-.687
  PL3    = E/0,0,1,-2,125
  PL4    = E/0,0,1,-2,362
  PL5    = E/PA,PL1,ZS,1.5
  PL7A   0000140PL1,ZL,.05
  PL6    = E/0,0,1,-.5
  LX     = L/XAXIS
  LY     = L/YAXIS
  L1     = L/PA,LY,XS,4.1875
  L2     = L/PA,LY,XL,4.1875
  L3     = L/PA,L1,XL,,25
  L4     = L/PA,L3,XL,7.875
  L5     = L/PA,L3,XL,1.937
  L6     = L/PA,L5,XL,4.0
  L7     = L/PA,L5,XS,,437
  L8     = L/PA,L5,XL,,437
  L9     = L/PA,L6,XS,,437
  L10    = L/PA,L6,XL,,437
  L11    = L/PA,LX,YL,2.1875
  L12    = L/PA,LX,YS,2.1875
  L13    = L/PA,L12,YL,.25
  L14    = L/PA,L13,YL,3.875
  L11A   = L/PA,L11,YL,.03
  L12A   = L/PA,L12,YS,.03
  L1A    = L/PA,L1,XS,,25
  L2A    = L/PA,L2,XL,,040000008
  L16    = L/PA,L1,XL,8.5
  L17    = L/PA,LX,YL,,187
  L18    = L/PA,LX,YS,,187
  L19    = L/PA,LX,YL,,5
  L20    = L/PA,LX,YS,,5
  M1     = MATRIX/YZROT,90
           REFSYS/M1
  PL1A   = L/XAXIS
  PL1B   = L/YAXIS
  PL2A   = L/PA,PL1A,YS,,687
  PL3A   = L/PA,PL1A,YS,2,125
  PL5A   = L/PA,PL1B,XS,2
  PL6A   = L/PA,PL1B,XL,2
  L9A    =L/PA,PL6A,XS,,437
  L10A   = L/PA,PL6A,XL,,437
  L7A    = L/PA,PL5A,XS,,437
  L8A    = L/PA,PL5A,XL,,437
  L21A   = L/PA,PL1A,YS,,906
  L22A   = L/PA,L21A,YS,,75
  L15    = L/PA,PL1B,XS,4,962
  C9     = C/XL,L15,YS,PL1A,RA,,5
          ZSURF/(E/0,0,1,-,287)
  P9     = P/CE,C9
  PT     = P/C9,AA,225
```

```
LB      = L/PT,AA,-45                                   00006200
LC      = L/PA,PL5A,XS,1,937                            00006300
PTT     = P/IT,LB,LC                                    00006400
L21     = L/PTT,LEFT,TA,C9                              00006500
          ZSURF/(E/0,0,1,2,287)                         00006600
P1      = P/IT,PL5A,L21A                                00006700
P2      = P/IT,PL5A,L22A                                00006800
P3      = P/IT,PL6A,L22A                                00006900
P4      = P/IT,PL6A,L21A                                00007000
          ZSURF/(E/0,0,-1,2,287)                        00007100
P5      = P/IT,PL5A,L21A                                00007200
P6      = P/IT,PL5A,L22A                                00007300
P7      = P/IT,PL6A,L22A                                00007400
P8      = P/IT,PL6A,L21A                                00007500
P10     = P/4,4125,0,-1.0                               00007600
C10     = C/YS,PL2A,XL,L7A,RA,,06                       00007700
C11     = C/YS,PL2A,XS,L8A,RA,,06                       00007800
C14     = C/YS,PL2A,XL,L9A,RA,,06                       00007900
C15     = C/YS,PL2A,XS,L10A,RA,,06                      00008000
L15A    = L/PA,L15,XS,,03                               00008100
L16A    = L/PA,L16,XL,,03                               00008200
          REF SYS/NOMORE                                00008300
C1      = C/XS,L2,YL,L12,RA,,687                        00008400
C2      = C/XS,L2,YS,L11,RA,,687                        00008500
C3      = C/XL,L1,YS,L11,RA,,687                        00008600
C4      = C/XL,L1,YL,L12,RA,,687                        00008700
C5      = C/XL,L3,YL,L13,RA,,125                        00008800
C6      =C/XS,L4,YL,L13,RA,,125                         00008900
C7      = C/XS,L4,YS,L14,RA,,125                        00009000
C8      = C/XL,L3,YS,L14,RA,,125                        00009100
SPA     = P/0,3,5,2                                     00009200
SPB     = P/5,0,2                                       00009300
SPC     = P/0,-3,5,2                                    00009400
SPD     =P/-7,0,2                                       00009500
PT1     = P/-,9220,0,-,1                                00009600
PT2     = P/-,9845,0,-,1                                00009700
PT3     = P/-,9845,0,,1                                 00009800
HOPE1   = E/0,0,1,0
HOPE2   = E/0,1,0,0
HOPE3   = E/1,0,0,0
HOPE    = E/0,,70711,,70711,0
          PRINT/3,ALL                                   00009900
HOXYMI  (    0) POINT   3   -5,000000   -5,000000   -5,000000
HOXYMA  (    0) POINT   3    5,000000    5,000000    5,000000
PL0     (    0) PLANE   4    0,000000    0,000000    1,000000    0,100000
PL1     (    0) PLANE   4    0,000000    0,000000    1,000000    0,000000
PL2     (    0) PLANE   4    0,000000    0,000000    1,000000   -0,687000
PL3     (    0) PLANE   4    0,000000    0,000000    1,000000   -2,125000
PL4     (    0) PLANE   4    0,000000    0,000000    1,000000   -2,362000
PL5     (    0) PLANE   4    0,000000    0,000000   -1,000000    1,500000
PL7A    (    0) PLANE   4    0,000000    0,000000    1,000000    0,050000
PL6     (    0) PLANE   4    0,000000    0,000000    1,000000   -0,500000
LX      (    0) LINE    4    0,000000    1,000000    0,000000    0,000000
LY      (    0) LINE    4    1,000000    0,000000    0,000000    0,000000
L1      (    0) LINE    4   -1,000000    0,000000    0,000000    4,187500
L2      (    0) LINE    4    1,000000    0,000000    0,000000    4,187500
L3      (    0) LINE    4    1,000000    0,000000    0,000000   -3,937500
L4      (    0) LINE    4    1,000000    0,000000    0,000000    3,937500
L5      (    0) LINE    4    1,000000    0,000000    0,000000   -2,000500
L6      (    0) LINE    4    1,000000    0,000000    0,000000    1,999500
L7      (    0) LINE    4   -1,000000    0,000000    0,000000    2,437500
L8      (    0) LINE    4    1,000000    0,000000    0,000000   -1,563500
L9      (    0) LINE    4   -1,000000    0,000000    0,000000   -1,562500
L10     (    0) LINE    4    1,000000    0,000000    0,000000    2,436500
L11     (    0) LINE    4    0,000000    1,000000    0,000000    2,187500
L12     (    0) LINE    4    0,000000   -1,000000    0,000000    2,187500
```

.000000	1,000000	0,000000	-1,937500			
.000000	1,000000	0.000000	1,937500			
.000000	1,000000	0,000000	2,217500			
,000000	-1,000000	0.000000	2,217500			
.000000	0,000000	0,000000	4,437500			
.000000	0,000000	0.000000	4,217500			
000000	0,000000	0.000000	4,312500			
,000000	1,000000	0,000000	0,187000			
.000000	-1,000000	0.000000	0,187000			
000000	1,000000	0.000000	0,500000			
.000000	-1,000000	0,000000	0,500000			
.000000	0,000000	0,000000	0,000000	0,000000	0.000000	-1.000000
.000000	0,000000	1.000000	0,000000	0,000000		
.000000	0,000000	1.000000	0,000000			
,000000	0,000000	0.000000	0,000000			
.000000	-0,000000	-1.000000	0,687000			
.000000	-0,000000	-1.000000	2,125000			
.000000	0,000000	0,000000	2,000000			
000000	0,000000	0.000000	2,000000			
,000000	0,000000	0,000000	-1,563000			
.000000	0,000000	0.000000	2,437000			
000000	0,000000	0,000000	2,437000			
000000	0,000000	0,000000	-1,563000			
,000000	-0,000000	-1,000000	0,906000			
000000	-0,000000	-1,000000	1,656000			
.000000	0,000000	0,000000	4,962000			
,462000	-0,000000	-0.500000	0,000000	-1,000000	0.000000	0.500000
,462000	0,287000	-0.500000				
815553	0,287000	-0,853553				
,707107	0,000000	0.707107	-4,008664			
,000000	0,000000	0.000000	3,937000			
937000	0,287000	-1,732107				
,707107	0,000000	0.707107	-4,008664			
,000000	-2,287000	-0,906000				
,000000	-2,287000	-1,656000				
.000000	-2,287000	-1,656000				
.000000	-2,287000	-0,906000				
,000000	2,287000	-0,906000				
,000000	2,287000	-1,656000				
.000000	2,287000	-1,656000				
,000000	2,287000	-0,906000				
,412500	1,000000	-0.000000				
,377000	-0,000000	-0,747000	0,000000	-1,000000	0.000000	0.060000
.623000	-0,000000	-0,747000	0,000000	-1,000000	0.000000	0.060000
,623000	-0,000000	-0,747000	0,000000	-1,000000	0.000000	0.060000
,377000	-0,000000	-0.747000	0,000000	-1,000000	0,000000	0.060000
,000000	0,000000	0.000000	4,992000			
000000	0,000000	0,000000	4,342500			
,500500	-1,500500	0.000000	0,000000	0,000000	1,000000	0.687000
,500500	1,500500	0.000000	0,000000	0,000000	1,000000	0.687000
.500500	1,500500	0.000000	0,000000	0,000000	1,000000	0.687000
,500500	-1,500500	0.000000	0,000000	0,000000	1,000000	0.687000
812500	-1,812500	0.000000	0,000000	0,000000	1,000000	0.125000
,812500	-1,812500	0.000000	0,000000	0,000000	1,000000	0.125000
.812500	1,812500	0.000000	0,000000	0,000000	1,000000	0.125000

UNCLASSIFIED

UNCLASSIFIED

L13	(	0)	LINE	4	0
L14	(	0)	LINE	4	0
L11A	(	0)	LINE	4	0
L12A	(	0)	LINE	4	0
L1A	(	0)	LINE	4	-1
L2A	(	0)	LINE	4	1
L16	(	0)	LINE	4	1
L17	(	0)	LINE	4	0
L18	(	0)	LINE	4	0
L19	(	0)	LINE	4	0
L20	(	0)	LINE	4	0
M1	(	0)	MATRIX	12	1
					0
PL1A	(	0)	LINE	4	0
PL1B	(	0)	LINE	4	1
PL2A	(	0)	LINE	4	0
PL3A	(	0)	LINE	4	0
PL5A	(	0)	LINE	4	-1
PL6A	(	0)	LINE	4	1
L9A	(	0)	LINE	4	-1
L10A	(	0)	LINE	4	1
L7A	(	0)	LINE	4	-1
L8A	(	0)	LINE	4	1
L21A	(	0)	LINE	4	0
L22A	(	0)	LINE	4	0
L15	(	0)	LINE	4	-1
C9	(	0)	CIRCLE	7	-4
P9	(	0)	POINT	3	-4
PT	(	0)	POINT	3	-[illegible]
LB	(	0)	LINE	4	0
LC	(	0)	LINE	4	-1
PTT	(	0)	POINT	3	-[illegible]
L21	(	0)	LINE	4	0
P1	(	0)	POINT	3	-[illegible]
P2	(	0)	POINT	3	-[illegible]
P3	(	0)	POINT	3	[illegible]
P4	(	0)	POINT	3	[illegible]
P5	(	0)	POINT	3	-[illegible]
P6	(	0)	POINT	3	-[illegible]
P7	(	0)	POINT	3	[illegible]
P8	(	0)	POINT	3	[illegible]
P10	(	0)	POINT	3	[illegible]
C10	(	0)	CIRCLE	7	-[illegible]
C11	(	0)	CIRCLE	7	-1
C14	(	0)	CIRCLE	7	1
C15	(	0)	CIRCLE	7	[illegible]
L15A	(	0)	LINE	4	-1
L16A	(	0)	LINE	4	1
C1	(	0)	CIRCLE	7	[illegible]
C2	(	0)	CIRCLE	7	[illegible]
C3	(	0)	CIRCLE	7	-[illegible]
C4	(	0)	CIRCLE	7	-[illegible]
C5	(	0)	CIRCLE	7	-[illegible]
C6	(	0)	CIRCLE	7	[illegible]
C7	(	0)	CIRCLE	7	[illegible]

UNCLASSIFIED

000	1.812500	0.000000	0.000000	0.000000	1.000000	0.125000
000	3.500000	2.000000				
000	0.000000	2.000000				
000	-3.500000	2.000000				
000	0.000000	2.000000				
000	0.000000	-0.100000				
000	0.000000	-0.100000				
300	0.000000	0.100000				
000	0.000000	1.000000	0.000000			
000	1.000000	0.000000	0.000000			
000	0.000000	0.000000	0.000000			
000	0.707107	0.707107	0.000000			

00010000

```
C8       (     0) CIRCLE  7     -3.81

SPA      (     0) POINT   3      0.00
SPB      (     0) POINT   3      5.00
SPC      (     0) POINT   3      0.00
SPD      (     0) POINT   3     -7.00
PT1      (     0) POINT   3     -0.92
PT2      (     0) POINT   3     -0.98
PT3      (     0) POINT   3     -0.98
HOPE1    (     0) PLANE   4      0.00
HOPE2    (     0) PLANE   4      0.00
HOPE3    (     0) PLANE   4      1.00
HOPE     (     0) PLANE   4      0.00
                PRINT/0

                END

FINI
```

```
HOPE1
A,B,C,D 0.0000000 0.0000000 1.0000000 0.0000000
```

ANGLES BETWEEN HOPE1 AND THE DEFINED PLANES

PL0	PARALLEL TO REFERENCE PLANE
PL1	PARALLEL TO REFERENCE PLANE
PL2	PARALLEL TO REFERENCE PLANE
PL3	PARALLEL TO REFERENCE PLANE
PL4	PARALLEL TO REFERENCE PLANE
PL7A	PARALLEL TO REFERENCE PLANE
PL6	PARALLEL TO REFERENCE PLANE
LX	PERPENDICULAR TO REFERENCE PLANE
LY	PERPENDICULAR TO REFERENCE PLANE
L1	PERPENDICULAR TO REFERENCE PLANE
L2	PERPENDICULAR TO REFERENCE PLANE
L3	PERPENDICULAR TO REFERENCE PLANE
L4	PERPENDICULAR TO REFERENCE PLANE
L5	PERPENDICULAR TO REFERENCE PLANE

L6	PERPENDICULAR TO REFERENCE PLANE
L7	PERPENDICULAR TO REFERENCE PLANE
L8	PERPENDICULAR TO REFERENCE PLANE
L9	PERPENDICULAR TO REFERENCE PLANE
L10	PERPENDICULAR TO REFERENCE PLANE
L11	PERPENDICULAR TO REFERENCE PLANE
L12	PERPENDICULAR TO REFERENCE PLANE
L13	PERPENDICULAR TO REFERENCE PLANE
L14	PERPENDICULAR TO REFERENCE PLANE
L11A	PERPENDICULAR TO REFERENCE PLANE
L12A	PERPENDICULAR TO REFERENCE PLANE
L1A	PERPENDICULAR TO REFERENCE PLANE
L2A	PERPENDICULAR TO REFERENCE PLANE
L16	PERPENDICULAR TO REFERENCE PLANE
L17	PERPENDICULAR TO REFERENCE PLANE
L18	PERPENDICULAR TO REFERENCE PLANE
L19	PERPENDICULAR TO REFERENCE PLANE

L20	PERPENDICULAR TO REFERENCE PLANE
PL1A	PARALLEL TO REFERENCE PLANE
PL1B	PERPENDICULAR TO REFERENCE PLANE
PL5A	PERPENDICULAR TO REFERENCE PLANE
PL6A	PERPENDICULAR TO REFERENCE PLANE
L9A	PERPENDICULAR TO REFERENCE PLANE
L10A	PERPENDICULAR TO REFERENCE PLANE
L7A	PERPENDICULAR TO REFERENCE PLANE
L8A	PERPENDICULAR TO REFERENCE PLANE
L15	PERPENDICULAR TO REFERENCE PLANE
LB	AT 45.0 DEGREES TO REFERENCE PLANE
LC	PERPENDICULAR TO REFERENCE PLANE
L21	AT 45.0 DEGREES TO REFERENCE PLANE
L15A	PERPENDICULAR TO REFERENCE PLANE
L16A	PERPENDICULAR TO REFERENCE PLANE

```
IN ORIGINAL COORDINATE SYSTEM


    MINIMUM VALUES

            -5.00000
            -5.00000
            -5.00000

    MAXIMUM VALUES

             5.00000
             5.00000
             5.00000


IN HOPE PLANE

SCALE...   0.62500 INCHES OF PLOT LENGTH = 1 INCH OF PART LENGTH

TO OBTAIN A 1 TO 1 PLOT ON THE GERBER USE A SCALE FACTOR IN BOTH X AND Y OF 1.60000
```

HOPE1   INTERSECTIONS WITH ALLOWED DEFINED SURFACES

```
LINE OF INTERSECTION FOR PLANE LX
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   0.00

LINE OF INTERSECTION FOR PLANE LY
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    0,00

LINE OF INTERSECTION FOR PLANE L1
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -4,19

LINE OF INTERSECTION FOR PLANE L2
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    4,19

LINE OF INTERSECTION FOR PLANE L3
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -3,94

LINE OF INTERSECTION FOR PLANE L4
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    3,94

LINE OF INTERSECTION FOR PLANE L5
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -2,00

LINE OF INTERSECTION FOR PLANE L6
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    2,00

LINE OF INTERSECTION FOR PLANE L7
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -2,44

LINE OF INTERSECTION FOR PLANE L8
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -1,56

LINE OF INTERSECTION FOR PLANE L9
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    1,56

LINE OF INTERSECTION FOR PLANE L10
```

IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 2,44

LINE OF INTERSECTION FOR PLANE L11
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 2.19

LINE OF INTERSECTION FOR PLANE L12
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2.19

LINE OF INTERSECTION FOR PLANE L13
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -1.94

LINE OF INTERSECTION FOR PLANE L14
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 1.94

LINE OF INTERSECTION FOR PLANE L11A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 2,22

LINE OF INTERSECTION FOR PLANE L12A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2,22

LINE OF INTERSECTION FOR PLANE L1A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -4,44

LINE OF INTERSECTION FOR PLANE L2A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 4,22

LINE OF INTERSECTION FOR PLANE L16
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 4,31

LINE OF INTERSECTION FOR PLANE L17
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.19

LINE OF INTERSECTION FOR PLANE L18
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.19

LINE OF INTERSECTION FOR PLANE L19
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.50

LINE OF INTERSECTION FOR PLANE L20
IS PARALLEL TO THE X PRIME AXIS AND

UNCLASSIFIED

0

0,00

-2,00

2,00

1,56

2,44

-2,44

-1,56

-4,96

-3,94

-4,99

4,34

LINE, FROM(-4,46, -0,00) TO (-4,46, -5,00)

EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER. THUS, NO INTERSECTION

EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER. THUS, NO INTERSECTION

UNCLASSIFIED

INTERSECTS THE Y PRIME AXIS AT 0.

LINE OF INTERSECTION FOR PLANE PL1B
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE PL5A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE PL6A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L9A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L10A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L7A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L8A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L15
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE LC
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L15A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L16A
IS VERTICAL,I,E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

C9 , INTERSECTION IS: SINGLE

C10 , INTERSECTION IS: PARALL

C11 , INTERSECTION IS: PARALL

UNCLASSIFIED

. TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER. THUS, NO INTERSECTION

. TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER. THUS, NO INTERSECTION

ENTER AT 3,50, -1.50, RADIUS = 0,69

ENTER AT 3,50, 1.50, RADIUS = 0,69

ENTER AT -3,50, 1.50, RADIUS = 0,69

ENTER AT -3,50, -1.50, RADIUS = 0,69

ENTER AT -3,81, -1.81, RADIUS = 0,13

ENTER AT 3,81, -1.81, RADIUS = 0,13

ENTER AT 3,81, 1.81, RADIUS = 0,13

ENTER AT -3,81, 1.81, RADIUS = 0,13

```
C14     , INTERSECTION IS:   PARALLE

C15     , INTERSECTION IS:   PARALLE

C1      CIRCULAR INTERSECTION WITH, C

C2      CIRCULAR INTERSECTION WITH, C

C3      CIRCULAR INTERSECTION WITH, C

C4      CIRCULAR INTERSECTION WITH, C

C5      CIRCULAR INTERSECTION WITH, C

C6      CIRCULAR INTERSECTION WITH, C

C7      CIRCULAR INTERSECTION WITH, C

C8      CIRCULAR INTERSECTION WITH, C
```

```
HOPE2
A,B,C,D 0,0000000 1.0000000 0,0000000 0,0000000
```

ANGLES BETWEEN HOPEZAND THE DEFINED PLANES

PL0	PERPENDICULAR TO REFERENCE PLANE
PL1	PERPENDICULAR TO REFERENCE PLANE
PL2	PERPENDICULAR TO REFERENCE PLANE
PL3	PERPENDICULAR TO REFERENCE PLANE
PL4	PERPENDICULAR TO REFERENCE PLANE
PL5	PERPENDICULAR TO REFERENCE PLANE
PL7A	PERPENDICULAR TO REFERENCE PLANE
PL6	PERPENDICULAR TO REFERENCE PLANE
LX	PARALLEL TO REFERENCE PLANE
LY	PERPENDICULAR TO REFERENCE PLANE
L1	PERPENDICULAR TO REFERENCE PLANE
L2	PERPENDICULAR TO REFERENCE PLANE
L3	PERPENDICULAR TO REFERENCE PLANE
L4	PERPENDICULAR TO REFERENCE PLANE

UNCLASSIFIED

L5	PERPENDICULAR TO REFERENCE PLANE
L6	PERPENDICULAR TO REFERENCE PLANE
L7	PERPENDICULAR TO REFERENCE PLANE
L8	PERPENDICULAR TO REFERENCE PLANE
L9	PERPENDICULAR TO REFERENCE PLANE
L10	PERPENDICULAR TO REFERENCE PLANE
L11	PARALLEL TO REFERENCE PLANE
L13	PARALLEL TO REFERENCE PLANE
L14	PARALLEL TO REFERENCE PLANE
L11A	PARALLEL TO REFERENCE PLANE
L1A	PERPENDICULAR TO REFERENCE PLANE
L2A	PERPENDICULAR TO REFERENCE PLANE
L16	PERPENDICULAR TO REFERENCE PLANE
L17	PARALLEL TO REFERENCE PLANE
L19	PARALLEL TO REFERENCE PLANE
PL1A	AT 90.0 DEGREES TO REFERENCE PLANE
PL1B	PERPENDICULAR TO REFERENCE PLANE

PL2A	AT 90.0 DEGREES TO REFERENCE PLANE
PL3A	AT 90.0 DEGREES TO REFERENCE PLANE
PL5A	PERPENDICULAR TO REFERENCE PLANE
PL6A	PERPENDICULAR TO REFERENCE PLANE
L9A	PERPENDICULAR TO REFERENCE PLANE
L10A	PERPENDICULAR TO REFERENCE PLANE
L7A	PERPENDICULAR TO REFERENCE PLANE
L8A	PERPENDICULAR TO REFERENCE PLANE
L21A	AT 90.0 DEGREES TO REFERENCE PLANE
L22A	AT 90.0 DEGREES TO REFERENCE PLANE
L15	PERPENDICULAR TO REFERENCE PLANE
LB	AT 90.0 DEGREES TO REFERENCE PLANE
LC	PERPENDICULAR TO REFERENCE PLANE
L21	AT 90.0 DEGREES TO REFERENCE PLANE
L15A	PERPENDICULAR TO REFERENCE PLANE
L16A	PERPENDICULAR TO REFERENCE PLANE

HOPE2   INTERSECTIONS WITH ALLOWED DEFINED SURFACES

LINE OF INTERSECTION FOR PLANE PL0
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.10

LINE OF INTERSECTION FOR PLANE PL1
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE PL2
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.69

LINE OF INTERSECTION FOR PLANE PL3
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2.13

LINE OF INTERSECTION FOR PLANE PL4
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2.36

LINE OF INTERSECTION FOR PLANE PL5
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -1.50

LINE OF INTERSECTION FOR PLANE PL7A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.05

LINE OF INTERSECTION FOR PLANE PL6
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.50

LINE OF INTERSECTION FOR PLANE LY
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE L1
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -4.19

LINE OF INTERSECTION FOR PLANE L2
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 4.19

LINE OF INTERSECTION FOR PLANE L3

```
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -3,94


LINE OF INTERSECTION FOR PLANE L4
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   3,94


LINE OF INTERSECTION FOR PLANE L5
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -2,00


LINE OF INTERSECTION FOR PLANE L6
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   2,00


LINE OF INTERSECTION FOR PLANE L7
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -2,44


LINE OF INTERSECTION FOR PLANE L8
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -1,56


LINE OF INTERSECTION FOR PLANE L9
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   1,56


LINE OF INTERSECTION FOR PLANE L10
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   2,44


LINE OF INTERSECTION FOR PLANE L1A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -4,44


LINE OF INTERSECTION FOR PLANE L2A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   4,22


LINE OF INTERSECTION FOR PLANE L16
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   4,31


LINE OF INTERSECTION FOR PLANE PL1A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   0.00


LINE OF INTERSECTION FOR PLANE PL1B
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   0,00


LINE OF INTERSECTION FOR PLANE PL2A
IS PARALLEL TO THE X PRIME AXIS AND
```

INTERSECTS THE Y PRIME AXIS AT -0.69

LINE OF INTERSECTION FOR PLANE PL3A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2.13

LINE OF INTERSECTION FOR PLANE PL5A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -2.00

LINE OF INTERSECTION FOR PLANE PL6A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 2.00

LINE OF INTERSECTION FOR PLANE L9A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 1.56

LINE OF INTERSECTION FOR PLANE L10A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 2.44

LINE OF INTERSECTION FOR PLANE L7A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -2.44

LINE OF INTERSECTION FOR PLANE L8A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -1.56

LINE OF INTERSECTION FOR PLANE L21A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.91

LINE OF INTERSECTION FOR PLANE L22A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -1.66

LINE OF INTERSECTION FOR PLANE L15
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -4.96

LINE OF INTERSECTION FOR PLANE LB
HAS A SLOPE = -45.0 DEGREES AND A Y INTERCEPT = -5.67

LINE OF INTERSECTION FOR PLANE LC
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -3.94

UNCLASSIFIED

Y INTERCEPT = -5.67

4.99

4.34

```
ENTER AT     -4.46,    -0.50, RADIUS =     0.50
ENTER AT     -2.38,    -0.75, RADIUS =     0.06
ENTER AT     -1.62,    -0.75, RADIUS =     0.06
ENTER AT      1.62,    -0.75, RADIUS =     0.06
ENTER AT      2.38,    -0.75, RADIUS =     0.06
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
```

LINE OF INTERSECTION FOR PLANE L21
HAS A SLOPE = -45.0 DEGREES AND

LINE OF INTERSECTION FOR PLANE L15A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L16A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT

C9 CIRCULAR INTERSECTION WITH,

C10 CIRCULAR INTERSECTION WITH,

C11 CIRCULAR INTERSECTION WITH,

C14 CIRCULAR INTERSECTION WITH,

C15 CIRCULAR INTERSECTION WITH,

C1 , INTERSECTION IS PARALL

C2 , INTERSECTION IS PARALL

C3 , INTERSECTION IS PARALL

C4 , INTERSECTION IS PARALL

C5 , INTERSECTION IS PARALL

C6 , INTERSECTION IS PARALL

C7 , INTERSECTION IS PARALL

C8 , INTERSECTION IS PARALL

```
HOPE3
A,B,C,D 1,0000000 0.0000000 0.0000000 0,0000000
```

ANGLES BETWEEN HOPE3 AND THE DEFINED PLANES

PL0	PERPENDICULAR TO REFERENCE PLANE
PL1	PERPENDICULAR TO REFERENCE PLANE
PL2	PERPENDICULAR TO REFERENCE PLANE
PL3	PERPENDICULAR TO REFERENCE PLANE
PL4	PERPENDICULAR TO REFERENCE PLANE
PL5	PERPENDICULAR TO REFERENCE PLANE
PL7A	PERPENDICULAR TO REFERENCE PLANE
PL6	PERPENDICULAR TO REFERENCE PLANE
LX	PERPENDICULAR TO REFERENCE PLANE
LY	PARALLEL TO REFERENCE PLANE
L2	PARALLEL TO REFERENCE PLANE
L3	PARALLEL TO REFERENCE PLANE
L4	PARALLEL TO REFERENCE PLANE
L5	PARALLEL TO REFERENCE PLANE

L6	PARALLEL TO REFERENCE PLANE
L8	PARALLEL TO REFERENCE PLANE
L10	PARALLEL TO REFERENCE PLANE
L11	PERPENDICULAR TO REFERENCE PLANE
L12	PERPENDICULAR TO REFERENCE PLANE
L13	PERPENDICULAR TO REFERENCE PLANE
L14	PERPENDICULAR TO REFERENCE PLANE
L11A	PERPENDICULAR TO REFERENCE PLANE
L12A	PERPENDICULAR TO REFERENCE PLANE
L2A	PARALLEL TO REFERENCE PLANE
L16	PARALLEL TO REFERENCE PLANE
L17	PERPENDICULAR TO REFERENCE PLANE
L18	PERPENDICULAR TO REFERENCE PLANE
L19	PERPENDICULAR TO REFERENCE PLANE
L20	PERPENDICULAR TO REFERENCE PLANE
PL1A	PERPENDICULAR TO REFERENCE PLANE
PL1B	PARALLEL TO REFERENCE PLANE

PL2A	PERPENDICULAR TO REFERENCE PLANE	
PL3A	PERPENDICULAR TO REFERENCE PLANE	
PL6A	PARALLEL TO REFERENCE PLANE	
L10A	PARALLEL TO REFERENCE PLANE	
L8A	PARALLEL TO REFERENCE PLANE	
L21A	PERPENDICULAR TO REFERENCE PLANE	
L22A	PERPENDICULAR TO REFERENCE PLANE	
LB	AT	45.0 DEGREES TO REFERENCE PLANE
L21	AT	45.0 DEGREES TO REFERENCE PLANE
L16A	PARALLEL TO REFERENCE PLANE	

HOPE3 INTERSECTIONS WITH ALLOWED DEFINED SURFACES

LINE OF INTERSECTION FOR PLANE PL0
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.10

LINE OF INTERSECTION FOR PLANE PL1
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE PL2
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.69

LINE OF INTERSECTION FOR PLANE PL3
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2.13

LINE OF INTERSECTION FOR PLANE PL4
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -2.36

LINE OF INTERSECTION FOR PLANE PL5
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -1.50

LINE OF INTERSECTION FOR PLANE PL7A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.05

LINE OF INTERSECTION FOR PLANE PL6
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.50

LINE OF INTERSECTION FOR PLANE LX
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE L11
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 2.19

LINE OF INTERSECTION FOR PLANE L12
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -2.19

LINE OF INTERSECTION FOR PLANE L13

UNCLASSIFIED

S
T =1,94

S
T 1,94

1A
S
T 2,22

2A
S
T -2,22

7
S
T 0,19

8
S
T -0,19

9
S
T 0,50

0
S
T -0,50

1A
ND
0.00

2A
ND
0.69

3A
ND
2.13

1A
ND
0.91

2A
ND
1.66

ALLEL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER. THUS, NO INTERSECTION

```
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE L
IS VERTICAL,I.E,SLOPE = 90 DEGREE
AND INTERSECTS THE X PRIME AXIS

LINE OF INTERSECTION FOR PLANE P
IS PARALLEL TO THE X PRIME AXIS
INTERSECTS THE Y PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE P
IS PARALLEL TO THE X PRIME AXIS
INTERSECTS THE Y PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE P
IS PARALLEL TO THE X PRIME AXIS
INTERSECTS THE Y PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L
IS PARALLEL TO THE X PRIME AXIS
INTERSECTS THE Y PRIME AXIS AT

LINE OF INTERSECTION FOR PLANE L
IS PARALLEL TO THE X PRIME AXIS
INTERSECTS THE Y PRIME AXIS AT

C9      , INTERSECTION IST    PA
```

```
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
EL TO PRESENT HOPE PLANE AT DISTANCE GT RADIUS OF CYLINDER.  THUS, NO INTERSECTION
```

C10 , INTERSECTION IST PARALL

C11 , INTERSECTION IST PARALL

C14 , INTERSECTION IST PARALL

C15 , INTERSECTION IST PARALL

C1 , INTERSECTION IST PARALL

C2 , INTERSECTION IST PARALL

C3 , INTERSECTION IST PARALL

C4 , INTERSECTION IST PARALL

C5 , INTERSECTION IST PARALL

C6 , INTERSECTION IST PARALL

C7 , INTERSECTION IST PARALL

C8 , INTERSECTION IST PARALL

```
HØPE
A,B,C,D 0,0000000 0.7071068 0,7071068 0,0000000
```

ANGLES BETWEEN HOPE AND THE DEFINED PLANES

```
PL0      AT     45.0 DEGREES TO REFERENCE PLANE

PL1      AT     45.0 DEGREES TO REFERENCE PLANE

PL2      AT     45.0 DEGREES TO REFERENCE PLANE

PL3      AT     45.0 DEGREES TO REFERENCE PLANE

PL4      AT     45.0 DEGREES TO REFERENCE PLANE

PL5      AT    135.1 DEGREES TO REFERENCE PLANE

PL7A     AT     45.0 DEGREES TO REFERENCE PLANE

PL6      AT     45.0 DEGREES TO REFERENCE PLANE

LX       AT     45.0 DEGREES TO REFERENCE PLANE

LY       PERPENDICULAR TO REFERENCE PLANE

L1       PERPENDICULAR TO REFERENCE PLANE

L2       PERPENDICULAR TO REFERENCE PLANE

L3       PERPENDICULAR TO REFERENCE PLANE

L4       PERPENDICULAR TO REFERENCE PLANE
```

L5	PERPENDICULAR TO REFERENCE PLANE	
L6	PERPENDICULAR TO REFERENCE PLANE	
L7	PERPENDICULAR TO REFERENCE PLANE	
L8	PERPENDICULAR TO REFERENCE PLANE	
L9	PERPENDICULAR TO REFERENCE PLANE	
L10	PERPENDICULAR TO REFERENCE PLANE	
L11	AT	45.0 DEGREES TO REFERENCE PLANE
L12	AT	135.1 DEGREES TO REFERENCE PLANE
L13	AT	45.0 DEGREES TO REFERENCE PLANE
L14	AT	45.0 DEGREES TO REFERENCE PLANE
L11A	AT	45.0 DEGREES TO REFERENCE PLANE
L12A	AT	135.1 DEGREES TO REFERENCE PLANE
L1A	PERPENDICULAR TO REFERENCE PLANE	
L2A	PERPENDICULAR TO REFERENCE PLANE	
L16	PERPENDICULAR TO REFERENCE PLANE	
L17	AT	45.0 DEGREES TO REFERENCE PLANE
L18	AT	135.1 DEGREES TO REFERENCE PLANE

L19	AT	45.0 DEGREES TO REFERENCE PLANE
L20	AT	135.1 DEGREES TO REFERENCE PLANE
PL1A	AT	45.0 DEGREES TO REFERENCE PLANE
PL1B	PERPENDICULAR TO REFERENCE PLANE	
PL2A	AT	135.1 DEGREES TO REFERENCE PLANE
PL3A	AT	135.1 DEGREES TO REFERENCE PLANE
PL5A	PERPENDICULAR TO REFERENCE PLANE	
PL6A	PERPENDICULAR TO REFERENCE PLANE	
L9A	PERPENDICULAR TO REFERENCE PLANE	
L10A	PERPENDICULAR TO REFERENCE PLANE	
L7A	PERPENDICULAR TO REFERENCE PLANE	
L8A	PERPENDICULAR TO REFERENCE PLANE	
L21A	AT	135.1 DEGREES TO REFERENCE PLANE
L22A	AT	135.1 DEGREES TO REFERENCE PLANE
L15	PERPENDICULAR TO REFERENCE PLANE	
LB	AT	60.0 DEGREES TO REFERENCE PLANE

LC	PERPENDICULAR TO REFERENCE PLANE
L21	AT 60.0 DEGREES TO REFERENCE PLANE
L15A	PERPENDICULAR TO REFERENCE PLANE
L16A	PERPENDICULAR TO REFERENCE PLANE

HOPE INTERSECTIONS WITH ALLOWED DEFINED SURFACES

LINE OF INTERSECTION FOR PLANE PL0
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.14

LINE OF INTERSECTION FOR PLANE PL1
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE PL2
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.97

LINE OF INTERSECTION FOR PLANE PL3
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 3.01

LINE OF INTERSECTION FOR PLANE PL4
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 3.34

LINE OF INTERSECTION FOR PLANE PL5
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 2.12

LINE OF INTERSECTION FOR PLANE PL7A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT -0.07

LINE OF INTERSECTION FOR PLANE PL6
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.71

LINE OF INTERSECTION FOR PLANE LX
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE LY
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT 0.00

LINE OF INTERSECTION FOR PLANE L1
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT -4.19

LINE OF INTERSECTION FOR PLANE L2

```
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   4,19


LINE OF INTERSECTION FOR PLANE L3
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -3,94


LINE OF INTERSECTION FOR PLANE L4
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   3,94


LINE OF INTERSECTION FOR PLANE L5
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -2,00


LINE OF INTERSECTION FOR PLANE L6
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   2,00


LINE OF INTERSECTION FOR PLANE L7
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -2,44


LINE OF INTERSECTION FOR PLANE L8
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -1,56


LINE OF INTERSECTION FOR PLANE L9
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   1,56


LINE OF INTERSECTION FOR PLANE L10
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   2,44


LINE OF INTERSECTION FOR PLANE L11
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   3.09


LINE OF INTERSECTION FOR PLANE L12
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT  -3.09


LINE OF INTERSECTION FOR PLANE L13
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT  -2.74


LINE OF INTERSECTION FOR PLANE L14
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   2.74


LINE OF INTERSECTION FOR PLANE L11A
IS PARALLEL TO THE X PRIME AXIS AND
```

```
INTERSECTS THE Y PRIME AXIS AT   3.14

LINE OF INTERSECTION FOR PLANE L12A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT  -3.14

LINE OF INTERSECTION FOR PLANE L1A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -4.44

LINE OF INTERSECTION FOR PLANE L2A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   4.22

LINE OF INTERSECTION FOR PLANE L16
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   4.31

LINE OF INTERSECTION FOR PLANE L17
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   0.26

LINE OF INTERSECTION FOR PLANE L18
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT  -0.26

LINE OF INTERSECTION FOR PLANE L19
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   0.71

LINE OF INTERSECTION FOR PLANE L20
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT  -0.71

LINE OF INTERSECTION FOR PLANE PL1A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   0.00

LINE OF INTERSECTION FOR PLANE PL1B
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   0.00

LINE OF INTERSECTION FOR PLANE PL2A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   0.97

LINE OF INTERSECTION FOR PLANE PL3A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT   3.01

LINE OF INTERSECTION FOR PLANE PL5A
IS VERTICAL,I.E,SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT  -2.00
```

```
LINE OF INTERSECTION FOR PLANE PL6A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    2,00

LINE OF INTERSECTION FOR PLANE L9A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    1,56

LINE OF INTERSECTION FOR PLANE L10A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT    2,44

LINE OF INTERSECTION FOR PLANE L7A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -2,44

LINE OF INTERSECTION FOR PLANE L8A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -1,56

LINE OF INTERSECTION FOR PLANE L21A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT    1.28

LINE OF INTERSECTION FOR PLANE L22A
IS PARALLEL TO THE X PRIME AXIS AND
INTERSECTS THE Y PRIME AXIS AT    2.34

LINE OF INTERSECTION FOR PLANE L15
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -4,96


LINE OF INTERSECTION FOR PLANE LB
HAS A SLOPE =     54.8  DEGREES  AND A Y INTERCEPT =    8.02

LINE OF INTERSECTION FOR PLANE LC
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -3,94


LINE OF INTERSECTION FOR PLANE L21
HAS A SLOPE =     54.8  DEGREES  AND A Y INTERCEPT =    8.02

LINE OF INTERSECTION FOR PLANE L15A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
AND INTERSECTS THE X PRIME AXIS AT   -4,99

LINE OF INTERSECTION FOR PLANE L16A
IS VERTICAL,I.E.SLOPE = 90 DEGREES
```

```
1, SEMI-MAJOR AXIS =    0.71, SEMI-MINOR AXIS =    0.50, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.08, SEMI-MINOR AXIS =    0.06, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.08, SEMI-MINOR AXIS =    0.06, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.08, SEMI-MINOR AXIS =    0.06, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.08, SEMI-MINOR AXIS =    0.06, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.97, SEMI-MINOR AXIS =    0.69, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.97, SEMI-MINOR AXIS =    0.69, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.97, SEMI-MINOR AXIS =    0.69, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.97, SEMI-MINOR AXIS =    0.69, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.18, SEMI-MINOR AXIS =    0.12, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.18, SEMI-MINOR AXIS =    0.12, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.18, SEMI-MINOR AXIS =    0.12, ECC = 0.70711, PHI =  -90.00
 , SEMI-MAJOR AXIS =    0.18, SEMI-MINOR AXIS =    0.12, ECC = 0.70711, PHI =  -90.00

        ELAPSED TIME =    4  MIN   58 SEC

        ELAPSED TIME =    5  MIN    3 SEC

X
```

AND INTERSECTS THE X PRIME AXIS AT 4,34

```
C9       ,ELLIPSE, CENTER AT   -4,46,    0,7
C10      ,ELLIPSE, CENTER AT   -2,38,    1.0
C11      ,ELLIPSE, CENTER AT   -1,62,    1.0
C14      ,ELLIPSE, CENTER AT    1,62,    1.0
C15      ,ELLIPSE, CENTER AT    2,38,    1.0
C1       ,ELLIPSE, CENTER AT    3,50,   -2,1
C2       ,ELLIPSE, CENTER AT    3,50,    2,1
C3       ,ELLIPSE, CENTER AT   -3,50,    2,1
C4       ,ELLIPSE, CENTER AT   -3,50,   -2,1
C5       ,ELLIPSE, CENTER AT   -3,81,   -2,5
C6       ,ELLIPSE, CENTER AT    3,81,   -2,5
C7       ,ELLIPSE, CENTER AT    3,81,    2,5
C8       ,ELLIPSE, CENTER AT   -3,81,    2,5
```

START EXECUTION OF SECTION 200

START EXECUTION OF SECTION 400

POST PROCESSOR NOT AVAILABLE ON LIBTAP BEND

PART PROGRAM EXECUTION TIME = 5 MIN 5 SEC

```
JOB MESSAGES
               JOB,403775,  600,573KPT
      JOB SEQ 03775 WILL PLOT       10 MIN,
       LENGTH   5 FT, 01 IN,
               RELEASED 49=MT04=(CROSEC CALCOM),ED=  ,RL=01,DATE=110370,RC=000
```

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)* Naval Research Laboratory Washington, D.C. 20390	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

CROSEC, A FORTRAN IV — APT PROGRAM TO GIVE ORTHOGRAPHIC, SECTION, AND DEFINABLE PERSPECTIVE VIEWS OF A PLANAR-CURVED SURFACE

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
An interim report on a continuing problem.

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

Kingsley P. Thompson

6. REPORT DATE April 28, 1971	7a. TOTAL NO. OF PAGES 240	7b. NO. OF REFS 10
8a. CONTRACT OR GRANT NO. NRL Problem Z00.01 b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) NRL Report 7228 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 Naval Research Laboratory

13. ABSTRACT

A Fortran IV program, CROSEC (MOD 2.0) has been written to operate in the environment of NRL's APT system on the CDC 3800 Computer. This program enables the part programmer for numerically controlled tooling systems to obtain plots of the intersections between a specially designated plane (called the HOPE plane) and certain defined forms in his APT part program. The forms with which intersections can be obtained are lines, planes, circles, cylinders, and spheres. When the HOPE plane corresponds to a coordinate plane (XY, XZ, and YZ), an orthographic projection pattern is obtained corresponding to engineering drawing's top, front, and end views, respectively. When the HOPE plane cuts through the volume of the part surface and the plotting limits are the part's minimum and maximum dimensions, a true view of the cross section is obtained. When the HOPE plane lies outside the volume of the part and is skewed with respect to the coordinate planes, an abstract pattern is obtained because the planes and cylinders are not bounded. Such an abstract pattern, when composed only of plane intersections, through an unprogrammed process, can be interpreted as a perspective view of the part. When the plotting limits extend beyond the dimensions of the part and the HOPE plane cuts its volume, a combination of true view and interpretable perspective view is obtained.

Preliminary considerations suggest that a hidden-line-detection addition to the program would not be difficult because the abstract pattern indicates depth in an easily identifiable manner, and that the abstract intersection pattern forms a suitable framework for describing tool motions in an interactive environment.

(Abstract continues)

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6801

Security Classification

UNCLASSIFIED

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
APT						
CROSEC						
Cross-sections						
Plane intersections						
Intersections						
Graphics						
Passive graphics						
Plotting						
Computer aided design						
Cross-sectional-plotting						

Up to ten viewing (i.e. HOPE) planes can be designated in one Part Program. The report includes programmer instructions, a complete description of CROSEC (MOD 2.0) with flowcharts and listings and, as an example, the computer output resulting from a run on a production piece. CROSEC (MOD 2.0) is the implementation of the computational forms developed in NRL Reports 7025 and 7202.