DANIEL F. MERRIAM, Editor

FORTRAN IV CDC 6400 COMPUTER PROGRAM FOR CONSTRUCTING ISOMETRIC DIAGRAMS

By

WILLIAM B. WRAY, Jr.

University of California, Berkeley

COMPUTER CONTRIBUTION 44

State Geological Survey

The University of Kansas, Lawrence

1970

in cooperation with the
American Association of Petroleum Geologists
Tulsa, Oklahoma

Editor's Remarks

The "FORTRAN IV CDC 6400 computer program for constructing isometric diagrams", by W.B. Wray, Jr., will find many uses by earth scientists. Long oriented to use of graphics such as maps, cross sections, and 3-D diagrams, geologists will welcome this contribution. Other COMPUTER CONTRIBUTIONS specifically concerned with graphic output are CC's 15, 23, 36, and 37. Most other programs also have output in graphic form.

For a limited time the Geological Survey will make available this program on magnetic tape for $20.00 (US). An extra charge of $10.00 is made if punched cards are required. When ordering a particular program please specify the language and computer facility needed because several versions of many programs are available.

A complete and up-to-date list of available COMPUTER CONTRIBUTIONS can be obtained by writing the Editor, COMPUTER CONTRIBUTIONS, Kansas Geological Survey, The University of Kansas, Lawrence, Kansas 66044.

Computer Contribution

1. Mathematical simulation of marine sedimentation with IBM 7090/7094 computers, by J.W. Harbaugh, 1966 . . . . (out of print)
2. A generalized two-dimensional regression procedure, by J.R. Dempsey, 1966 . . . . $0.50
3. FORTRAN IV and MAP program for computation and plotting of trend surfaces for degrees 1 through 6, by Mont O'Leary, R.H. Lippert, and O.T. Spitz, 1966 . . . . . $0.75
4. FORTRAN II program for multivariate discriminant analysis using an IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1966 . . . . $0.50
5. FORTRAN IV program using double Fourier series for surface fitting of irregularly spaced data, by W.R. James, 1966 . . . . $0.75
6. FORTRAN IV program for estimation of cladistic relationships using the IBM 7040, by R.L. Bartcher, 1966 . . . . $1.00
7. Computer applications in the earth sciences: Colloquium on classification procedures, edited by D.F. Merriam, 1966 . . . . $1.00
8. Prediction of the performance of a solution gas drive reservoir by Muskat's equation, by Apolonio Baca, 1967 . . . . $1.00
9. FORTRAN IV program for mathematical simulation of marine sedimentation with IBM 7040 or 7094 computers, by J.W. Harbaugh and W.J. Wahlstedt, 1967 . . . . . . $1.00
10. Three-dimensional response surface program in FORTRAN II for the IBM 1620 computer, by R.J. Sampson and J.C. Davis, 1967 . . . . $0.75
11. FORTRAN IV program for vector trend analyses of directional data, by W.T. Fox, 1967 . $1.00

(continued on inside back cover)

FORTRAN IV CDC 6400 COMPUTER PROGRAM FOR CONSTRUCTING ISOMETRIC DIAGRAMS

by

William B. Wray, Jr.

INTRODUCTION

Isometric pictorial representation has long been favored by geologists as a means of displaying three-dimensional surfaces and bodies. A principal advantage over perspective representation is that plotting and measurement can be made readily anywhere on the diagram. Lobeck (1924, p. 120-121) lists the following properties of isometric diagrams.

(1) "Measurements may be made to scale anywhere upon the upper surface of the block in the direction of either of the two coordinates, which in the object itself are at right angles to each other.
(2) Distances in other directions are not commensurate with each other unless measured along lines parallel to each other, and they are not commensurate with distances measured on the sides of the block.
(3) All the lines parallel in the object itself are parallel in the isometric drawing.
(4) All lines vertical in the object are vertical in the drawing.
(5) If the vertical scale of the drawing is the same as its horizontal scale, then the measurements made in the direction of any of the three coordinates are commensurate.
(6) All angles in an isometric drawing are distorted, and even two angles lying in the same plane cannot be compared with each other unless they lie in exactly similar positions."

The I3D subroutines included with this contribution provide an efficient and fast method of displaying a surface - a single valued function of two variables - in isometric or similar pictorial representation. Numerous options are available through the ISPECS and RSPECS parameter arrays. Among the most important options

(i) The diagrams can be seen from all theoretically possible viewpoints.
(ii) The data can be viewed in an unlimited number of rectangular subblocks, all automatically spaced to avoid point overlap (x and y direction spacings are variable). Any rectangular subarea within the data array area may be considered for plotting and any number and configuration of rectangular subblocks within this subarea may be skipped if desired.
(iii) Scale, line and point spacing, and density of lines and points to be plotted are variable.
(iv) The vertical (z) scale may be expanded or contracted relative to the horizontal (x, y) scale without restriction. Distances in x and/or y directions can be stretched or foreshortened independently, with scale remaining constant.
(v) Surfaces may be plotted with or without the traditional block diagram frame, and major and minor scale ticks are available. A standard labelling routine can be used.
(vi) Data input is flexible. Points can be arranged in north-south (y direction) or east-west (x direction) grid lines.

Possible applications of isometric pictorial representation can be considered within the framework of four general types of surfaces.

(1) North-south, east-west, elevation surfaces, such as topographic surfaces, tops and bottoms of beds, buried bedrock surfaces, etc.
(2) North-south, east-west, z surfaces, where z is any measureable quantity, such as, for example, thickness or percent of sand in a sedimentary bed, or Bouguer anomalies.
(3) X-y-z surfaces, for example equilibrium surfaces in chemical stability diagrams.
(4) X-y-f(x, y) surfaces, mathematical functions of two variables, including functions of many terms where the values at different points cannot be readily visualized. Many preceeding papers in the Computer Contributions series deal with functions of this type. Examples are the double (two-dimensional) polynomial and Fourier series currently important in trend-surface and harmonic and spectral analysis.

Because of wide interest in computer graphics, a number of algorithms have been written which have the capability of generating three-dimensional diagrams (e.g. Rohlf, 1969). Almost all of the algorithms produce perspective rather than isometric diagrams and most have more -or - less serious bugs and severe limitations in point -array size due to rapid increase of execution time with increase in number of points (A.R. Paradis, personal communication).

Algorithms to construct isometric diagrams seemingly are less common, as I do not know of any, other than I3D.

The I3D subroutines have been subjected to thorough testing, and numerous isometric drawings have been obtained, but it is entirely possible that algorithm defects are present due to the complexity of the subroutines.

Acknowledgments.- Mr. Arthur R. Paradis, in charge of computer graphics at the University of California, Berkeley, Computer Center, gave encouragement and arranged use of computer facilities during the program development. His methodology is evident in both the organization of the latter half of this paper and in the subroutine structuring. The Anaconda Company provided some development computer time and permitted use of the view of the Berkeley open-pit mine in Example 2. Prof. Charles Meyer, Prof. Garniss Curtis, and Mr. Paradis kindly reviewed the manuscript.

PROGRAM ALGORITHM

I3D1GL, the subroutine which constructs the isometric projection, processes parallel rows of points on the surface to be represented. A visibility or horizon line is constructed and is carried from the front to the rear of the surface. Each point on the surface need be tested only once to define visibility. As a consequence, calculation time increases as the first power of the number of points defining the surface, rather than the square of the number of points. This is of great practical importance in large and detailed projections (see Example 2).

An obvious next step is the development of routines to process multiple-valued functions of two variables. Both upper and lower visibility lines would be carried for each surface, so that the undersides of surfaces could be seen. In this manner, cut-away block diagrams, similar to those used to portray geology of underground mines, could be constructed.

Another useful feature to incorporate is wide-spaced surface gridding as an aid to rapid location of (x, y) positions. I have written a subroutine, I3D2GL, which permits this option, but it is not in a general form so has not been included in this paper. The subroutine takes longer to execute and requires significantly more core space, and at the present time conversion has not been justified.

I3D1GL and supporting subroutines were written to create projections at least possible cost so that, if desired, the program could be used on a routine basis. All I3D subroutines make use of a common work array BUFI and complete variable dimensioning to reduce central memory core storage requirements to a minimum. I3D1GL requires 3313_8 storage locations and the remaining subroutines total 3574_8, independent of work array requirements. To hold execution time for all subroutines to a minimum, some care was taken to make the FORTRAN coding of high efficiency. To reduce plotting cost, I3D1GL generates plotter construction codes in such a manner as to reduce as much as possible off-paper pen travel.

Execution data specifications are entered into the various subroutines from a main program via two common parameter arrays, rather than via data cards. This considerably simplifies data input, because the need for variable card formats (for the completely general situation) is eliminated. Also, particular specifications are carried automatically between subroutines unless changed.

RELATION OF VIEWPOINT TO BLOCK ORIENTATION

It is possible to characterize the orientation of a block in isometric projection by means of an azimuth, or line-of-sight, value and an apparent angle of inclination (looking down from above). The relation can be derived as follows.

Consider a rectilinear isometric block, Figures 1 and 2, with the properties given by Lobeck and with the edges trending north-south and east-west. To permit construction of the projection, it is necessary to solve for angles a_1 and a_2 in terms of ψ and γ .

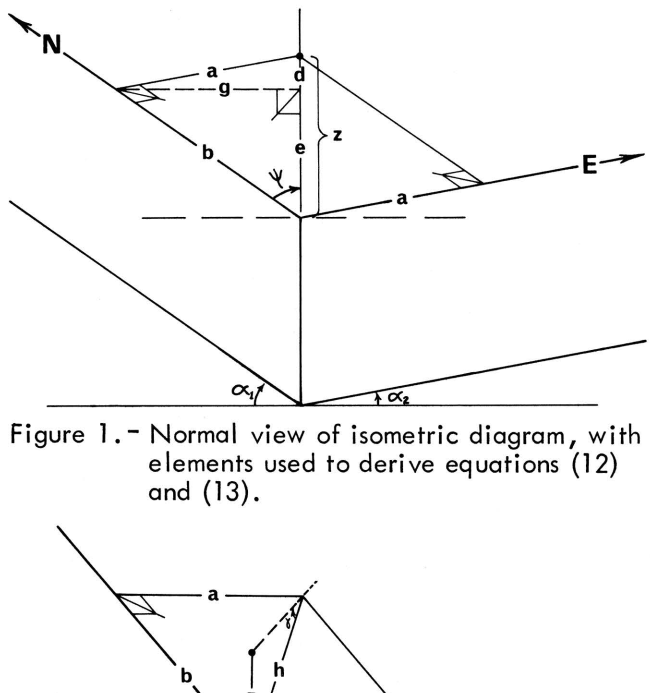

Figure 1.- Normal view of isometric diagram, with elements used to derive equations (12) and (13).

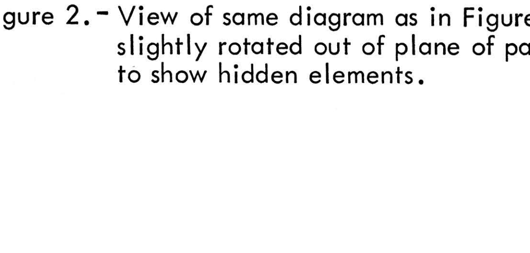

Figure 2.- View of same diagram as in Figure 1, slightly rotated out of plane of paper to show hidden elements.

$\psi = \arctan (a/b)$

$\gamma = \arctan (z/h)$

α_1 is defined as the angle to the N-S line from a horizontal line. α_1 is positive if clockwise (as shown), and negative otherwise.

α_2 is defined as the angle to the E-W line from a horizontal line. α_2 is positive if counterclockwise (as shown), and negative otherwise.

From Figure 2,

$$\tan \gamma = z/h = \frac{z \cos \psi}{b} \quad , \qquad (1)$$

and from Figure 1,

$$z = e + d,$$

so

$$z/b = e/b + d/b \quad ; \qquad (2)$$

also

$$e/b = \sin \alpha_1 , \qquad (3)$$

$$d/g = \tan \alpha_2 \text{ , and}$$

$$g/b = \cos \alpha_1 \quad .$$

So by multiplying the last two together,

$$d/b = \tan \alpha_2 \cos \alpha_1 \quad . \qquad (4)$$

Substituting (3) and (4) into (2) yields

$$z/b = \sin \alpha_1 + \tan \alpha_2 \cos \alpha_1 \quad . \qquad (5)$$

Substituting (5) into (1) results in

$$\tan \gamma / \cos \psi = \sin \alpha_1 + \tan \alpha_2 \cos \alpha_1 \quad , \qquad (6)$$

which can be rewritten as

$$\tan \gamma / \cos \psi = \sin \alpha_1 + \frac{\cos \alpha_1}{\cos \alpha_2} \sin \alpha_2 \, . \qquad * (7)$$

Now

$$\cos \alpha_1 = g/b \qquad (8)$$

and

$$\cos \alpha_2 = g/a \quad , \qquad (9)$$

thus (8) and (9) combine to yield, after cancelling g,

$$a/b = \cos \alpha_1 / \cos \alpha_2 \quad .$$

Because $\psi = \arctan (a/b)$,

$$\psi = \arctan (\cos \alpha_1 / \cos \alpha_2) \quad . \quad * (10)$$

Equations (7) and (10) now can be combined, and successively α_1 and α_2 isolated.

$$\frac{\tan \gamma}{\cos \psi} = \sin \alpha_1 + \tan \psi \sin \alpha_2 \, ,$$

where $\tan \psi$, $\left(\frac{\tan \gamma}{\cos \psi} \right)$ are numbers, call them A and B respectively. So on substitution and reordering,

$$\sin \alpha_1 = B - A \sin \alpha_2 \quad . \qquad (11)$$

From (10),

$$\cos \alpha_1 = A \cos \alpha_2 \text{ , and}$$

squaring both equations gives

$$\sin^2 \alpha_1 = B^2 - 2AB \sin \alpha_2 + A^2 \sin^2 \alpha_2$$

$$\cos^2 \alpha_1 = A^2 \cos^2 \alpha_2 \quad .$$

Adding them together gives

$$\sin^2 \alpha_1 + \cos^2 \alpha_1 = A^2 (\sin^2 \alpha_2 + \cos^2 \alpha_2) + B^2 - 2AB \sin \alpha_2 \quad .$$

Because $\sin^2 + \cos^2 = 1$,

$$1 = A^2 + B^2 - 2AB \sin \alpha_2 \text{, and}$$

solving for α_2 yields

$$\alpha_2 = \arcsin \left(\frac{A^2 + B^2 - 1}{2AB} \right) \quad .$$

Resubstituting for A and B results in

$$\alpha_2 = \arcsin \left(\frac{\tan^2 \psi + \frac{\tan^2 \gamma}{\cos^2 \psi} - 1}{\frac{2 \tan \psi \tan \gamma}{\cos \psi}} \right) .$$

After shuffling and clearing of internal fractions,

$$\alpha_2 = \arcsin \left(\frac{\sin^2 \psi + \tan^2 \gamma - \cos^2 \psi}{2 \sin \psi \tan \gamma} \right) . \qquad ** (12)$$

Angle α_1 can now be obtained by substituting (12) into (11).

$$\alpha_1 = \arcsin \left(\frac{\tan \gamma}{\cos \psi} - \tan \psi \sin \alpha_2 \right)$$

which, on rewriting, becomes

$$\alpha_1 = \arcsin \left(\frac{\tan \gamma - \sin \psi \sin \alpha_2}{\cos \psi} \right) . \qquad ** (13)$$

Because of the particular nonEuclidian properties of isometric projection, it turns out that not all combinations of ψ, γ will yield real (i.e. not imaginary) diagrams. Table 1 relates α_1 and α_2 to ψ and γ. Table 1 gives values for one quadrant in 5° intervals of ψ and γ, and values in 1° intervals are given in the Appendix. Note that a surface can be portrayed in unorthodox orientations. For instance, either α_1 or α_2 can be negative, resulting in a skewed diagram. The other three quadrants are mirror images of the first, as shown in Figure 3.

If $\psi = \frac{\pi i}{4}$, $i = 1, 3, 5, 7$, then $\alpha_1 = \alpha_2$ and equations (12) and (13) reduce to

$$\alpha = \arcsin \left(\frac{1}{\sqrt{2}} \tan \gamma \right) . \qquad (14)$$

For the standard isometric projection which has $\alpha = 30°$, $\gamma = 35.2644°$.

Table 1.– Listing of α_1 and α_2 as a function of ψ and γ, in five degree intervals for one quadrant.

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	0	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90
55																			
50				83.2	77.5	72.8	68.5	64.5	60.8	57.4	54.5	52.1	50.6	50.5	53.4	64.0			
				64.0	53.4	50.5	50.6	52.1	54.5	57.4	60.8	64.5	68.5	72.8	77.5	83.2			
45	90.0	85.0	80.0	75.0	70.0	65.0	60.0	55.0	50.0	45.0	40.0	35.0	30.0	25.0	20.0	15.0	10.0	5.0	-.1
	-.0	5.0	10.0	15.0	20.0	25.0	30.0	35.0	40.0	45.0	50.0	55.0	60.0	65.0	70.0	75.0	80.0	85.0	90.0
40			84.0	75.6	68.8	62.3	55.9	49.5	43.1	36.4	29.5	22.1	14.1	5.0	-6.2	-21.9	-53.9		
			-53.9	-21.9	-6.2	5.0	14.1	22.1	29.5	36.4	43.1	49.5	55.9	62.3	68.8	75.6	84.0		
35					72.7	63.2	54.7	46.5	38.2	29.7	20.6	10.6	-.8	-14.9	-35.2				
					-35.2	-14.9	-.8	10.6	20.6	29.7	38.2	46.5	54.7	63.2	72.7				
30						68.9	56.4	45.6	35.0	24.1	12.4	-.8	-16.8	-39.4					
						-39.4	-16.8	-.8	12.4	24.1	35.0	45.6	56.4	68.9					
25							62.7	47.1	33.2	19.3	4.2	-13.5	-37.3						
							-37.3	-13.5	4.2	19.3	33.2	47.1	62.7						
20								52.7	33.3	14.9	-5.0	-30.1							
								-30.1	-5.0	14.9	33.3	52.7							
15								70.5	36.7	10.9	-17.2	-61.5							
								-61.5	-17.2	10.9	36.7	70.5							
10									49.3	7.2	-39.0								
									-39.0	7.2	49.3								
5										3.5									
										3.5									
0										.0									
										-.0									

I3D SUBROUTINES – INTRODUCTION

General Attributes

The remainder of this paper can be considered a manual on how to use the I3D subroutines, with examples. Each subroutine is treated in detail so as to make all execution alternatives as clear as possible, and each section can be read independently of the others.

I3D subroutines deal with points arranged in parallel equally spaced rows called grid lines. For input into I3D1GL, the points must be evenly spaced along each line. I3DPTS may be used to calculate additional lines. I3DPTA can calculate evenly spaced points from grid lines of irregularly spaced points. I3DCVT converts Hollerith card data to binary data suitable for input into I3DPTS, I3DPTA, or I3D1GL. Because Hollerith data must be arranged in a specific manner (see the section on Point Coordinate Data Structure), I3DCHK is available to catch gross keypunching errors and misplaced data cards.

Execution times (CP) are printed for each subroutine called. Some typical timings are

I3DCHK:	800–1400 points per second.
I3DCVT:	1000–1600 points per second.
I3DPTS:	800–1400 input points per second, less if many intermediate points are calculated.
I3DPTA:	9000–10000 final points per second.
I3D1GL:	1500–2600 points per second.

For I3D1GL, a subequal amount of plotter post–

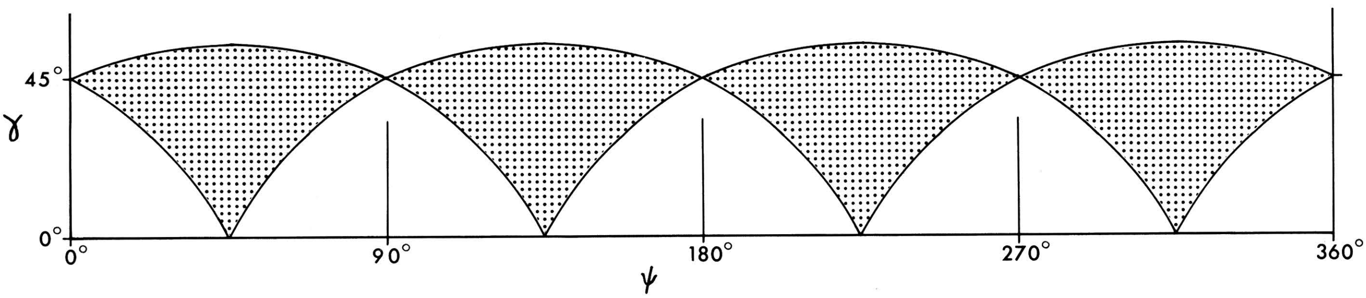

Figure 3.–Plot of γ versus ψ showing area within which isometric diagrams can be drawn.

processor time is needed in addition.

I3D1GL and the other subroutines can, in principle, process an unlimited number of (x, y, z) points, although the practical limit at Berkeley (due to limitations of one-job disk storage capacity) is about 1,800,000 points.

Selection of the value for final point spacing (FPS) along the grid lines is of importance because the positioning of the end point of a visible line as it passes into a hidden line can be in error by an average of one-half the FPS. Spacing depends upon values of RSPECS(7), RSPECS(10), ISPECS(10). Sixty to 80 points per inch give pleasing results; a general purpose range might be 10 to 100 points per inch. Also, for best results the FPS should be less than the final row spacing (FRS), which depends on RSPECS(1), RSPECS(10), ISPECS(9). Common usage has FRS = (3 to 4) * FPS, or approximately 20 lines per inch.

Terminology

Some special terminology as used in this paper:
(A) grid lines and points (Fig. 4 illustrates terminology);
(B) viewing the surface.

The attitude of the plotted surface is determined by the direction of sight (RSPECS(14), ψ) and the apparent angle of inclination (RSPECS(13), γ). Psi can range from 0° (looking due north, or in the +y direction) through 90° (looking due east, or in the +x direction) around to 359.9+°. Note particularly the requirement that the positive y direction be due north. Gamma can range from 0° (horizontal) to approximately 54.7°. See Table 1 and Figure 3 for relation of α_1 and α_2 to ψ and γ.

Program Specifications

I3D subroutines require a work array BUFI, dimensioned in the calling program, and two parameter arrays placed in labelled common as follows:

COMMON /WSPECS/ RSPECS(30), ISPECS(10)

RSPECS and ISPECS are described below. Distances given in inches refer to the actual distances as they appear when drawn on paper. Distances not so specified are data distances.

List and Description of the RSPECS Array (real (FP) numbers)

Parameter	Suggested Mnemonic	Subroutines	Description
RSPECS(1)	SPACEL	PTS 1GL	Space between adjacent grid lines (final, after calculation of intermediate lines, if any). Value remains the same regardless of value of ISPECS(9).
RSPECS(2)	TSTART	1GL	Transverse axis coordinate of first (westernmost if N-S, or northernmost if E-W) final grid line input.
RSPECS(3)	PSTART	CHK 1GL	Parallel axis coordinate of first point (southernmost if N-S grid line or westernmost if E-W grid line) of any grid line.
RSPECS(4)	ZBOTM	1GL	Reference elevation or z-coordinate, generally $\leq$ smallest z_{ij}. ZBOTM is used to position blocks on paper, or if frames are to be drawn, ZBOTM will be the z-coordinate of the bottom of the frames. ZBOTM ties data z-coordinates to RSPECS(12), the paper position of the near corner of block 1.
RSPECS(5)	ZTOP	1GL	Reference elevation or z-coordinate, generally $\geq$ largest z_{ij}. ZTOP helps control block spacing: if ZTOP = largest z_{ij}, then spacing is entirely a function of RSPECS(28) and RSPECS(29). If ZTOP > largest z_{ij}, then spacing increases by that difference.
RSPECS(6)	DELFAC	PTS	DELFAC is used in the subroutine equation DELTAD = DELFAC*width between basic grid lines, to define the point influence distance used to calculate intermediate points. DELFAC = 1.732 (or $\sqrt{3}$) is my standard value. See also the angularity algorithm under section I3DPTS.
RSPECS(7)	PS	PTA 1GL	Distance interval between adjacent final grid-line points.
RSPECS(8)	PFNISH	CHK	Parallel axis coordinate of last point (northernmost or easternmost) of any basic grid line.
RSPECS(9)	ZDIFF	CHK	Maximum difference in adjacent z-coordinate points along basic grid lines. Used to catch gross errors in data preparation and coding.
RSPECS(10)	SCALE	1GL	Scale of the diagram, in real (user) units per inch on the paper.
RSPECS(11)	APX	1GL	Paper x-coordinate, given in inches, of the near corner of the first block position of the rectangular subarea considered (see Figure 7), whether or not the block is actually plotted.
RSPECS(12)	APY	1GL	Paper y-coordinate, as above.
RSPECS(13)	GAMMA	1GL	Apparent angle of view, in degrees, measured from the horizontal downward. The possible range of values is 0. to approximately 54.7.
RSPECS(14)	PSI	1GL	Direction of line-of-sight, in degrees, measured clockwise from due north, 0. to 359.9+. Refer to Table 1 for tables of conversion into α_1 and α_2, or to equations (12) and (13). No internal check is made on compatibility of ψ and γ, so select values with care.
RSPECS(15)	FRAME	1GL	Set = 1. if frames are wanted, or = 0. if frames are not wanted.
	If no frames are wanted, no need to specify RSPECS(16) through (20) inclusive.		
RSPECS(16)	TICKS	1GL	Set = 1. if tick marks are wanted, or = 0. if not.
	If no tick marks are wanted, no need to specify RSPECS(17) through (20) inclusive.		
RSPECS(17)	MAJINT	1GL	Major tick interval, i.e. the distance (in real user units) between adjacent major tick marks.
RSPECS(18)	MININT	1GL	Minor tick interval, i.e. the distance between adjacent minor tick marks. Caution: MININT must divide evenly into MAJINT. If only one size tick

			mark is desired, either set RSPECS(20) = RSPECS(19) or set RSPECS(18) = RSPECS(17).
RSPECS(19)	MAJLNT	1GL	Major tick mark length, given in inches.
RSPECS(20)	MINLNT	1GL	Minor tick mark length, given in inches.
RSPECS(21)	LABEL	1GL	Set = 1. if the standard explanation is wanted, or = 0. if it is not (i.e. if you want to make your own explanation or if you do not want an explanation). See Example 2 for the format of the standard explanation.

If standard explanation is not wanted, no need to specify RSPECS(22) and (23). RSPECS(24) through (27) inclusive are returned from I3D1GL and can be used for new or additional labelling.

RSPECS(22)	YPOSL	1GL	Paper y-coordinates, given in inches, of base of top line of standard explanation, if standard explanation is used. YPOSL must be $\geqslant$ 3.8 inches above the lower edge of the paper to allow room for the full explanation.
RSPECS(23)	XPOSL	1GL	Paper x-coordinate, given in inches, of left edge of any line of the standard explanation. If XPOSL set = -1., then explanation will automatically be placed starting 9 inches to the left of the left-most data point position, or if that is not possible, at 1 inch to the right of the defined plotting area.
RSPECS(24)	VDATAP	1GL	(Returned from I3D1GL). Elevation (z-coordinate) of the near corner data point of the first drawn block. For elevation reference in labelling.
RSPECS(25)	XFRSVE	1GL	(Returned from I3D1GL). Paper x-coordinate, given in inches, of the left-most data point position of the plot. For use in placing explanation.
RSPECS(26)	CPC1	1GL	(Returned from I3D1GL). Parallel-axis coordinate of the near corner of the first block position (not necessarily the first block drawn).
RSPECS(27)	CTC1	1GL	(Returned from I3D1GL). Transverse-axis coordinate of the near corner of the first block position.
RSPECS(28)	PLSEP	1GL	Separation of blocks along parallel-axis direction, given in inches. One-half inch (.5) is commonly used. If only one block is to be drawn, set = 0.
RSPECS(29)	TLSEP	1GL	Separation of blocks along transverse-axis direction, as above.
RSPECS(30)	ZFACTR	1GL	Factor of vertical exaggeration or contraction. If no exaggeration wanted (i.e. true scale), set = 1. If 10x vertical exaggeration wanted, set = 10. If one-fourth vertical scale wanted, set = .25, etc. Tick marks are scaled accordingly.

List and Description of the ISPECS Array (Integer Numbers)

Parameter	Suggested Mnemonic	Subroutines	Description
ISPECS(1)	MAXBGP	CHK CVT PTS	Maximum number of basic grid-line data points per grid line.
ISPECS(2)	NUMGL	CVT PTS PTA 1GL	Number of grid lines total, including intermediate lines if any. The total number of grid lines (hence logical records) input into TAPE83 for use by I3D1GL.
ISPECS(3)	INTLIN	CVT PTS	Number of intermediate lines between two adjacent basic grid lines.
ISPECS(4)	MAXIGP	PTS PTA	Maximum number of calculated skeleton points per intermediate grid line. If the points are calculated using I3DPTS, then MAXIGP = 2 * ISPECS(1).
ISPECS(5)	NUMFPT	PTA 1GL	Number of final grid-line points in any final grid line.
ISPECS(6)	LNEDIR	CHK 1GL	Direction of data grid lines. Set = 1 if N-S lines, set = 2 if E-W lines.
ISPECS(7)	BLOCKP	1GL	Number of rows of blocks that are running parallel to data grid lines. BLOCKP + 1 is the size of CTC array (part of BUFI array). See Figure 7.
ISPECS(8)	BLOCKT	1GL	Number of rows of blocks running transverse to data grid lines. BLOCKT + 1 is the size of CPC array (part of BUFI array). See Figure 7.
ISPECS(9)	LSKIP	1GL	The number plus one of grid lines in file 83 to skip at execution of I3D1GL, e.g. if you want to draw only every third grid line (i.e. skip 2), set LSKIP = 3. If none to be skipped, set LSKIP = 1. Note that RSPECS(1) and ISPECS(2) remain the same regardless of value of LSKIP. See section D of I3D1GL.
ISPECS(10)	PTSKIP	1GL	The number plus one of data (z-coordinate) points in file 83 to skip at execution time, examples as above. Note that RSPECS(7) and ISPECS(5) remain the same regardless of value of PTSKIP. See section D of I3D1GL.

In addition, for I3D1GL, provision must be made for general plotting system parameters. At Berkeley, parameters are held in a special array, SPECS(30). SPECS(1-8), (11-12) are declared in the MAIN program and define general plotting bounds, data scaling and system constants. I3D1GL calculates projected data-point coordinates in graph-paper inches, so in your plotting system no data scaling should be used.

SPECS(13-15), (17-23), (28) are calculated within I3D1GL and its subroutines, so you should not declare them in the MAIN program. The following definitions, abstracted from the Berkeley Graphical Display System (GDS) manual, are given in order that appropriate parameters may be substituted when I3D1GL is adapted to other computer systems.

SPECS(13)	specifies the total number of data points to be plotted.
SPECS(14)	specifies the number plus one of X-array points to be skipped between adjacent plotted points.
SPECS(15)	specifies the number plus one of Y-array points to be skipped.
SPECS(17)	specifies the base dimension of the rectangle in which a plot symbol or character is defined.
SPECS(18)	specifies the height dimension of the rectangle.

SPECS(19) specifies the additional separation distance between adjacent text characters.

SPECS(20) specifies the rotation in degrees of a line of text.

SPECS(21) specifies the character set to be used for text.

SPECS(22) specifies the X distance to the lower left corner of the first character of text.

SPECS(23) specifies the Y distance to the lower left corner of the first character of text.

SPECS(28) specifies the number of decimal places to be used.

Point Coordinate Data Structure

Some details are given under subroutine headings, but an overview here is necessary. Points defining the surface in space are described by (x, y, z) coordinates. For purposes of the I3D subroutines, the points must be arranged in parallel rows called grid lines (see Fig. 4 and 5). Grid lines may be north-south trending (parallel to the y-axis) or east-west trending (parallel to the x-axis), as defined by ISPECS(6). Each grid line of points constitutes a logical record.

Because of the regular (gridded) arrangement of data points, the I3D subroutines require only partial coordinate information, calculating the rest internally. I3DCHK, I3DCVT, I3DPTS, and I3DPTA require for each grid line a sequence of point coordinate pairs (x, z) or (y,z) depending on line direction. The points must be arranged (or calculated) in sequential order so that $(x \text{ or } y)_{i+1} > (x \text{ or } y)_i$. Grid lines must be considered sequentially as well. Figure 5 illustrates point and line relationships. I3D1GL requires even less point coordinate information, because all points on grid lines are spaced equally. A logical record (one grid line) for use of I3D1GL consists of sequential z-coordinates only, each z-coordinate representing one point.

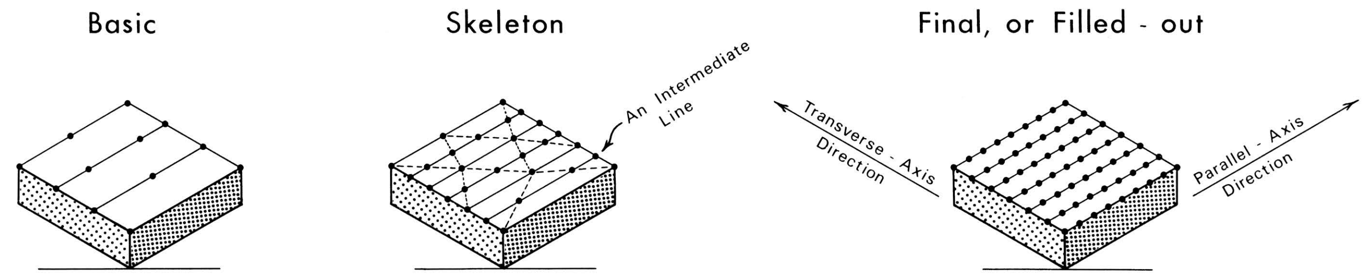

Figure 4. - Types of grid lines and points.

Three GDS subroutines are called from I3D1GL and its subroutines, and these, similar to the SPECS array, must be replaced by their counterparts if I3D1GL is used elsewhere than Berkeley. SLLILI serially connects a set of points with straight lines. TITLEG prints a Hollerith explanation. DECVAL prints a floating-point or integer number. GDSEND, used in MAIN programs, terminates plotting records.

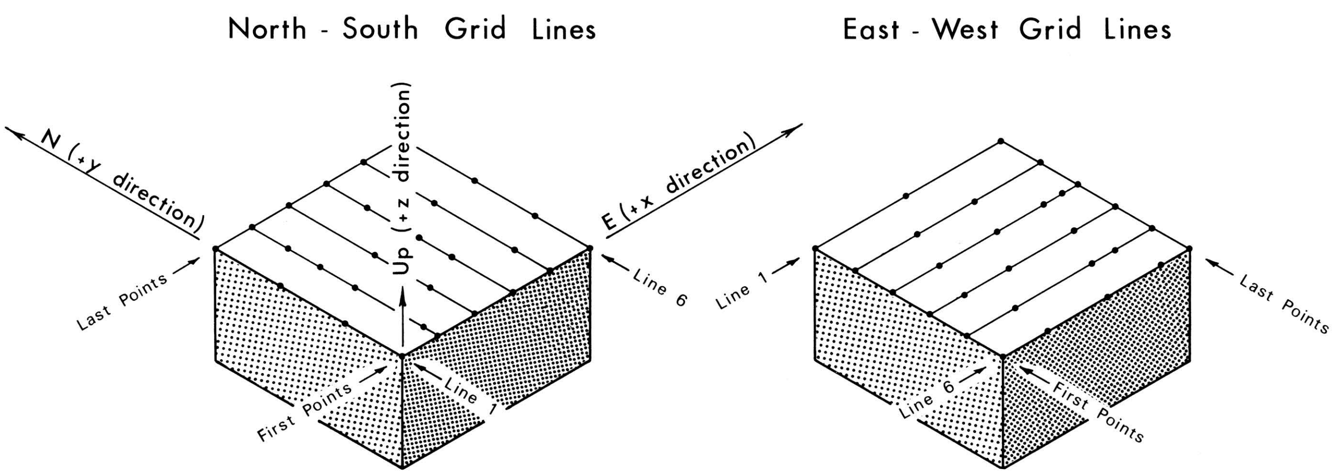

Figure 5. - Grid line and point conventions.

I3D SUBROUTINES - USAGE

Calling Program

The calling program, although variable in detail according to the needs of specific subroutines, must contain the following nonexecutable elements.

(a) JOB CARD, naming all input/output files required by the subroutines to be called, as well as any other needed files. File combinations required are given under each subroutine heading. There is no conflict of file names between subroutines.

(b) COMMON/WSPECS/RSPECS(30), ISPECS(10)

(c) Work array BUFI (LNGTHB), where LNGTHB is the maximum array length required by individual I3D subroutines. If LNGTHB is large, it is advantageous to place BUFI into unlabelled common where it will overlay the loader. A maximum of about 1680_{10} locations may be saved this way.

Subroutine I3D1GL

I3D1GL maps a three-dimensional surface, a single-valued function of two variables, into isometric or other similar pictorial representation. One direction of grid lines is used for shading and relief. Because of this subroutine's importance it is discussed first, although in the calling sequence it would be called after other I3D subroutines. It can also be used alone (see Fig. 6).

A. JOB CARD with possible file combinations.

(1) PROGRAM MAIN (OUTPUT, TAPE83, TAPE99)

For use when point data already exists on TAPE83. Point data logical records must not be preceded on TAPE83 by other logical records. TAPE99 is the logical tape unit for recording construction codes generated by the plotting subroutines.

(2) PROGRAM MAIN (INPUT, OUTPUT, TAPEi = INPUT, TAPE83, TAPE99)

For use when the logical records are punched on binary cards (i.e. from I3DCVT, I3DPTA, or other routines). Your main program must read the records off the INPUT file (TAPEi) and transfer them to TAPE83 prior to calling I3D1GL.

B. Required specifications. These must be declared

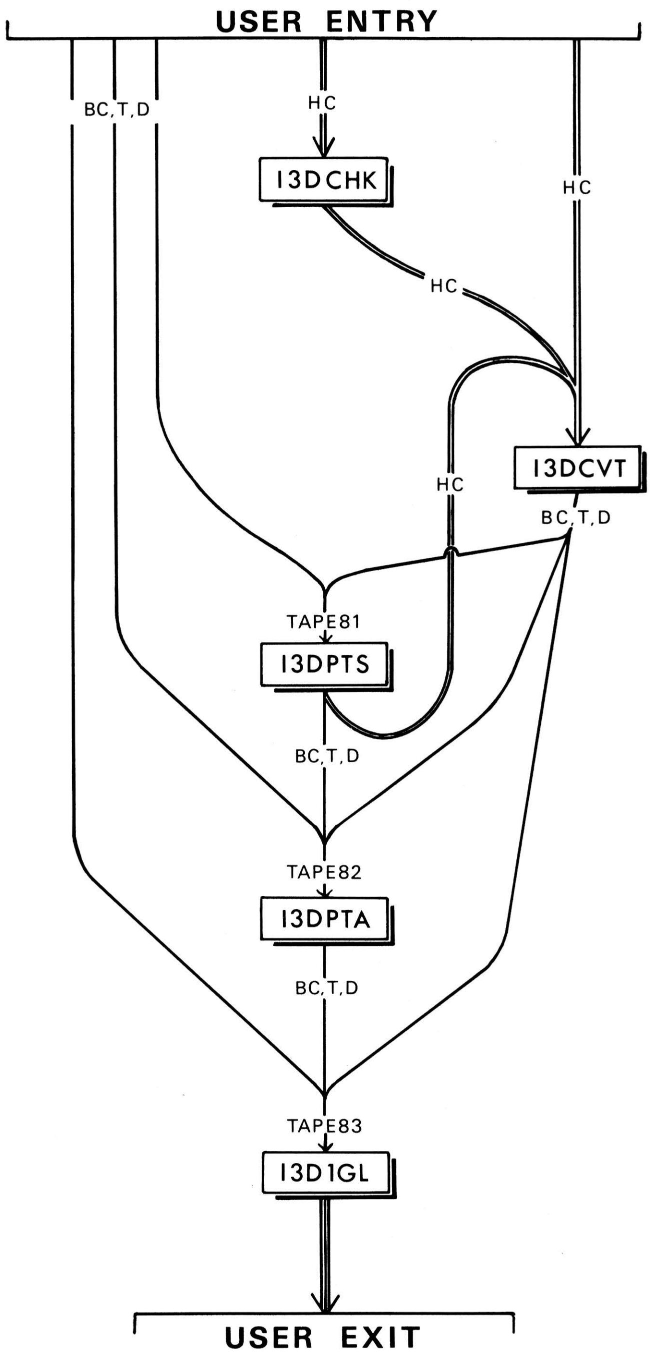

Figure 6.- User flowsheet guide to use of I3D subroutines. HC means Hollerith cards, BC means binary cards, T means magnetic tape, and D means disk storage.

in the main program prior to calling I3D1GL.

RSPECS (1-5), (7), (10-23), (28-30)

Depending on options chosen, some of these do not have to be declared. See RSPECS listing for details.

RSPECS (24-27) are returned from I3D1GL and may be used for labelling.

ISPECS (2), (5-10)

C. FORTRAN calling sequence.

CALL I3D1GL (BUFI, LNGTHB, SPECS)

LNGTHB must be set ⩾ 4*ISPECS(5) + 2*ISPECS(7)*ISPECS(8) + ISPECS(7) + ISPECS(8) + larger of ISPECS(7), ISPECS(8) + 2*ISPECS(2) + 3

D. Other information required.

Certain information must be entered into portions of the array BUFI prior to calling I3D1GL. Array addresses in BUFI are tabulated as follows.

AA1 = 4*ISPECS(5) + 1
AA2 = AA1 + ISPECS(7)*ISPECS(8) - 1
AA3 = AA2 + ISPECS(7)*ISPECS(8) + 1
AA4 = AA3 + ISPECS(7)
AA5 = AA4 + 1
AA6 = AA5 + ISPECS(8)

(1) That portion of BUFI array from (AA1) to (AA2) inclusive is used to determine if a particular rectangular block of data points is to be plotted or skipped. Blocks are bounded by CPC and CTC coordinates (see Fig. 7). Block number 1 is always in the southwest corner of the subarea to be considered for plotting, regardless of the angle-of-sight, RSPECS(14), or direction of grid lines, ISPECS(6). Block number 2 is always in the next position in the transverse direction, and so on as illustrated. If block i is to be plotted, set BUFI (AA1 + i - 1) = 1. If not, set it =0.

(2) Those portions of BUFI from (AA3) to (AA4) and from (AA5) to (AA6) are used to determine the bounds of blocks within the subarea considered for plotting. The first portion can be thought of as equivalent to a dummy array CTC of length ISPECS(7) + 1, and the second portion can be thought of similarly as a dummy array CPC of length ISPECS(8) + 1. CPC(1) is always the smallest parallel direction coordinate, and so on in increasing order if grid lines are north-south. If grid lines are east-west, CTC(1) is the largest transverse direction coordinate and so on in decreasing order. Figure 7 illustrates the situation for east-west grid lines.

Some restrictions exist on possible values for CTC and CPC. All CTC(i) must be equal to the transverse-axis coordinate of a grid line. Furthermore, if ISPECS(9) > 1, that is if lines are to be skipped at execution, additional caution is needed. The first block may start on any grid line, which thus becomes the first line, but all succeeding block CTC(i) must lie on a drawn (not skipped) grid line.

An exactly parallel situation exists for CPC.

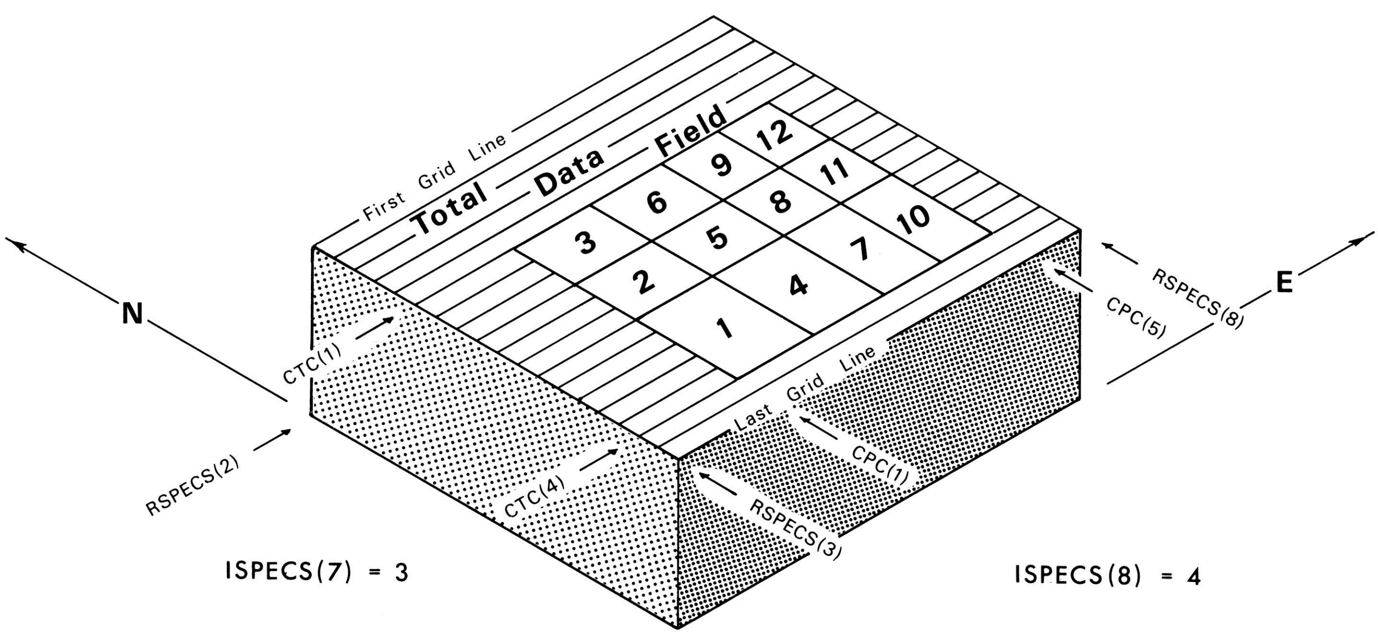

Figure 7.- Block diagram showing a possible relationship of input data points to drawn subareas. See text for explanation.

All CPC(i) must be equal to the parallel-axis coordinate of a point on a grid line and if ISPECS(10) > 1, the same caution must be exercised.

Subroutine I3DCHK

I3DCHK is a small, independent subroutine which can be used to catch gross keypunching errors and data cards out of place in a Hollerith data deck. Basic and skeleton data points, in order to be acceptable to I3DCVT, must satisfy the following relationships.

(1) Points are arranged in data grid lines.

(2) All lines must begin and end by data points with parallel-axis coordinates RSPECS(3) and RSPECS(8) respectively.

(3) Each data point is represented by a pair of coordinates, (y, z) if north-south grid lines or (x, z) if east-west grid lines. The north (or east) coordinate must be monotonically increasing (e.g. $x_{i+1} > x_i$ for all i).

(4) Each grid line (logical record) must be terminated by a user-option special end-of-record (EOR) punch in the last north (or east) coordinate position on a data card.

(5) The set of logical records is terminated by a 'double' EOR, the identical EOR punch in the z-coordinate position as well.

(6) Any configuration of blank cards or spaces is allowed. This permits easy data change or update. Note that, because of this feature, any point with a parallel-axis coordinate of 0. is ignored.

A. JOB CARD with files required.

(1) PROGRAM MAIN (INPUT, OUTPUT)

B. Required specifications.

RSPECS (3), (8-9)
ISPECS (1), (6)

C. FORTRAN calling sequence.

CALL I3DCHK (BUFI, LNGTHB, NUMPTC, FMT, EOR)
LNGTHB must be set ⩾ 2*NUMPTC
NUMPTC is the number of data point pairs per data card. Maximum value is thus 40.
FMT represents up to 25 Hollerith characters describing the format of the data on a card. Example: 16H(10X, 6(F6.3, F5.1)), 16 Hollerith characters.
EOR is the special (and unique) number to be used as an end-of-record and end-of-file marker. The use of this is described above.

Subroutine I3DCVT

I3DCVT reads a Hollerith data deck, possibly with embedded blanks, and reduces the data to a set of binary logical records, on TAPE81, TAPE82, or TAPE83 as specified in the calling list, suitable for direct input into I3DPTS, I3DPTA, or I3D1GL respectively. Remember to rewind or make certain the tape unit is at the load point prior to calling I3DCVT. Any of the units can be equivalenced to PUNCHB in the JOB CARD, thus permitting a binary deck to be punched. After calling I3DCVT, if I3DPTS or I3DPTA is to be called from this main program, you must rewind the tape unit to just before the first logical record written.

A. JOB CARD with possible file combinations. See Figure 6 also.

(1) PROGRAM MAIN (INPUT, OUTPUT, TAPEi)
i = 81, or 82, or 83

For use when I3DPTS, I3DPTA, or I3D1GL is called from the same main program, or when the data logical records are written onto magnetic tape for later use.

(2) PROGRAM MAIN (INPUT, OUTPUT, PUNCHB, TAPEi = PUNCHB)

For use if data are to be punched on binary cards for later use.

B. Program specifications.

ISPECS (1-3)

C. FORTRAN calling sequence.

CALL I3DCVT (BUFI, LNGTHB, NUMPTC, FMT, EOR,

$\begin{bmatrix} \text{6HTAPE81} \\ \text{6HTAPE82} \\ \text{6HTAPE83} \end{bmatrix}$, $\begin{bmatrix} \text{6HPNCHOT} \\ \text{6HNOPNCH} \end{bmatrix}$)

LNGTHB must be set ⩾ 4*ISPECS(1) + ISPECS(2)
NUMPTC, FMT, EOR are described in the section on FORTRAN calling sequence of I3DCHK.
The tape parameter, which must be one of the three given, must correspond to the tape unit declared on the JOB CARD.
The punch parameter, which must be one of the

two given, determines whether or not a binary record, the array NUMPT, will be punched, each word of which contains in order the number of points on a single grid line. Word 1 corresponds to grid line 1, and so on. There are three possible situations which may arise.

(1) I3DPTS, I3DPTA, or I3D1GL will be called from the same main program. No data binary cards are to be punched and the punch parameter chosen must be 6HNOPNCH.

(2) No other I3D subroutine will be called now, TAPEi is equivalenced to PUNCHB in the JOB CARD, and data binary cards are to be punched. If i is equal to 81 or 82, choose 6HPNCHOT as the punch parameter; the last logical record punched will be the array NUMPT.

(3) No other I3D subroutine will be called now. The data logical records are to be stored on magnetic tape equivalenced to the files TAPE81, TAPE82, or TAPE83. If tape unit is 81 or 82, choose 6HPNCHOT as the punch parameter; the array NUMPT will be written as the last record on the tape.

Subroutine I3DPTS

I3DPTS reads an array of data points on basic grid lines and calculates points on any number of intermediate lines, resulting in an array of points on skeleton grid lines. Points on basic grid lines may be regularly or irregularly spaced, and the number of points per line may range between the limits 2 and ISPECS(1). Each line, however, must begin and end with fixed parallel-axis coordinates RSPECS(3) and RSPECS(8) respectively, and each successive point parallel-axis coordinate must be greater than the preceding one, within a given grid line (e.g. $x_{i+1} > x_i$ for all i).

I3DPTS uses linear interpolation between points on adjacent basic grid lines to calculate points on intermediate lines. I3DPTS is unusual, however, in that angularities between horizontal and inclined portions of the data surface are preserved in calculation of intermediate line points. Ordinary linear interpolation of irregularly spaced points would generally result in intermediate lines showing rounded corners where they should be angular. This may be of some importance in the representation of artificially created topography (see Example 2) and other surfaces. Data points ideally need only be entered at breaks in slope, thus minimizing the number of points needed to represent the surface fully.

The algorithm for preserving angularities is not perfect, and certain extreme and generally unusual basic data point arrangements will cause aberrations. Perfection may be difficult to achieve, in part because of the problem of unintentional angularity situations created by a normally rolling surface. As a guide to proper selection of data-point positions to minimize plot aberrations, the basic algorithm is given. Numbers and letters correspond to numbers and letters in comment cards in I3DPTS listing. If you are not concerned with angularities and wish a simple linear interpolation, there is little need to study the algorithm, except to note the importance of DELTAD, derived from RSPECS(6), in 2iv.

<u>I3DPTS angularity algorithm</u>

(1) Consider a first grid line A and a second, adjacent, grid line B. Consider a point i on line A.

(2) Consider the point j on line B last used for interpolation and the next point j + 1.
- (i) If both B_j and B_{j+1} are of the same elevation (z-coordinate) as A_i, then use both with A_i to form two new grid points for each intermediate line (linear interpolation).
- (ii) If B_j is not at the same elevation as A_i but B_{j+1} is, then consider B_{j+2}.
  - (a) If B_{j+2} is at the same elevation as B_{j+1} and A_i, and at least one of B_{j+1}, B_{j+2} is within DELTAD parallel-axis distance of A_i (see RSPECS(6) description), then use B_{j+1}, for linear interpolation.
  - (b) If B_{j+2} is not at the same elevation, and if at least one of B_j, B_{j+1} is within DELTAD of A_i, then use B_j, B_{j+1} for linear interpolation.
- (iii) If B_j is at the same elevation as A_i, but B_{j+1} is not, and if at least one of B_j, B_{j+1} is within DELTAD of A_i, then use B_j, B_{j+1} for linear interpolation.
- (iv) If neither B_j nor B_{j+1} is at the same elevation as A_i:
  - (a) Test additional B points to see if there are two consecutive B points, at least one of which is within DELTAD of A_i, that are both equal to the elevation of A_i. If so, use the two B points for linear interpolation. If not, go on to b.
  - (b) Starting with B_j, consider, in order, up to a maximum of two B points for linear interpolation. Each B point so used must be within DELTAD of A_i.
- (v) Cycle until all A points and all B points are exhausted.

If I3DPTS calculates erroneous points and these prove objectionable when plotted, they may be corrected as follows. Run I3DPTS a second time, specifying a Hollerith format in the argument list.

A complete skeleton-array data deck will be punched, and the point coordinates in error can be readily corrected. Rerun the corrected deck, calling I3DCVT, I3DPTA, and I3D1GL.

I3DPTS presents one general approach to the problem of calculating additional data lines. Other algorithms may prove equally or better suited to your particular application. The I3D subroutines are structured so that substitution of one or several other subroutines as needed should be fairly simple and straightforward.

A. JOB CARD with possible file combinations. See Figure 6 also.

(1) PROGRAM MAIN (OUTPUT, TAPE81, TAPE82)

For use when basic point data to be read are contained on TAPE81, generally direct from I3DCVT, and I3DPTA will be called next from the current main program, or the calculated skeleton point array logical records are to be written onto magnetic tape for later use.

(2) PROGRAM MAIN (INPUT, OUTPUT, TAPE81=INPUT, TAPE82)

For use when basic point data to be read are contained in file INPUT, that is when a binary card deck is to be read.

(3) PROGRAM MAIN ($\begin{bmatrix} \text{———} \\ \text{INPUT} \end{bmatrix}$, OUTPUT, PUNCHB, $\begin{bmatrix} \text{TAPE81} \\ \text{TAPE81=INPUT} \end{bmatrix}$, TAPE82= PUNCHB)

For use when the skeleton point array calculated by I3DPTS is to be output as punched binary cards, one logical record per grid line.

(4) PROGRAM MAIN ($\begin{bmatrix} \text{———} \\ \text{INPUT} \end{bmatrix}$, OUTPUT PUNCH, $\begin{bmatrix} \text{TAPE81} \\ \text{TAPE81=INPUT} \end{bmatrix}$)

For use when the skeleton point array is to be output as punched Hollerith cards, according to the format specified in the CALL I3DPTS argument list. Terminating each set of cards which contains the coordinate-pairs of all points on a single grid line is a blank card.

(5) PROGRAM MAIN ($\begin{bmatrix} \text{———} \\ \text{INPUT} \end{bmatrix}$, OUTPUT, PUNCH, PUNCHB, $\begin{bmatrix} \text{TAPE81} \\ \text{TAPE81=INPUT} \end{bmatrix}$, TAPE82=PUNCHB)

For use as number 4, but in addition at the end a binary logical record is punched, each word of which contains the number of points on a single grid line.

B. Program specifications.

RSPECS (1), (6)
ISPECS (1-4)

C. FORTRAN calling sequence.

CALL I3DPTS (BUFI, LNGTHB, $\begin{bmatrix} \text{6HPNCHOT} \\ \text{6HNOPNCH} \end{bmatrix}$, $\begin{bmatrix} \text{6HNOHOLP} \\ \text{FMT, NUMPTC} \end{bmatrix}$)

LNGTHB must be set $\geq$ 4*ISPECS(1) + ISPECS(2) + 2*ISPECS(3)*ISPECS(4)

The punch parameter, which must be either 6HNOPNCH or 6HPNCHOT, determines whether or not a binary record, the array NUMPT, will be punched, each word of which, in order, contains the number of points on a single skeleton grid line. Word 1 in the array corresponds to grid line 1, and so on. There are four possible situations which may arise.

(1) I3DPTA will be called from the same main program. No data binary cards are to be punched and the punch parameter chosen must be 6HNOPNCH.

(2) I3DPTA will not be called now, and the data logical records are to be stored on magnetic tape equivalenced to file TAPE82. Choose 6HPNCHOT as the punch parameter; the array NUMPT will be written as the last record on the tape. For later entry into I3DPTA, this record must be read from tape into the work array BUFI at location SA3 prior to calling the subroutine.

(3) I3DPTA will not be called now, TAPE82 is equivalenced to PUNCHB in the JOB CARD, and data logical records are to be punched. Choose 6HPNCHOT as the punch parameter; the NUMPT array will be punched as the last logical record. Note that, for later entry into I3DPTA, this last record must be placed first and a single READ executed prior to calling the subroutine, to place the record into the work array BUFI beginning at location SA3.

(4) Hollerith skeleton-array data cards are to be punched according to the format specified in the subroutine argument list.

(a) If NUMPT array is desired (note that it is not necessary for entry into I3DCVT), specify 6HPNCHOT. The record is written in binary at the end of the Hollerith deck.
(b) If NUMPT array is not desired, specify 6HNOPNCH.

The format parameter must be either 6HNOHOLP or a combination of FMT and NUMPTC in the order given. If 6HNOHOLP is specified, then no Hollerith data cards are punched. If a Hollerith format is given (see the section on FORTRAN calling sequence of I3DCHK for explanation), then each grid-line logical record is punched in Hollerith mode on cards, with a blank card separating records. Note that all EOR punches must be added afterwards. A word of caution: Hollerith output is punched with the decimal point included, so be certain to include the decimal point in the format declaration. Example: the number 32500. can be read in under format F5.0, but must be punched under format F6.0.

D. Other information required.

If data point logical records are on binary cards or magnetic tape, and I3DCVT has not been called from this main program, then part of the work array BUFI must be filled with the array NUMPT as punched on cards or written on magnetic tape from I3DCVT. Location AA1 within BUFI is equal to 4* ISPECS(1) + 1.

(a) If the data are on binary cards, take the last logical record punched from I3DCVT and place it first in the data deck. NUMPT must be read into the work array BUFI beginning at location AA1, prior to calling I3DPTS. Because ISPECS(2) is, and must be, greater than the number of basic grid lines, the words in this first logical record must be spaced in BUFI accordingly. An example will make this clear.

```
     Example: Number of basic grid
              lines = 101
              ISPECS(2) = final number
                          of grid lines
                          = 301
                          (ISPECS(3)=2)
              and say ISPECS(1) = 200
     READ (81) (BUFI(I), I=801, 1101, 3)

     CALL I3DPTS (BUFI, LNGTHB, . . .)
```

(b) If the data are on magnetic tape, the procedure is identical except NUMPT will be read directly from the tape.

Subroutine I3DPTA

I3DPTA reads an array of data points on skeleton grid lines and calculates a final, or filled-out, orthogonal array of points suitable for direct input into I3D1GL. Points read are defined by two coordinates, elevation and parallel-axis, whereas points calculated within I3DPTA are defined by only one coordinate, elevation, because the other two may be calculated as needed. Linear interpolation is used to calculate final points.

A. JOB CARD with possible file combinations.

(1) PROGRAM MAIN (OUTPUT, TAPE82, TAPE83)

For use when skeleton point data to be read are contained on TAPE82, generally direct from I3DPTS, and I3D1GL will be called next from the current main program, or the calculated filled-out point array logical records are to be written onto magnetic tape for later use.

(2) PROGRAM MAIN (INPUT, OUTPUT, TAPE82=INPUT, TAPE83)

For use when skeleton point data to be read are contained in file INPUT, that is when a binary card deck is to be read.

(3) PROGRAM MAIN ([— / INPUT], OUTPUT, PUNCHB, [TAPE82 / TAPE82=INPUT], TAPE83=PUNCHB)

For use when the filled-out point array is to be output as punched binary cards, one logical record per grid line.

B. Program specifications.

RSPECS(7)
ISPECS(2), (4-5)

C. FORTRAN calling sequence.

CALL I3DPTA (BUFI, LNGTHB)
LNGTHB must be set $\geq$ 2* ISPECS(4) + ISPECS(2) + ISPECS(5)

D. Other information required.

As in the case with I3DPTS, if data point logical records are contained in binary cards or magnetic tape, and I3DPTS or I3DCVT has not been called from this main program, then part of the work array BUFI beginning at location 2* ISPECS(4) + 1 must be filled with the complete array NUMPT as punched on cards or written on magnetic tape from I3DCVT or I3DPTS. The procedure is exactly analogous to that described in section D. of I3DPTS, except that the NUMPT array is complete so that, in the example given, the READ statement should be

READ (82) (BUFI(I), I=801, 1101).

EXAMPLES

Three examples are included to illustrate use of I3D subroutines. The calling programs show the parameters used. Subroutine listings are given with Examples 1 and 2 (see Appendix).

Example 1 illustrates use of I3DCHK. Errors were introduced into the first seven grid lines of a data deck to show the types of diagnostic messages printed. Figure 8 reproduces the printed output, and Figure 9 is a listing of the complete data deck.

Example 2 illustrates use of I3DCVT, I3DPTS, I3DPTA, and I3D1GL called from the same main program. Seventy-six basic grid lines of data points were read in, containing a total of approximately 4500 (north, elevation) points taken at breaks in slope. From these initial points, 121,102 final points were calculated using I3DPTS and I3DPTA, and these points were transformed into the isometric diagram shown in Figure 10, by I3D1GL. Execution times (central processor) for the various subroutines are:

I3DCVT	3.9 seconds
I3DPTS	4.4 seconds
I3DPTA	13.3 seconds
I3D1GL	61.0 seconds

The plotted surface resembles the actual surface, and elevation variations of only a few feet (at this scale) can be seen easily. Attention is drawn to four east-west streets in the southerly block.

Example 3 (Appendix) illustrates use of I3D1GL alone in portraying a mathematical function. This particular function (see the example main program listing) was chosen to show the effect of sharp peaks in the data (Figure 11). I3D1GL execution time was 12.6 seconds.

All examples were run on a Berkeley Computer Center CDC 6400 computer with disk storage unit. Plotting was done on a 31-inch CalComp plotter monitored by an IBM 360/40. All plotter software were written at Berkeley.

```
 SUBROUTINE I3DCHK

  NORTH COORD. OF FIRST POINT ON N-S LINE IS    32000.0000
  NORTH COORD. OF LAST POINT ON N-S LINE IS    32800.0000
  MAXIMUM ALLOWABLE NUMBER OF POINTS PER N-S LINE IS    110
  IF SUCCEEDING ELEVATIONS DIFFER BY MORE THAN      67.0000, AN ERROR MESSAGE IS PRINTED


THE NORTH COORD. OF N-S LINE NUMBER    3, CARD     1, POINT  3, IS INCORRECT.
        CHECK ALSO THE ELEVATION COORDINATE.

FIRST POINT ON N-S LINE NUMBER     4 IS INCORRECT

THE NORTH COORD. OF N-S LINE NUMBER    5, CARD     2, POINT  1, IS INCORRECT.
        CHECK ALSO THE ELEVATION COORDINATE.

THE ELEVATION COORD. OF N-S LINE NUMBER    5, CARD     2, POINT  1, IS INCORRECT.

THE ELEVATION COORD. OF N-S LINE NUMBER    5, CARD     3, POINT  1, IS INCORRECT.

THE ELEVATION COORD. OF N-S LINE NUMBER    5, CARD     3, POINT  2, IS INCORRECT.

FOR N-S LINE NUMBER    6, EITHER THE FLAG ELEVATION OR THE PRECEEDING POINT ELEVATION IS MISPUNCHED.
        NO TEST IS MADE TO CHECK LAST INPUT POINT NORTH COORDINATE.

THE NORTH COORD. OF N-S LINE NUMBER    7, CARD     2, POINT  4, IS INCORRECT.
        CHECK ALSO THE ELEVATION COORDINATE.

LAST POINT ON N-S LINE NUMBER    7 IS INCORRECT.


NUMBER OF BASIC GRID LINES =    26
NUMBER OF BASIC GRID POINTS =     349
```

Figure 8.- Printed output from I3DCHK, using data given in Figure 9.

```
32000 575032059 577032141 576032220 574032360 572032479 570032598 563332800 5633
                                                                      99999
32000 574532045 577032130 575632200 5734 32300 571532370 570332443 567032520 5633
32618 5633         32800 5633                                         99999
32000 574032039 576530180 575032180 572032269 569032338 566232406 563332521 5633
32577 560032750 560032800 5600                                        99999
32477 560032604 560032640 556732800 5567                              99999
32000 573532016 576032090 573032152 570032210 568032240 566232280 563332432 5633
32000 573032069 571532121 568032180 566532208 563332311 563332374 560032515 5600
32540 576732600 556732800 5567                                        99999
32000 572032048 570032150 566032179 563332240 563332260 560332290 560032417 5600
32456 556732520 556732800 5567                                        99999 5567
32000 571532071 568032130 566032152 563332200 563332208 561232242 560032323 5600
32359 556732420 556732750 556732700 5567                              99999
32000 570532033 568032115 565732135 563332174 563332208 562532212 560032270 5600
32300 556732360 556732684 556732723 553332782 553332800 5544          99999
32000 568032050 569032106 565332123 563332159 563332180 563032186 560032240 5600
32261 556732317 556732655 556732700 553332800 5533                    99999
32000 568032041 569032096 565032117 563332147 563332163 560032210 560032237 5567
32344 556732430 553332561 553332601 553332700 553332800 5533          99999
32000 569032040 568032090 564732104 563332134 563332150 560032191 560032219 5567
32296 556732338 553332420 553332660 553332727 553332800 5533          99999
32000 569532080 564532091 563332121 563332139 560032178 560032200 556732266 5567
32300 553332377 553332414 550032760 550032786 553332800 5533          99999
32000 567532032 565532069 564032080 563332118 563332130 560032162 560032192 5567
32258 556732290 553332355 553332379 550032800 5500                    99999
32000 565532065 563832067 563332106 563332120 560032156 560032180 556732250 5567
32281 553332340 553332370 550032590 550032800 5500                    99999
32000 564532058 563532060 563332091 563332109 560032143 560032172 556732250 5567
32279 553332338 553332370 550032500 550032558 546732630 546732710 550032800 5500
                                                                      99999
32000 564032037 563532041 563332082 563332100 560032141 560032167 556732250 5567
32278 553332338 553332361 550032451 550032478 547032482 546732540 546732660 5467
32730 546732760 550032800 5500                                        99999
32000 563532016 563332070 563332081 560032130 560032156 556732250 556732280 5533
32339 553332360 550032442 550032450 547532478 546732761 546732782 550032800 5500
                                                                      99999
32000 563332050 563332068 560032120 560032139 556732250 556732279 553332336 5533
32359 550032435 550032440 548532463 546732779 546732800 5496          99999
32000 563332039 563332050 560032102 560032120 556732242 556732270 553332322 5533
32357 550032430 550032436 549032459 547132463 546732770 546732800 550099999
32000 563332022 563332035 560032088 560032114 556732222 556732249 553332317 5533
32340 550032431 550032453 548032460 546732748 546732780 550032800 550099999
32000 563332018 563332025 560032080 560032101 556732190 556732220 553332296 5533
32322 550032439 550032442 549632458 546732726 546732769 550032800 550099999
32000 563332010 563332021 560032072 560032093 556732162 556732198 553332270 5533
32300 550032440 550032466 546732692 546732728 550032800 5500          99999
32000 563332008 563332018 560032070 560032089 556732157 556732182 553332251 5533
32280 550032459 550032483 546732650 546732695 550032779 550032800 552599999
32000 563332018 560032066 560032088 556732157 556732186 553332250 553332278 5500
32482 550032540 546732593 546732647 550032742 550032781 553332800 553399999
32000 562832016 560032064 560032083 556732160 556732200 553332269 553332290 5500
32558 550032700 550032737 553332800 5533                              99999
32000 562832018 560032062 560032083 556732180 556732220 553332320 553332350 5500
32613 550032670 553332760 553332790 556732800 5567                    9999999999
```

Figure 9.– Listing of data used for Example 1.

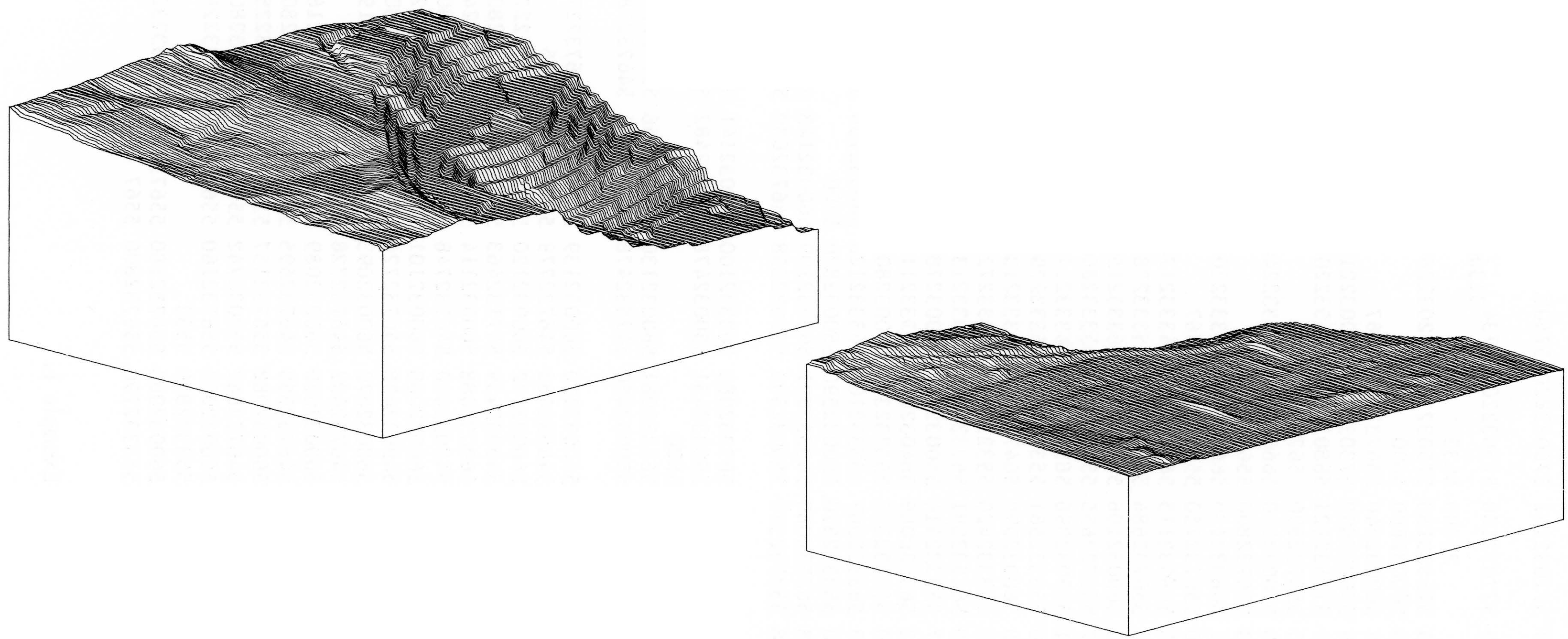

Figure 10.- View of a portion of the Berkeley open-pit mine, Butte, Montana

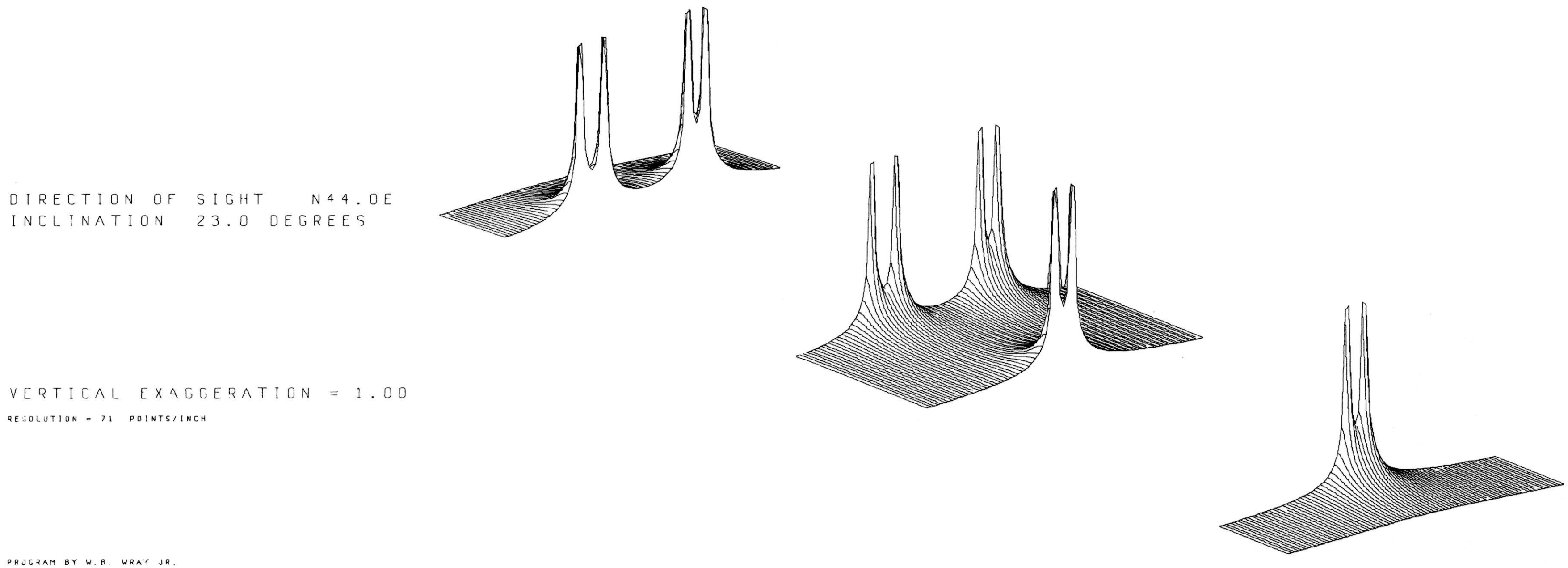

Figure 11.- View of mathematical function (see listing).

REFERENCES

Lobeck, A.K., 1924, Block diagrams: John Wiley & Sons, Inc., New York, 206 p.

Rohlf, F.J., 1969, GRAFPAC, graphic output subroutines for the GE 635 computer: Kansas Geol. Survey Computer Contr. 36, 50 p.

APPENDIX

Example 1

```
PROGRAM MAIN (INPUT,OUTPUT)
COMMON /WSPECS/ RSPECS(30),ISPECS(10)
RSPECS (3) = 32000.
RSPECS(8) = 32800.
RSPECS (9) = 67.
ISPECS(1) = 110
ISPECS(6) = 1
CALL I3DCHK (8,8H(16F5.0),99999.)
STOP
END
```

```
      SUBROUTINE I3DCHK (NP,FMT,EOR)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      COMMON /TIME/ TO
      REAL COORDN(40),COORDZ(40)
      DIMENSION FMT(3),LDIR1(2),LDIR2(2)
      EQUIVALENCE (RSPECS(3),START), (RSPECS(8),FINISH), (RSPECS(9),ELDI
     1FF), (ISPECS(1),NPOINT), (ISPECS(6),I6)
      LOGICAL FRSTPT
      DATA TLBL /6HI3DCHK/
      DATA LDIR1(1),LDIR1(2),LDIR2(1),LDIR2(2) /3HN-S,3HE-W,5HNORTH,4HEA
     1ST/
C
C---  SUBROUTINE I3DCHK READS A BASIC GRID POINT DATA DECK AND CHECKS
C---  FOR GROSS ERRORS.  I3DCHK MAKES CERTAIN THAT (1) EACH RECORD STARTS
C---  WITH ≠START≠ COORDINATE AND ENDS WITH ≠FINISH≠, (2) EACH NORTH (OR
C---  EAST) COORDINATE ON A CERTAIN GRID LINE IS GREATER THAN THE PRECEEDING
C---  COORDINATE ON THE SAME LINE, (3) EACH GRID LINE ENDS WITH A SINGLE
C---  EOR AND THE LAST GRID LINE ENDS WITH A DOUBLE EOR, (4) THERE ARE
C---  NO MORE THAN ≠NPOINT≠ POINTS PER GRID LINE (A GOOD TEST FOR CATCHING
C---  MISSING EOR PUNCHES), AND (5) EACH ELEVATION COORDINATE ON A CERTAIN
C---  GRID LINE IS WITHIN +/- ≠ELDIFF≠ OF THE PRECEEDING ELEVATION ( A
C---  TEST TO CATCH GROSS ERRORS IN ELEVATION PUNCHES).  IF ERRORS ARE
C---  FOUND, APPROPRIATE ERROR MESSAGES ARE PRINTED.
C---  I3DCHK IGNORES BLANKS ON DATA CARDS AND USES VARIABLE CARD FORMAT,
C---  AS SPECIFIED BY USER.
C
      CALL SECOND (TO)
      PRINT 1
    1 FORMAT (1H1, 8X, *SUBROUTINE I3DCHK*/)
C
C---  INITIALIZE COUNTERS.  ICARD = NUMBER OF CARDS READ THIS GRID LINE.
C---  IN = NUMBER OF THE POINT ON THE CARD.  NUMPT = NUMBER OF REAL
C---  (NON-BLANK) POINTS IN THE GRID LINE.  NUMTOT = TOTAL NUMBER OF DATA
C---  POINTS (EXCLUDING FLAGS).  IEAST = NUMBER OF THE GRID LINE.
C
      IEAST = 1
      NUMTOT = ICARD = IN = NUMPT = 0
      FRSTPT = .TRUE.
```

```
      PRINT 4, LDIR2(I6),LDIR1(I6),START,LDIR2(I6),LDIR1(I6),FINISH,
     1LDIR1(I6),NPOINT,ELDIFF
    4 FORMAT (10X, A5, * COORD. OF FIRST POINT ON *, A3, * LINE IS *, F1
     12.4/10X, A5, * COORD. OF LAST POINT ON *, A3, * LINE IS *, F12.4/1
     20X, *MAXIMUM ALLOWABLE NUMBER OF POINTS PER *, A3, * LINE IS *, I5
     3/10X, *IF SUCCEEDING ELEVATIONS DIFFER BY MORE THAN *, F12.4, *, A
     4N ERROR MESSAGE IS PRINTED*///)
C
C---  ROUTINE TO CONSIDER THE FIRST NON-ZERO POINT OF THE CARD.
C
    5 READ FMT, (COORDN(I), COORDZ(I),I=1,NP)
      ICARD = ICARD + 1
      IF (.NOT. FRSTPT) GO TO 7
    9 IF (IN .LT. NP) GO TO 8
      IN = 0
      GO TO 5
    8 IN = IN + 1
      IF (COORDN(IN) .EQ. 0.) GO TO 9
C
C---  ROUTE OF THE FIRST NON-ZERO POINT OF A GRID LINE.
C
      IF (COORDN(IN) .EQ. START) GO TO 10
      PRINT 11, LDIR1(I6), IEAST
   11 FORMAT (8X, *FIRST POINT ON *, A3, * LINE NUMBER *, I4, * IS INCOR
     1RECT*/)
   10 FRSTPT = .FALSE.
      NUMPT = 1
      PTNSVE = COORDN(IN)
      PTZSVE = COORDZ(IN)
      IF (IN .EQ. NP) GO TO 5
      IN1 = IN + 1
      GO TO 12
C
C---  THIS ROUTE, CARD HAS JUST BEEN READ, BUT PAST THE FIRST POINT.
C
    7 IN1 = 1
C
C---  NOTE THAT HERE, PTNSAVE ETC CANNOT BE ZERO OR UNDEFINED.
C---  DO LOOP TO PROCESS BALANCE OF CARD.
C
   12 DO 13 I=IN1,NP
      IF (COORDN(I) .EQ. 0.) GO TO 13
      NUMPT = NUMPT + 1
      IF (COORDN(I) .GT. PTNSVE) GO TO 15
      PRINT 16, LDIR2(I6),LDIR1(I6),IEAST,ICARD,I
   16 FORMAT (8X, *THE *, A5, * COORD. OF *, A3, * LINE NUMBER *, I4, *,
     1 CARD *, I4, *, POINT *, I2, *, IS INCORRECT.*/16X, *CHECK ALSO TH
     2E ELEVATION COORDINATE.*/)
   15 IF (ABS(COORDZ(I) - PTZSVE) .LE. ELDIFF) GO TO 17
      IF (I .EQ. NP .AND. COORDN(I) .EQ. EOR) GO TO 13
      PRINT 18, LDIR1(I6), IEAST, ICARD, I
   18 FORMAT (8X, *THE ELEVATION COORD. OF *, A3, * LINE NUMBER *, I4,
     1*, CARD *, I4, *, POINT *, I2, *, IS INCORRECT.*/)
   17 PTNSVE = COORDN(I)
      PTZSVE = COORDZ(I)
   13 CONTINUE
C
C---  TESTS ON THE LAST NON-ZERO POINT OF THE CARD.
C
      IF (NUMPT .LE. NPOINT) GO TO 14
```

```
      PRINT 19, LDIR1(I6),IEAST,EOR
   19 FORMAT (8X, A3, * LINE NUMBER *, I4, * HAS TOO MANY POINTS, OR A *
     1, F12.4, * PUNCH IS MISSING.*/)
   14 IF (COORDN(NP) .LT. EOR) GO TO 5
      NUMPT = NUMPT - 1
      IF (PTNSVE .NE. EOR) GO TO 20
      PRINT 21,LDIR1(I6),IEAST,LDIR2(I6)
   21 FORMAT (8X, *FOR *, A3, * LINE NUMBER *, I4, *, EITHER THE FLAG EL
     1EVATION OR THE PRECEEDING POINT ELEVATION IS MISPUNCHED.*/16X, *NO
     2 TEST IS MADE TO CHECK LAST INPUT POINT *, A5, * COORDINATE.*/)
      GO TO 22
   20 IF (PTNSVE .EQ. FINISH) GO TO 22
      PRINT 23, LDIR1(I6),IEAST
   23 FORMAT (8X, *LAST POINT ON *, A3, * LINE NUMBER *, I4, * IS INCORR
     1ECT.*/)
   22 IF (COORDZ(NP) .EQ. EOR) GO TO 24
      ICARD = IN = 0
      IEAST = IEAST + 1
      FRSTPT = .TRUE.
      NUMTOT = NUMTOT + NUMPT
      GO TO 5
   24 NUMTOT = NUMTOT + NUMPT
      PRINT 25, IEAST, NUMTOT
   25 FORMAT (/8X, *NUMBER OF BASIC GRID LINES = *, I4/8X, *NUMBER OF BA
     1SIC GRID POINTS = *, I6/)
      CALL TYME (TLBL)
      RETURN
      END




      SUBROUTINE TYME(TLBL)
      COMMON /TIME/ TO
C
C---  TYME READS THE SYSTEMS CLOCK AND PRINTS OUT THE TIME SINCE THE
C---  FIRST CALL SECOND.
C
      CALL SECOND (A)
      T = A-TO
      PRINT 1, TLBL,T
    1 FORMAT (//105X, *TIME IN *, A6, * =*, F8.3/)
      RETURN
      END
```

Example 2

```
PROGRAM MAIN (INPUT,OUTPUT,TAPE81,TAPE82,TAPE83,TAPE99)
DIMENSION SPECS(30)
COMMON BUFI (3518)
COMMON /WSPECS/ RSPECS(30),ISPECS(10)
SPECS(1) = 1.
SPECS(2) = 0.5
SPECS(3) = 51.
SPECS(4) = 1.
SPECS(5) = 29.5
SPECS(6) = 0.5
SPECS(7) = 60.
SPECS(8) = 29.
SPECS(11) = 1.
SPECS(12) = 99.
CALL PLOT3D (SPECS)
ISPECS(1) = 110
ISPECS(2) = 151
ISPECS(3) = 1
ISPECS(4) = 220
ISPECS(5) = 801
ISPECS(6) = 1
ISPECS(7) = 1
ISPECS(8) = 2
ISPECS(9) = 1
ISPECS(10) = 1
RSPECS(1) = 20.
RSPECS(2) = 30000.
RSPECS(3) = 31200.
RSPECS(4) = 4600.
RSPECS(5) = 6000.
RSPECS(6) = 1.732
RSPECS(7) = 6.
RSPECS(10) = 400.
RSPECS(11) = 35.
RSPECS(12) = 2.
RSPECS(13) = 20.
RSPECS(14) = 45.
RSPECS(15) = 1.
RSPECS(16) = 0.
RSPECS(21) = 0.
RSPECS(28) = .5
RSPECS(29) = .5
RSPECS(30) = 1.
REWIND 81
REWIND 82
REWIND 83
CALL I3DCVT (BUFI,3518,8,8H(16F5.0),99999.,6HTAPE81,6HNOPNCH)
REWIND 81
CALL I3DPTS (BUFI,3518,6HNOPNCH,6HNOHCLP)
REWIND 82
CALL I3DPTA (BUFI,3518)
BUFI(3205) = BUFI(3206) = 1.
BUFI(3209) = 30000.
BUFI(3210) = 33000.
BUFI(3211) = 31200.
```

```
BUFI(3212) = 33420.
BUFI(3213) = 35000.
CALL I3D1GL (BUFI,3518,SPECS)
CALL GDSEND (SPECS)
STOP
END
```

```
      SUBROUTINE I3D1GL (BUFI,LNGTHB,SPECS)
      DIMENSION BUFI(1),SPECS(30)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      COMMON /TIME/TO
      DATA TLBL /5HI3D1GL/
      EQUIVALENCE (ISPECS(5),I5),(ISPECS(7),I7),(ISPECS(8),I8),(ISPECS(2
     1),I2)
C
C---  I3D1GL MAPS A THREE-DIMENSIONAL SURFACE - A SINGLE-VALUED FUNCTION
C---  OF TWO VARIABLES - INTO ISOMETRIC, CAVALIER, DIMETRIC, OR CABINET
C---  PICTORIAL REPRESENTATION.  ONE SET OF GRID LINES IS USED.
C
      CALL SECOND (TO)
      L1 = 4*I5     + 2*I7*I8 + I7 + I8 + 3 + MAX0(I7,I8) + 2*I2
      IF (LNGTHB .GE. L1) GO TO 1
      PRINT 2
    2 FORMAT (6X,*LNGTHB AS SET BY USER IS TOO SMALL FOR SUBROUTINE I3D1
     1GL.*)
      STOP
    1 N2 = 1 + I5
      N3 = N2 + I5
      N4 = N3 + I5
      N5 = N4 + I5
      N6 = N5 + I7*I8
      N7 = N6 + I7*I8
      N8 = N7 + I7 + 1
      N9 = N8 + I8 + 1
      N10 = N9 + 1 + MAX0(I7,I8)
      N11 = N10 + I2
      CALL EXC1GL (BUFI(1),BUFI(N2),BUFI(N3),BUFI(N4),BUFI(N5),BUFI(N6),
     1BUFI(N7),BUFI(N8),BUFI(N9),BUFI(N10),BUFI(N11),LNGTHB,SPECS)
      CALL TYME (TLBL)
      RETURN
      END
```

```
      SUBROUTINE EXC1GL (PLOTZ,XPLOT,YPLOT,YPLSVE,BLOCK,BLOCKH,CEC,CNC,
     1CH,BOTEND,TOPEND,LNGTHB,SPECS)
      REAL PLOTZ(1),XPLOT(1),YPLOT(1),YPLSVE(1),BLOCK(1),BLOCKH(1),
     1CEC(1),CNC(1),CH(1),BOTEND(1),TOPEND(1),NSTART,LABEL
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      EQUIVALENCE (ISPECS(6),I16),
     1(ISPECS(7),I7),(ISPECS(8),I8),(ISPECS(9),I9),(ISPECS(10),I10)
      EQUIVALENCE (RSPECS(1),SL),(RSPECS(2),ESTART),(RSPECS(3),NSTART),
     1(RSPECS(4),VSTART),(RSPECS(5),VTOP),(RSPECS(7),PS),(RSPECS(10),
     2SCALE),(RSPECS(11),APX),(RSPECS(12),APY),(RSPECS(13),GAMMA),
     3(RSPECS(14),PSIDIR),(RSPECS(15),FRAME),(RSPECS(16),TICKS),
     4(RSPECS(21),LABEL),(RSPECS(24),VDATAP),(RSPECS(25),XFRSVE),
     5(RSPECS(26),CNC111),(RSPECS(27),CEC111),(RSPECS(28),PLSEP),
     6(RSPECS(29),TLSEP),(RSPECS(30),ZFACTR)
      DIMENSION CNCOR(4),CECOR(4),SPECS(30),XFRAME(4),YFRAME(4)
      LOGICAL EVEN,SOUTH,WEST,UU
C
C---  CONVERT PSIDIR, TRUE LINE-OF-SIGHT, TO DUMMY PSI, WHICH DEPENDS
C---  ALSO IN DIRECTION OF GRID LINES.
C
      IF (I16 .EQ. 1) GO TO 3
      PSI = PSIDIR - 90.
      IF (PSI .GE. 0.) GOTO 4
      PSI = 360. + PSI
      GO TO 4
    3 PSI = PSIDIR
C
C---  DETERMINE SOUTH AND WEST FOR USE IN IF STATEMENTS BELOW.
C
    4 IF (PSI .GE. 90. .AND. PSI .LT. 270.) GO TO 66
      SOUTH = .FALSE.
      GO TO 67
   66 SOUTH = .TRUE.
   67 IF (PSI .GE. 180. .AND. PSI .LT. 360.) GO TO 68
      WEST = .FALSE.
      GO TO 69
   68 WEST = .TRUE.
C
C---  CONVERT GAMMA AND PSI TO RADIAN MEASURE.  CALCULATE ALPHA1 AND ALPHA2 AS
C---  FUNCTIONS OF PSI, GAMMA.
C
   69 RGAMMA = GAMMA*.0174532925
      IF (GAMMA .EQ. 0.) RGAMMA = .00001
      RPSI = 0.
      IF (PSI .EQ. 0.) RPSI = .00001
      IF (PSI .EQ. 90.) RPSI = 1.57081
      IF (PSI .EQ. 180.) RPSI = 3.14161
      IF (PSI .EQ. 270.) RPSI = 4.71241
      IF (RPSI .EQ. 0.) RPSI = PSI*.0174532925
      ALPHA2 = ASIN((SIN(RPSI)**2 + TAN(RGAMMA)**2 - COS(RPSI)**2)/(2.*S
     1IN(RPSI)*TAN(RGAMMA)))
      ALPHA1 = ASIN((TAN(RGAMMA) - SIN(RPSI)*SIN(ALPHA2))/COS(RPSI))
C
C---  SET CONSTANTS FOR USE IN RESETTING ≠BLOCK≠.
C

      IBB2 = I7*I8 - I7
      IBB3 = I7*I8 + 1
      IBB4 = I7 + 1
      IBBJ = 2*I7
```

```
C
C---   SET VARIABLES FOR RESETTING ≠BLOCK≠ ACCORDING TO QUADRANT OF PSI.
C
       IF (PSI .GE. 90. .AND. PSI .LT. 180.) GO TO 170
       IF (PSI .GE. 180. .AND. PSI .LT. 270.) GO TO 171
       IF (PSI .GE. 270. .AND. PSI .LT. 360.) GO TO 172
       IF (II6 .EQ. 2) 8,7
   172 ALPHA1 = 3.14159265 - ALPHA1
       ALPHA2 = 3.14159265 + ALPHA2
       IF (II6 .EQ. 2) GO TO 7
     8 IB = IBB4
       IB1 = -1
       IBJMP = IBBJ
       GO TO 173
     7 IB = 0
       IB1 = 1
       IBJMP = 0
       GO TO 173
   171 ALPHA1 = -ALPHA1
       ALPHA2 = -ALPHA2
       IF (II6 .EQ. 2) 6,5
   170 ALPHA1 = 3.14159265 + ALPHA1
       ALPHA2 = 3.14159265 - ALPHA2
       IF (II6 .EQ. 2) GO TO 5
     6 IB = IBB2
       IB1 = 1
       IBJMP = -IBBJ
       GO TO 173
     5 IB = IBB3
       IB1 = -1
       IBJMP = 0
C
C---   REORDER CEC, CNC AS NECESSARY.
C
   173 I81 = I8 + 1
       IF (SOUTH) GO TO 143
       IF (PSI .GE. 270. .AND. PSI .LT. 360.) GO TO 144
       GO TO 145
   143 DO 149 I=1,I81
       IH = I81 - I + 1
   149 CH(IH) = CNC(I)
       DO 150 I=1,I81
   150 CNC(I) = CH(I)
       IF (PSI .GE. 180. .AND. PSI .LT. 270.) GO TO 144
       GO TO 145
   144 DO 154 I=1,IBB4
       IH = IBB4 - I + 1
   154 CH(IH) = CEC(I)
       DO 155 I=1,IBB4
   155 CEC(I) = CH(I)
C
C---   DEFINE SIN AND COS VARIABLES FOR USE LATER.
C
   145 SALPH2 = SIN(ALPHA2)
       SALPH1 = SIN(ALPHA1)
       CALPH2 = COS(ALPHA2)
       CALPH1 = COS(ALPHA1)
C
C---   REORDER BLOCK AND PLACE INTO BLOCKH USING VARIABLES SET PRIOR.
```

```
C---  BLOCKH(1) WILL REFER TO THE DIAGRAM AT THE APEX, ETC.
C
      IDE = 0
      I78 = I7*I8
      DO 9  I=1,I78
      IB = IB + IB1
      IDE = IDE + 1
      IF (IDE .LE. I7) GO TO 10
      IDE = 0
      IB = IB + IBJMP
   10 BLOCKH(I) = BLOCK(IB)
    9 CONTINUE
C
C---  INITIALIZE SYSTEM AND CALCULATE EQUATION CONSTANTS TO BE USED IN
C---  MAIN DO LOOPS BELOW.
C
      I91 = I9 + 1
      I9M1 = I9 - 1
      EVEN = .FALSE.
      I10M = I10 - 1
      I10I = 1-I10
      CNC111 = CNC(1)
      CEC111 = CEC(1)
      C = (VTOP-VSTART)*ZFACTR/SCALE
      HEW = C/(CALPH2*TAN(ALPHA1) + SALPH2) + TLSEP
      HNS = C/(CALPH1*TAN(ALPHA2) + SALPH1) + PLSEP
      XHEW = CALPH2*HEW
      YHEW = SALPH2*HEW
      XHNS = CALPH1*HNS
      YHNS = SALPH1*HNS
      XPART = CALPH1*PS/SCALE
      YPART = SALPH1*PS/SCALE
      XLINEP = SL*CALPH2/SCALE
      YLINEP = SL*SALPH2/SCALE
      XFRSVE = 1000.
      IB = 0
      IBCNT = 0
C
C---  MAIN DO LOOPS TO PROCESS ONE BLOCK DIAGRAM AT A TIME.
C
      DO 11 NI = 1,I8
      DO 12 NN1 = 1,I7
      REWIND 83
      IB = IB + 1
      IF (BLOCKH(IB) .NE. 1.) GO TO 12
      IBCNT = IBCNT + 1
C
C---  CALCULATE CNCOR AND CECOR ARRAYS FROM CURRENT CNC, CEC VALUES.
C
      CNCOR(1) = CNCOR(4) = CNC(NI)
      NI1 = NI + 1
      CNCOR(2) = CNCOR(3) = CNC(NI1)
      CECOR(1) = CECOR(2) = CEC(NN1)
      NN2 = NN1 + 1
      CECOR(3) = CECOR(4) = CEC(NN2)
C
C---  DETERMINE NUMBERS OF FIRST PRIOR AND LAST NEEDED GRID LINES,
C---  INDEPENDENT OF LINES TO BE SKIPPED.
C
      IF (I16 .EQ. 1) GO TO 13
```

```
      IN = IFIX((ESTART-CECOR(1))/SL)
      ILAST = IFIX((ESTART-CECOR(4))/SL) + 1
      GO TO 14
   13 IN = IFIX((CECOR(1)-ESTART)/SL)
      ILAST = IFIX((CECOR(4)-ESTART)/SL) + 1
C
C---  DETERMINE FIRST (NEAREST TO VIEWER) AND LAST NEEDED GRID LINE ARRAY
C---  SUBSCRIPTS, INDEPENDENT OF POINTS TO BE SKIPPED.
C
   14 IBOT  = IFIX((CNCOR(1)-NSTART)/PS) + 1
      ITOP  = IFIX((CNCOR(2)-NSTART)/PS) + 1
      IBOTTP = IBOT + ITOP
C
C---  CALCULATE SMALLEST AND LARGEST ARRAY SUBSCRIPTS, IMIN AND IMAX.
C
      IF (SOUTH) GO TO 158
      IMIN  = IBOT
      IMAX  = ITOP
      GO TO 159
  158 IMIN  = ITOP
      IMAX  = IBOT
C
C---  SKIP FORWARD THROUGH FIRST *IN* GRID LINE ARRAYS TO GET TO FIRST
C---  NEEDED ONE, CORRESPONDING TO DIAGRAM NEAR EDGE.
C
  159 IF (IN .EQ. 0) GO TO 44
      DO 112 I=1,IN
  112 READ (83)
C
C---  CALCULATE AND INITIALIZE VARIABLES FOR USE IN HIDDEN LINE TEST.
C---  INITIALIZE VARIABLES TO CORRECT VIEW-LINE DRIFT FACTOR.
C
   44 SLI9  = SL*I9
      PSI10 = PS*I10
      U = SLI9*CALPH2/CALPH1
      UDEL  = U/PSI10 - FLOAT(INT(U/PSI10))
      IF (UDEL .LE. 0.5) GO TO 45
      ISHIFT = I10*(INT(U/PSI10) + 1)
      IONE  = -I10
      UD = 1.0 - UDEL
      GO TO 46
   45 ISHIFT = I10*INT(U/PSI10)
      IONE  = I10
      UD = UDEL
   46 UTEST = 0.5
      UCUM  = 0.
      UU =  .FALSE.
C
C---  CALCULATE VARIABLES WHICH VARY ONLY BETWEEN BLOCKS.
C
      NM = NN1 - 1
      XAPEX = APX + NM*XHEW + ABS(CEC(NN1) - CEC(1))*CALPH2 /SCALE - (NI
     1 - 1)*XHNS - ABS(CNC(NI) - CNC(1))*CALPH1 /SCALE
      YAPEX = APY+ NM*YHEW + ABS(CEC(NN1) - CEC(1))*SALPH2 /SCALE + (NI
     1- 1)*YHNS + ABS(CNC(NI) - CNC(1))*SALPH1 /SCALE
C
C---  SET DO LOOP RANGE.
C
      IF (WEST) GO TO 156
      IN1 = IN + 1
```

```
      GO TO 157
  156 IN1 = ILAST
      ILAST = IN + 1
C
C---  OUTER DO LOOP - PROCESS ALL GRID LINES, ONE AT A TIME.
C
  157 DO 47 II=IN1,ILAST,I9
      READ (83) (PLOTZ(I),I=1,IMAX)
      IF (WEST) GO TO 160
      IF (I9 .EQ. 1) GO TO 162
      DO 17 I=1,I9M1
   17 READ (83)
      GO TO 162
  160 DO 16 I=1,I91
   16 BACKSPACE 83
  162 NCOUNT = I10I
C
C---  TEST FOR SERIOUS VIEW-LINE DRIFT, AND CORRECT IF NECESSARY.
C
      IF (UCUM .GT. UTEST) GO TO 70
      IF (.NOT. UU) GO TO 71
      ISHIFT = ISHIFT - IONE
      UU = .FALSE.
      GO TO 71
   70 UTEST = UTEST + 1.0
      UU = .TRUE.
      ISHIFT = ISHIFT + IONE
   71 UCUM = UCUM + UU
C
C---  CALCULATE VARIABLES AND SET LINE PLOT DIRECTION (FORWARD OR REVERSE).
C
      X2X = XAPEX + (II-IN1)*XLINEP
      Y2Y = YAPEX + (II-IN1)*YLINEP
      IF (.NOT. EVEN) GO TO 137
      EVEN = .FALSE.
      SPECS(14) = SPECS(15) = -FLOAT(I10)
      GO TO 138
  137 EVEN = .TRUE.
      SPECS(14) = SPECS(15) =  FLOAT(I10)
  138 IF (II .EQ. IN1 .OR. ISHIFT .EQ. 0) GO TO 64
      YPLOT(IMIN) = Y2Y + (PLOTZ(IBOT)  - VSTART)*ZFACTR/SCALE
      YYI = (YPLOT(IMIN) - YPLSVE(IBOT))/ISHIFT
C
C---  INNER DO LOOP - PROCESS ALL POINTS ON A GRID LINE, ONE POINT AT A TIME.
C---  TEST FOR HIDDEN POINTS AND CONSTRUCT THE CURRENT VISIBILITY (HORIZON)
C---  LINE.  DRAW VISIBLE LINE SEGMENTS AS NEEDED.
C
   64 DO 48 INDEX=IMIN,IMAX,I10
      IF (EVEN) III = INDEX
      IF (.NOT. EVEN) III = IBOTTP - INDEX
      IF (SOUTH) GO TO 176
      IREV = III
      ISH = IREV - ISHIFT
      GO TO 177
  176 IREV = IBOTTP - III
      ISH = IREV + ISHIFT
  177 XPLOT(III) = X2X - XPART*(III-IMIN)
      YPLOT(III) = Y2Y + YPART*(III-IMIN) + (PLOTZ(IREV) - VSTART)*ZFACT
     1R/SCALE
      IF (II .EQ. IN1) GO TO 50
```

```
      IF ((III-IMIN) .GE. ISHIFT) GO TO 49
      IF (YPLOT(III) .GE. (YPLOT(IMIN) - YYI*(III-IMIN)))GO TO 50
      PLOTZ(IREV) = YPLOT(IMIN) - YYI*(III-IMIN)
      GO TO 113
   50 PLOTZ(IREV) = YPLOT(III)
      NCOUNT = NCOUNT + I10
      GO TO 48
   49 IF (YPLOT(III) .GT. YPLSVE(ISH)) GO TO 50
      PLOTZ(IREV) = YPLSVE(ISH)
  113 IF (NCOUNT .EQ. I101) GO TO 48
      SPECS(13) = FLOAT((NCOUNT + I10M)/I10)
      IF (EVEN) I5 = III - NCOUNT - I10M
      IF (.NOT. EVEN) I5 = III + NCOUNT + I10M
      CALL SLLILI (XPLOT(I5),YPLOT(I5),SPECS)
      NCOUNT = I101
   48 CONTINUE
C
C---  DRAW LAST LINE SEGMENT IF NEEDED, AND LOAD NEW VISIBILITY LINE INTO
C---  YPLSVE.
C
      IF (NCOUNT .EQ. I101) GO TO 52
      SPECS(13) = FLOAT((NCOUNT + I10M)/I10)
      IF (EVEN) I5 = IMAX-NCOUNT+1
      IF (.NOT. EVEN) I5 = IMIN+NCOUNT-1
      CALL SLLILI (XPLOT(I5),YPLOT(I5),SPECS)
   52 DO 54 I=IMIN,IMAX,I10
   54 YPLSVE(I) = PLOTZ(I)
      I6 = II - IN1 + 1
C
C---  SAVE END-LINE POINTS
C
      BOTEND(I6) = PLOTZ(IBOT)
      TOPEND(I6) = PLOTZ(ITOP)
   47 CONTINUE
C
C---  ALL VISIBLE GRID LINE SEGMENTS HAVE BEEN DRAWN FOR A SINGLE BLOCK.
C---  NOW DRAW THE FAR-EDGE END LINE.
C
      SPECS(14) = SPECS(15) = FLOAT(I9)
      I5 = ILAST - IN1 + 1
      X3X = XAPEX - XPART*(IMAX-IMIN)
      DO 55 I=1,I5,I9
   55 PLOTZ(I) = X3X + (I-1)*XLINEP
      SPECS(13) = FLOAT((I5 + I9M1)/I9)
      CALL SLLILI (PLOTZ,TOPEND,SPECS)
      IF (IBCNT .EQ. 1) VDATAP = VSTART + (BOTEND(1)-YAPEX)*SCALE/ZFACTR
      IF (PLOTZ(1) .LT. XFRSVE)  XFRSVE = PLOTZ(1)
C
C---  DRAW GRID-LINE SIDE FRAME, IF WANTED.
C
      IF (FRAME .NE. 1.) GO TO 118
      XFRAME(1) = XFRAME(2) = PLOTZ(1)
      XFRAME(3) = XFRAME(4) = XAPEX
      YFRAME(1) = TOPEND(1)
      YFRAME(2) = YAPEX + YPART*(IMAX-IMIN)
      YFRAME(3) = YAPEX
      YFRAME(4) = BOTEND(1)
      SPECS(13) = 4.
      SPECS(14) = SPECS(15) = 1.
      CALL SLLILI(XFRAME,YFRAME,SPECS)
```

```
C
C---   DRAW TICKS IF NEEDED.
C
       IF (TICKS .NE. 1.) GO TO 118
       SPECS(13) = 2.
       CALL I3DTC1 (YFRAME(1),YFRAME(2),PLOTZ(1),CALPH1,0,-SALPH1,0,SPECS
      1)
       CALL I3DTC2 (CNCOR(2),CNCOR(1),SOUTH,XFRAME(2),YFRAME(2),CALPH1,SA
      1LPH1,1,SPECS)
       SPECS(13) = 3.
       CALL I3DTC1 (YFRAME(4),YFRAME(3),XAPEX,CALPH2,CALPH1,SALPH2,
      1SALPH1,SPECS)
C
C---   DRAW NEAR-EDGE END LINE.
C
  118 DO 56 I=1,I5,I9
   56 PLOTZ(I) = XAPEX + (I-1)*XLINEP
       SPECS(13) = FLOAT((I5 + I9M1)/I9)
       SPECS(14) = SPECS(15) = FLOAT(I9)
       CALL SLLILI (PLOTZ,BOTEND,SPECS)
       IF (PLOTZ(I5) .LT. XFRSVE) XFRSVE = PLOTZ(I5)
C
C---   DRAW TRANSVERSE-LINE SIDE FRAME, IF WANTED.
C
       IF (FRAME .NE. 1.) GO TO 12
       XFRAME(1) = XFRAME(2) = PLOTZ(I5)
       YFRAME(1) = BOTEND(I5)
       YFRAME(2) = YAPEX + YLINEP*(ILAST-IN1)
       SPECS(13) = 3.
       SPECS(14) = SPECS(15) = 1.
       CALL SLLILI(XFRAME,YFRAME,SPECS)
C
C---   DRAW TICKS IF NEEDED.
C
       IF (TICKS .NE. 1.) GO TO 12
       SPECS(13) = 2.
       CALL I3DTC2 (CECOR(1),CECOR(4),WEST,XAPEX,YAPEX, CALPH2,-SALPH2,2,
      1SPECS)
       CALL I3DTC1 (YFRAME(1),YFRAME(2),PLOTZ(I5),-CALPH2,0,-SALPH2,0,SPE
      1CS)
   12 CONTINUE
   11 CONTINUE
C
C---   ALL BLOCKS ARE COMPLETED.  PRINT STANDARD LEGEND IF WANTED.
C
       IF (LABEL .NE. 1.) GO TO 15
       CALL I3DLBL (SPECS)
   15 RETURN
       END
```

```
      SUBROUTINE I3DTC1 (YFA,YFB,XTC,CA,CB,SA,SB,SPECS)
      DIMENSION SPECS(30),XTIC(3),YTIC(3)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      EQUIVALENCE (RSPECS(4),VSTART),(RSPECS(17),MAJI),(RSPECS(18),MINI)
     1,(RSPECS(19),MAJL),(RSPECS(20),MINL),(RSPECS(10),SCALE),(RSPECS(30
     2),ZFACTR)
      REAL MINI,MAJI,MINL,MAJL
C
C---  I3DTC1 DRAWS TWO OR THREE POINT TICKS ON VERTICAL FRAME LINES.
C
      IS = IFIX(SPECS(13))
      IS1 = IS - 1
      IZTICS = INT((VSTART + (YFA-YFB)*SCALE/ZFACTR)/MINI) - INT(VSTART/
     1MINI)
      IF (IZTICS .EQ. 0) GO TO 1
      TICDIF = (VSTART/MINI - FLOAT(INT(VSTART/MINI)))*MINI*ZFACTR/SCALE
      TICREF = FLOAT(INT(VSTART/MINI))*MINI
      XTIC(IS1) = XTC
      YT = YFB - TICDIF
      SM = MINI*ZFACTR/SCALE
      DO 4 I=1,IZTICS
      YTIC(IS1) = YT + I*SM
      IF (FLOAT(INT((TICREF+MINI*I)/MAJI)) .EQ. (TICREF+MINI*I)/MAJI)
     1 GO TO 2
      TICVAR = MINL
      GO TO 3
    2 TICVAR = MAJL
    3 XTIC(IS) = XTIC(IS1) + TICVAR*CA
      YTIC(IS) = YTIC(IS1) + TICVAR*SA
      IF (IS .EQ. 2) GO TO 4
      XTIC(1) = XTIC(2)-TICVAR*CB
      YTIC(1) = YTIC(2) + TICVAR*SB
    4 CALL SLLILI (XTIC,YTIC,SPECS)
    1 RETURN
      END
```

```
      SUBROUTINE I3DTC2 (CORA,CORB,QUADRT,PTA,PTB,C,S,NCALL,SPECS)
      DIMENSION SPECS(30),XTIC(2),YTIC(2)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      EQUIVALENCE (RSPECS(10),SCALE),(RSPECS(17),MAJI),(RSPECS(18),MINI)
     1,(RSPECS(19),MAJL),(RSPECS(20),MINL),(ISPECS(6),I6)
      LOGICAL QUADRT
      REAL MINI,MAJI,MINL,MAJL
C
C---  I3DTC2 DRAWS TWO POINT TICKS ON HORIZONTAL FRAME LINES.
C
      TICSN = ABS(FLOAT(INT(CORA/MINI))-FLOAT(INT(CORB/MINI)))
      IF (TICSN .EQ. 0.) GO TO 1
      IF (NCALL .EQ. 1) GO TO 2
      IF (I6 .EQ. 1) GO TO 3
      IF (QUADRT) 4,5
    2 IF (.NOT. QUADRT) 5,4
    3 IF (QUADRT) 5,4
```

```
    4 TICDIF = (CORA/MINI - FLOAT(INT(CORA/MINI)))*MINI/SCALE
      TICREF = FLOAT(INT(CORA/MINI))*MINI
      SGNTIC = -1.
      GO TO 6
    5 TICDIF = (1. - CORA/MINI + FLOAT(INT(CORA/MINI)))*MINI/SCALE
      TICREF = (FLOAT(INT(CORA/MINI)) + 1.)*MINI
      SGNTIC = 1.
    6 ITICSN = IFIX(TICSN)
      SM =  SGNTIC*MINI
      PTAT  = PTA - TICDIF*C
      PTBT  = PTB + TICDIF*S
      CMS = C*MINI/SCALE
      SMS = S*MINI/SCALE
      DO 7  I=1,ITICSN
      XTIC(1) = PTAT + I*CMS
      XTIC(2) = XTIC(1)
      YTIC(1) = PTBT - I*SMS
      IF (FLOAT(INT((TICREF-SM*I)/MAJI)) .EQ. (TICREF-SM*I)/MAJI) GOTO 8
      TICVAR = MINL
      GO TO 9
    8 TICVAR = MAJL
    9 YTIC(2) = YTIC(1) + TICVAR
    7 CALL  SLLILI (XTIC,YTIC,SPECS)
    1 RETURN
      END




      SUBROUTINE I3DLBL (SPECS)
      DIMENSION SPECS(30)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      EQUIVALENCE (RSPECS(22),HITE),(RSPECS(23),XPOSL),(RSPECS(14),PSI),
     1(RSPECS(13),GAMMA),(RSPECS(25),XFRSVE),(RSPECS(7),PS),(RSPECS(10),
     2SCALE),(ISPECS(10),I10),(RSPECS(30),ZFACTR)
C
C---  I3DLBL LABELS THE DIAGRAM, GIVING DIRECTION OF SIGHT, INCLINATION
C---  (APPARENT ANGLE, TO THE HORIZONTAL, OF VIEW), AND POINT RESOLUTION.
C---  OTHER LETTERING IS LEFT TO THE USER.
C
      SPECS(17) = .2
      SPECS(18) = .2
      SPECS(19) = 0.
      SPECS(20) = 0.
      SPECS(21) =1.
      IF (XPOSL .EQ. -1.) GO TO 1
      SPECS(22) = XPOSL
      GO TO 2
    1 IF (XFRSVE .GE. 10.) SPECS(22) = XFRSVE - 9.0
      IF (XFRSVE .LT. 10.) SPECS(22) = 1.0
    2 SPECS(23) = HITE
      CALL  TITLEG  (18HDIRECTION OF SIGHT,SPECS)
      SPECS(22) = SPECS(22) + 4.2
      IF (PSI .GE. 0. .AND. PSI .LT. 90.) GO TO 166
      IF (PSI .GE. 90. .AND. PSI .LT. 180.) GO TO 167
      IF (PSI .GE. 180. .AND. PSI .LT. 270.) GO TO 168
      CALL TITLEG (6HN     W,SPECS)
      PSILBL = 360. - PSI
      GO TO 169
```

```
  166 CALL TITLEG (6HN     E,SPECS)
      PSILBL = PSI
      GO TO 169
  167 CALL TITLEG (6HS     E,SPECS)
      PSILBL = 180. - PSI
      GO TO 169
  168 CALL TITLEG (6HS     W,SPECS)
      PSILBL = PSI - 180.
  169 SPECS(28) = 1.
      SPECS(22) = SPECS(22) + .2
      CALL DECVAL (PSILBL,SPECS)
      SPECS(23) = SPECS(23) - .3
      SPECS(22) = SPECS(22) - 4.4
      CALL TITLEG (25HINCLINATION       DEGREES,SPECS)
      SPECS(22) = SPECS(22) + 2.6
      CALL DECVAL (GAMMA,SPECS)
      SPECS(22) = SPECS(22) - 2.6
      SPECS(23) = SPECS(23) - 2.4
      CALL TITLEG (24HVERTICAL EXAGGERATION = ,SPECS)
      SPECS(28) = 2.
      SPECS(22) = SPECS(22) + 4.8
      CALL DECVAL (ZFACTR,SPECS)
      SPECS(22) = SPECS(22) - 4.8
      SPECS(23) = SPECS(23) - .3
      SPECS(17) = SPECS(18) = .1
      RISP = SCALE/(PS*I10)
      SPECS(28) = 0.
      CALL TITLEG (28HRESOLUTION =      POINTS/INCH,SPECS)
      SPECS(22) = SPECS(22) + 1.3
      CALL DECVAL (RISP,SPECS)
      SPECS(22) = SPECS(22) - 1.3
      SPECS(23) = 2.0
      CALL TITLEG (24HPROGRAM BY W.B. WRAY JR.,SPECS)
      RETURN
      END
```

```
      SUBROUTINE I3DCVT (BUFI,LNGTHB,NUMPTC,FMT,EOR,DVCEOT,PNCH)
      DIMENSION BUFI(1),FMT(3)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      COMMON /TIME/ TO
      EQUIVALENCE (ISPECS(1),I1),(ISPECS(2),I2)
      DATA TLBL /6HI3DCVT/
C
C---  I3DCVT READS A HOLLERITH CARD DECK, POSSIBLY WITH EMBEDDED BLANKS,
C---  AND REDUCES THE DATA TO A SET OF BINARY LOGICAL RECORDS ON TAPE81,
C---  TAPE82, OR TAPE83 AS SPECIFIED BY USER.  OUTPUT IS ON TAPE, DISK,
C---  OR PUNCHED AS BINARY CARDS IF USER EQUIVALENCES PUNCHB.  THE
C---  ARRAY NUMPT IS FILLED, AND WILL BE PUNCHED AS THE LAST LOGICAL RECORD
C---  IF 6HPNCHOT IS USED IN THE CALLING ARGUMENT LIST.
C
      CALL SECOND (TO)
      L1 = 4*I1 + I2
      IF (LNGTHB .GE. L1) GO TO 1
      PRINT 2
```

```
    2 FORMAT (6X,*LNGTHB AS SET BY USER IS TOO SMALL FOR SUBROUTINE I3DC
     1VT*//)
      STOP
    1 N2 = 1 + I1
      N3 = N2 + 3*I1
      CALL EXCCVT (BUFI(1),BUFI(N2),BUFI(N3),LNGTHB,NUMPTC,FMT,EOR,DVCEO
     1T,PNCH)
      CALL TYME (TLBL)
      RETURN
      END



      SUBROUTINE EXCCVT (GRIDN,GRIDEL,NUMPT,LNGTHB,NUMPTC,FMT,EOR,DVCEOT
     1,PNCH)
      REAL GRIDN(1),GRIDEL(1),NUMPT(1),FMT(3),COORDN(40),COORDZ(40)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      DATA D1,D2,D3 /6HTAPE81,6HTAPE82,6HPNCHOT /
      EQUIVALENCE                    (ISPECS(2),I2), (ISPECS(3),I3)
C
C---  PREPARE FOR SELECTION OF OUTPUT TAPE UNIT.
C
      IF (DVCEOT .EQ. D1) GO TO 3
      IF (DVCEOT .EQ. D2) GO TO 4
      IDVCE = 83
      GO TO 5
    4 IDVCE = 82
      GO TO 5
    3 IDVCE = 81
C
C---  INITIALIZE COUNTERS AND LIMITS FOR DATA INPUT.
C
    5 IE = 0
      IEAST = 1
      I31 = I3 + 1
      L1 = NUMPTC
      L2 = L1 - 1
C
C---  BEGIN DATA INPUT AND REDUCTION CYCLE.  TESTS FOR END-OR RECORD
C---  FLAGS AND EMBEDDED BLANKS.
C
    6 ICOUNT = 0
    7 READ FMT, (COORDN(I),COORDZ(I),I=1,L1)
      IF (COORDN(L1) .NE. EOR) GO TO 9
      IF (L2 .EQ. 0) GO TO 11
      DO 8 I=1,L2
      IF (COORDN(I).EQ. 0.) GO TO 8
      ICOUNT = ICOUNT + 1
      GRIDN(ICOUNT) = COORDN(I)
      GRIDEL(ICOUNT) = COORDZ(I)
    8 CONTINUE
      GO TO 11
    9 DO 10 I=1,L1
      IF (COORDN(I) .EQ. 0.) GO TO 10
      ICOUNT = ICOUNT + 1
      GRIDN(ICOUNT) = COORDN(I)
      GRIDEL(ICOUNT) = COORDZ(I)
   10 CONTINUE
      GO TO 7
```

```
C
C---  AT THIS POINT ONE LOGICAL RECORD HAS BEEN REDUCED.  NOW INCREMENT
C---  RECORD COUNTERS AND PLACE ARRAYS ONTO CORRECT TAPE.
C
   11 IE = IE + 1
      WRITE (IDVCE) (GRIDN(I),GRIDEL(I),I=1,ICOUNT)
      NUMPT(IEAST) = FLOAT(ICOUNT)
      IEAST = IEAST + I31
C
C---  TEST FOR LAST COORDINATE ARRAY LOGICAL RECORD.
C
      IF (COORDZ(L1) .NE. EOR) GO TO 6
C
C---  TEST TO MAKE CERTAIN ENOUGH SPACE IS RESERVED FOR NUMPT ARRAY.
C
      IEAST = IEAST - I31
      IF (IEAST .EQ. I2) GO TO 22
      PRINT 21
   21 FORMAT (/10X,*ISPECS(2) INCORRECTLY SPECIFIED BY USER.  PROGRAM TE
     1RMINATED.*/)
      STOP
   22 PRINT 16,IE
   16 FORMAT (1H1,20X,*SUBROUTINE I3DCVT*/10X,*NUMBER OF COORDINATE ARRA
     1Y LOGICAL RECORDS PROCESSED = *,I4///)
C
C---  DETERMINE IF NUMPT ARRAY IS TO BE PUNCHED AS LAST LOGICAL RECORD.
C
      IF (PNCH .NE.D3) GO TO 20
      WRITE (IDVCE) (NUMPT(I),I=1,I2)
   20 RETURN
      END




      SUBROUTINE I3DPTA (BUFI,LNGTHB)
      DIMENSION BUFI(1)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      COMMON /TIME/ TO
      EQUIVALENCE (ISPECS(4),I4),(ISPECS(2),I2),(ISPECS(5),I5)
      DATA TLBL /5HI3DPTA/
C
C---  SUBROUTINE I3DPTA GENERATES A FULL SET OF ARRAY POINTS SPACED ACCORDING
C---  TO PS FROM A SKELETON ARRAY INPUT ON TAPE/DISK OR CARDS.
C
      CALL SECOND (TO)
      L1 = 2*I4 + I5 + I2
      IF (LNGTHB .GE. L1) GO TO 1
      PRINT 2
    2 FORMAT (6X,*LNGTHB AS SET BY USER IS TOO SMALL FOR SUBROUTINE I3DP
     1TA.*/)
      STOP
    1 N2 = 1 + I4
      N3 = N2 + I4
      N4 = N3 + I2
      CALL EXCPTA (BUFI(1),BUFI(N2),BUFI(N3),BUFI(N4),LNGTHB)
      CALL TYME (TLBL)
      RETURN
      END
```

```
      SUBROUTINE EXCPTA (GRIDN,GRIDEL,NUMPT,PLOTZ,LNGTHB)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      EQUIVALENCE (RSPECS(7),PS), (ISPECS(2),I2), (ISPECS(5),I5)
      REAL GRIDN(1),GRIDEL(1),PLOTZ(1),NUMPT(1)
C
C---  DO LOOP TO FILL OUT ALL I2 GRID LINES.
C
    1 DO 3 II=1,I2
      ICOUNT = IFIX(NUMPT(II))
      READ (82) (GRIDN(I),GRIDEL(I),I=1,ICOUNT)
C
C---  INITIALIZE VARIABLES TO HANDLE INITIAL POINTS ON GRID LINES, BEFORE
C---  BRANCH TO 300.
C
      PLOTN = GRIDN(1) - PS
      L1 = 2
      GELN = (GRIDEL(2)-GRIDEL(1))/(GRIDN(2)-GRIDN(1))
      GEL = GRIDEL(1)
      GN = GRIDN(1)
      DO 29 I=1,I5
      PLOTN = PLOTN + PS
      IF (PLOTN .LE. GRIDN(L1)) GO TO 30
C
C---  RESET L AND L1 SO THAT NEXT CORRECT PAIR OF SKELETON GRID POINTS IS
C---  USED TO CALCULATE PLOTZ POINTS.
C
  300 L1 = L1 + 1
      IF (PLOTN .GT. GRIDN(L1)) GO TO 300
      L = L1 - 1
      GELN = (GRIDEL(L1)-GRIDEL(L))/(GRIDN(L1)-GRIDN(L))
      GEL = GRIDEL(L)
      GN = GRIDN(L)
   30 PLOTZ(I) = (PLOTN-GN)*GELN + GEL
   29 CONTINUE
    3 WRITE (83) (PLOTZ(I),I=1,I5)
      RETURN
      END
```

```
      SUBROUTINE I3DPTS (BUFI,LNGTHB,PNCH,FMT,NUMPTC)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      COMMON /TIME/ TO
      DIMENSION BUFI(1),FMT(3)
      EQUIVALENCE (ISPECS(1),I1),(ISPECS(2),I2),(ISPECS(3),I3),(ISPECS(4
     1),I4)
      DATA TLBL /6HI3DPTS/
C
C---  SUBROUTINE I3DPTS CALCULATES INTERMEDIATE LINES IN ANY NUMBER FROM
C---  A BASIC ARRAY OF POINTS IN LINES.  THE BASIC ARRAY INPUT IS FROM
C---  TAPE81, EITHER MAGNETIC TAPE, DISK, OR BINARY CARDS IF EQUIVALENCED TO
C---  INPUT.  OUTPUT IS EITHER IN BINARY OR HOLLERITH MODE.  IF TAPE82
C---  IS EQUIVALENCED TO PUNCHB, A BINARY DECK IS PUNCHED AND THE ARRAY
C---  NUMPT IS PUNCHED AS THE LAST LOGICAL RECORD.  IF A HOLLERITH FORMAT IS
C---  SPECIFIED IN THE CALL LIST, HOLLERITH DATA CARDS ARE PUNCHED.  PUNCH
C---  MUST BE DECLARED.  IF BOTH 6HPNCHOT AND A HOLLERITH FORMAT ARE
C---  DECLARED, THEN BOTH HOLLERITH DATA CARDS AND BINARY NUMPT ARRAY
C---  ARE PUNCHED.
```

```
C
      CALL SECOND (TJ)
      L1 = 4*I1 + I2 + 2*I3*I4
      IF (LNGTHB .GE. L1) GO TO 1
      PRINT 2
    2 FORMAT (6X,*LNGTHB AS SET BY USER IS TOO SMALL FOR SUBROUTINE I3DP
     1TS.*/)
      STOP
    1 N2 = 1 + I1
      N3 = N2 + I1
      N4 = N3 + I1
      N5 = N4 + I1
      N6 = N5 + I2
      N7 = N6 + I3*I4
      CALL EXCPTS (BUFI(1),BUFI(N2),BUFI(N3),BUFI(N4),BUFI(N5),BUFI(N6),
     1BUFI(N7),LNGTHB,I3,I4,PNCH,FMT,NUMPTC)
      CALL TYME (TLBL)
      RETURN
      END




      SUBROUTINE EXCPTS (GRIDN,GRIDEL,GRIDN1,GRIDL1,NUMPT,XGRIDN,XGRIDZ,
     1LNGTHB,I3,I4,PNCH,FMT,NUMPTC)
      REAL GRIDN(1),GRIDEL(1),GRIDN1(1),GRIDL1(1),NUMPT(1),XGRIDN(I3,I4)
     1,XGRIDZ(I3,I4)
      INTEGER PTPLT
      DIMENSION FMT(3)
      COMMON /WSPECS/ RSPECS(30),ISPECS(10)
      COMMON /I3DPS/ INTCT,FI3

      EQUIVALENCE (RSPECS(1),SPACEL), (RSPECS(6),DELFAC), (ISPECS(1),
     1I1), (ISPECS(2),I2)
      DATA D1 /6HPNCHOT/
      DATA D2 /6HNOHOLP/
      I32 = I3 + 2
      I31 = I3 + 1
      ICOUNT = IFIX(NUMPT(1))
      READ (81) (GRIDN(I),GRIDEL(I),I=1,ICOUNT)
      NUM = 0
      FI3 = 1./(I3 + 1)
      DELTAD = DELFAC*SPACEL*(I3+1)
C
C---  BEGIN MAIN DO LOOP TO CALCULATE COORDINATES FOR SKELETON POINTS ON
C---  ALL INTERMEDIATE GRID LINES.  EACH PASS THROUGH THE LOOP CONSIDERS
C---  ONE PAIR OF BASIC GRID LINES AND THE INTERMEDIATE GRID LINES
C---  BETWEEN.
C
      DO 9 II=I32,I2,I31
      ICNT1 = IFIX(NUMPT(II))
      READ (81) (GRIDN1(I),GRIDL1(I),I=1,ICNT1)
C
C---  INITIALIZE VALUES FOR ARRAY SUBSCRIPTS.
C
      J = 1
      JL = ICOUNT
      K1 = 1
      K2 = 2
```

```
      KL = ICNT1
      INTCT = 0
      PTPLT = 0
C
C---  BEGIN TESTS OF POINTS ON THE TWO BASIC LINES.  RELATIVE ELEVATIONS
C---  AND DISTANCES APART DETERMINE WHICH POINT PAIRS ARE USED IN THE
C---  CALCULATION OF INTERMEDIATE LINE SKELETON POINTS.
C
   13 IF (GRIDEL(J) .NE. GRIDL1(K1) .OR. GRIDEL(J) .NE. GRIDL1(K2))
     1GO TO 10
   16 IF (ABS(GRIDN(J) - GRIDN1(K1)) .GT. DELTAD .AND.ABS(GRIDN(J) - GRI
     1DN1(K2)) .GT. DELTAD) GO TO 107
      GO TO 24
  107 IF (GRIDN(J) .LT. GRIDN1(K2)) GO TO 11
      K1 = K2
      K2 = K2 + 1
      GO TO 13
C
C---  POINTS K1 AND K2 ON LINE B ARE AT THE SAME ELEVATION AS POINT J ON
C---  LINE A.  CALL SUBROUTINE INTLNE TWICE TO CALCULATE TWO SETS OF
C---  POINTS ON INTERMEDIATE LINES.  TEST 2I) IS SUCCESSFUL.
C
   24 CALL INTLNE (K1,J,I3,I4,XGRIDN,XGRIDZ,GRIDN,GRIDEL,GRIDN1,GRIDL1)
  108 CALL INTLNE (K2,J,I3,I4,XGRIDN,XGRIDZ,GRIDN,GRIDEL,GRIDN1,GRIDL1)
  110 J = J + 1
C
C---  SEVERAL TESTS TO SEE IF J BEYOND LAST POINT ON THE GRID LINE (J
C---  .GT. JL).    BRANCHING DEPENDENT ON VALUE OF K2.
C
      IF (J .GT. JL .AND. K2 .EQ. KL) GO TO 12
      IF (J .GT. JL .AND. K2 .LT. KL) GO TO 109
      K1 = K2
      K2 = K2 + 1
      IF (K2 .LE. KL) GO TO 13
C
C---  TRY TO MATCH UP ALL REMAINING *J* POINTS ON LINE A WITH THE LAST
C---  (NOW K1) B-LINE POINT.
C
      DO 14 I=J,JL
      IF (ABS(GRIDN(I) - GRIDN1(K1)) .GT. DELTAD) GO TO 14
      CALL INTLNE (K1,I,I3,I4,XGRIDN,XGRIDZ,GRIDN,GRIDEL,GRIDN1,GRIDL1)
   14 CONTINUE
      GO TO 12
C
C---  MATCH UP, ONE AT A TIME, THE LAST (J = JL) POINT ON LINE A WITH
C---  THE REMAINING POINTS ON LINE B.  CYCLE THROUGH STATEMENT 108.
C
  109 J = J - 1
      K2 = K2 + 1
      IF (ABS(GRIDN(J) - GRIDN1(K2)) .LE. DELTAD) GO TO 108
      GO TO 110
C
C---  TEST FOR SITUATION 2II)
C
   10 IF (GRIDEL(J) .NE. GRIDL1(K1) .AND. GRIDEL(J) .EQ. GRIDL1(K2))
     1GO TO 15
C
C---  TEST FOR SITUATION 2III)
C
      IF (GRIDEL(J) .EQ. GRIDL1(K1) .AND. GRIDEL(J) .NE. GRIDL1(K2))
     1GO TO 16
```

```
C
C---    SITUATION 2IV) BY DEFAULT.  TEST FOR TWO CONSECUTIVE EQUAL
C---    ELEVATION POINTS SATISFYING DISTANCE REQUIREMENTS.
C
        K3 = K2 + 1
        IF (K3 .GT. KL) GO TO 17
        DIST1 = ABS(GRIDN(J) - GRIDN1(K2))
        DO 18 I=K3,KL
        DIST2 = ABS(GRIDN(J)  - GRIDN1(I))
        IF (DIST2 .GT. DELTAD .AND. DIST2 .GT. DIST1) GO TO 17
        I1 = I + 1
        IF (I1 .GT. KL) GO TO 18
        DISTI1 = ABS(GRIDN(J) - GRIDN1(I1))
        IF ((GRIDEL(J) - GRIDL1(I)) .EQ. 0. .AND. (GRIDEL(J) - GRIDL1(I1))
     1 .EQ. 0. .AND. (DIST2 .LE. DELTAD .OR. DISTI1 .LE. DELTAD)) GO TO
     2 19
        DIST1 = DIST2
   18 CONTINUE
        GO TO 17
   15 K3 = K2 + 1
        IF (K3 .GT. KL) GO TO 16
        IF (GRIDEL(J) .NE. GRIDL1(K3)) GO TO 16
        IF (J .EQ. 1) GO TO 20
   21 K1 = K2
        K2 = K3
        GO TO 16
C
C---    ROUTINE TO MAKE CERTAIN THAT THERE WILL DEFINITELY BE A POINT ON
C---    THE INT. GRID LINES EQUAL TO NSTART.  THUS, J = 1 POINT WILL BE
C---    USED TO CALCULATE 3 INSTEAD OF 2 SETS OF INT. GRID POINTS.
C
   20 CALL INTLNE (K1,J,I3,I4,XGRIDN,XGRID2,GRIDN,GRIDEL,GRIDN1,GRIDL1)
        GO TO 21
   19 IF (J .EQ. 1) GO TO 22
   23 K1 = I
        K2 = I1
        GO TO 24
C
C---    ROUTINE TO MAKE CERTAIN THAT THERE WILL DEFINITELY BE A POINT ON
C---    THE INT. GRID LINES EQUAL TO NSTART.
C
   22 CALL INTLNE (K1,J,I3,I4,XGRIDN,XGRID2,GRIDN,GRIDEL,GRIDN1,GRIDL1)
        GO TO 23
C
C---    SITUATION 2IVB) FOR TWO B-LINE POINTS AT UNEQUAL ELEVATIONS - THE
C---    GENERAL, HILLSLOPE CASE.  NOTE - FOR EACH ≠A≠ POINT THERE DOES NOT
C---    HAVE TO BE ANY ≠B≠ POINTS ≠USED UP≠.  IF ONE OR NO INT. LINE SETS
C---    CALCULATED, K1 REMAINS EQUAL TO K1.  PTPLT USED AS COUNTER.
C
   17 KK = K1
        PTPLT = 0
        DIST1 = ABS(GRIDN(J)  - GRIDN1(KK))
        DO 25 I=K2,KL
        DIST2 = ABS(GRIDN(J)  - GRIDN1(I))
        IF (DIST1 .LE. DELTAD) GO TO 26
   27 IF (DIST2 .GT. DELTAD .AND. DIST2 .GT. DIST1) GO TO 11
        DIST1 = DIST2
   25 KK = KK + 1
        GO TO 11
C
```

```
C---  K1 SET TO KK SO THAT K1 WILL BE THE INDEX OF THE LAST USED B POINT
C
   26 K1 = KK
      K2 = KK + 1
      PTPLT = PTPLT + 1
      CALL INTLNE (KK,J,I3,I4,XGRIDN,XGRIDZ,GRIDN,GRIDEL,GRIDN1,GRIDL1)
      IF (PTPLT .EQ. 2) GO TO 11
      GO TO 27
C
C---  NOTE THAT AT STATEMENT 11, K2 CANNOT EQUAL KK+1 WHEN KK = K2,
C---  SINCE AT COMPLETION OF DO LOOP I=K2,KL, PROGRAM GOES DIRECTLY TO
C---  STATEMENT 11.  THUS, K2 CAN NEVER BE .GT. K.
C
   11 J = J + 1
      IF (J .LE. JL) GO TO 13
C
C---  THIS STEP BELOW ALWAYS OK.  SEE NOTE JUST ABOVE.
C
      IF (K2 .EQ. KL) GO TO 106
      J = J - 1
      IF (PTPLT .EQ. 2) GO TO 111
      K1 = K2
      K2 = K2 + 1
      GO TO 13
  106 IF (KK .LT. KL) GO TO 12
      IF (DIST1 .GT. DELTAD) GO TO 12
      J = J - 1
  111 CALL INTLNE (KL,J,I3,I4,XGRIDN,XGRIDZ,GRIDN,GRIDEL,GRIDN1,GRIDL1)
C
C---  AT THIS POINT, ALL INTERMEDIATE LINES BETWEEN A AND B ARE
C---  COMPLETE.  NOW, PLACE ARRAYS INTO UNIT 82 OR PUNCH.
C
   12 IF (FMT(1) .NE. D2) GO TO 200
      WRITE (82)(GRIDN(I),GRIDEL(I),I=1,ICOUNT)
      GO TO 201
  200 INDEX = ICOUNT/NUMPTC
      INDEXA = ICOUNT - INDEX*NUMPTC
      DO 206 J=1,INDEX
  206 PUNCH FMT, (GRIDN(I),GRIDEL(I),I=1,NUMPTC)
      IF (INDEXA .GT. 0) PUNCH FMT, (GRIDN(I),GRIDEL(I),I=1,INDEXA)
      PUNCH 800
  800 FORMAT (80X)
  201 NUM = NUM + 1
      DO 51 L=1,I3
      IF (FMT(1) .NE. D2) GO TO 202
      WRITE (82) (XGRIDN(L,I),XGRIDZ(L,I),I=1,INTCT)
      GO TO 203
  202 INDEX = INTCT/NUMPTC
      INDEXA = INTCT - INDEX*NUMPTC
      DO 207 J=1,INDEX
  207 PUNCH FMT, (XGRIDN(L,I),XGRIDZ(L,I),I=1,NUMPTC)
      IF (INDEXA .GT. 0) PUNCH FMT, (XGRIDN(L,I),XGRIDZ(L,I),I=1,NUMPTC)
      PUNCH 800
  203 NUM = NUM + 1
   51 NUMPT(NUM) = FLOAT(INTCT)
      IF (II .LT.I2) GO TO 53
      IF (FMT(1) .NE. D2) GO TO 204
      WRITE (82) (GRIDN1(I),GRIDL1(I),I=1,ICNT1)
      GO TO 205
  204 INDEX = ICNT1/NUMPTC
```

```
      INDEXA = ICNT1 - INDEX*NUMPTC
      DO 208 J=1,INDEX
  208 PUNCH FMT, (GRIDN1(I),GRIDL1(I),I=1,NUMPTC)
      IF (INDEXA .GT. 0) PUNCH FMT, (GRIDN1(I),GRIDL1(I),I=1,INDEXA)
      PUNCH 800
  205 GO TO 9
C
C---  LOAD LINE B POINTS INTO LINE A, PRIOR TO NEXT PASS THROUGH LOOP.
C
   53 DO 52 I=1,ICNT1
      GRIDN(I) = GRIDN1(I)
   52 GRIDEL(I) = GRIDL1(I)
      ICOUNT = ICNT1
    9 CONTINUE
C
C---  DETERMINE IF NUMPT ARRAY IS TO BE PUNCHED AS LAST LOGICAL RECORD.
C
      IF (PNCH .NE. 01) GO TO 40
      WRITE (82) (NUMPT(I),I=1,I2)
   40 RETURN
      END
```

```
      SUBROUTINE INTLNE (K,J,I3,I4,XN,XZ,GN,GEL,GN1,GL1)
      COMMON /I3DPS/ INTCT,FI3
      REAL XN(I3,I4),XZ(I3,I4),GN(1),GEL(1),GN1(1),GL1(1)
C
C---  ROUTINE TO CALCULATE ONE POINT FOR EACH OF I3 INT. GRID LINES.
C
      INTCT = INTCT + 1
      F = FI3
      DO 1 I=1,I3
      XN(I,INTCT) = (GN1(K) - GN(J))*F + GN(J)
      XZ(I,INTCT) = (GL1(K) - GEL(J))*F + GEL(J)
    1 F = F + FI3
      RETURN
      END
```

Example 3

```
PROGRAM MAIN (OUTPUT,TAPE83,TAPE99)
DIMENSION SPECS(30),Z(283)
COMMON BUFI(1308)
COMMON /WSPECS/RSPECS(30),ISPECS(10)
SPECS(1) = 1.
SPECS(2) = 0.5
SPECS(3) = 46.
SPECS(4) = 1.
SPECS(5) = 10.0
SPECS(6) = 0.5
SPECS(7) = 45.0
SPECS(8) = 9.5
SPECS(11) = 1.
SPECS(12) = 99.
ISPECS(2) = 80
ISPECS(5) = 283
ISPECS(6) = 1
ISPECS(7) = 1
ISPECS(8) = 3
ISPECS(9) = 1
ISPECS(10)= 1
RSPECS(1)  = .25
RSPECS(2) = 0.
RSPECS(3)  = 0.
RSPECS(4) = 0.
RSPECS(5) = 12.5
RSPECS(7)  = .07
RSPECS(10) = 5.
RSPECS(11) = 28.
RSPECS(12) = 2.
RSPECS(13) = 23.
RSPECS(14) = 44.
RSPECS(15) = 0.
RSPECS(16) = 0.
RSPECS(21) = 1.
RSPECS(22) = 7.
RSPECS(23) = -1.
RSPECS(28) = .5
RSPECS(29) = .5
RSPECS(30) = 1.
BUFI(1133) = BUFI(1134) = BUFI(1135) = 1.
BUFI(1139) = 0.
BUFI(1140) = 19.75
BUFI(1141) = 0.
BUFI(1142) = 5.11
BUFI(1143) = 14.84
BUFI(1144) = 19.74
REWIND 83
REAST = 0.
DO 1 I=1,80
RNORTH = 0.
DO 2 J=1,283
Z(J) = 1.5/SQRT((REAST - 5.375)**2 + (RNORTH -14.875)**2)
1+       1.5/SQRT((REAST - 7.125)**2 + (RNORTH -14.875)**2)
2+       1.5/SQRT((REAST -13.125)**2 + (RNORTH -14.875)**2)
```

```
 3+      1.5/SQRT((REAST -14.375)**2 + (RNORTH -14.875)**2)
 4+      1.5/SQRT((REAST - 9.125)**2 + (RNORTH - 5.125)**2)
 5+      1.5/SQRT((REAST -10.375)**2 + (RNORTH - 5.125)**2)
  RNORTH = RNORTH + .07
2 CONTINUE
  WRITE (83) (Z(N),N=1,283)
  REAST = REAST + .25
1 CONTINUE
  REWIND 83
  CALL I3D1GL (BUF1,1308,SPECS)
  CALL GDSEND(SPECS)
  STOP
  END
```

Listings of α_1 and α_2 as a function of ψ and γ , in five degree intervals for one quadrant.

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	PSI 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
55																		
54																		
53																		
52																		
51																	88.3	85.3
																	83.9	74.5
50														86.9	84.8	83.2	81.9	80.7
														76.6	68.7	64.0	60.6	58.0
49											87.2	85.2	83.7	82.4	81.2	80.1	79.1	78.0
											73.8	64.5	58.9	55.1	52.3	50.2	48.5	47.3
48								88.4	86.2	84.7	83.4	82.3	81.2	80.1	79.1	78.1	77.1	76.2
								76.4	61.7	54.4	49.6	46.3	43.8	42.0	40.7	39.7	39.0	38.5
47						87.7	86.1	84.9	83.7	82.7	81.6	80.6	79.6	78.6	77.7	76.7	75.7	74.8
						62.1	50.0	43.4	39.2	36.4	34.4	33.0	32.0	31.4	31.0	30.8	30.7	30.8
46				87.9	86.7	85.6	84.6	83.6	82.5	81.6	80.6	79.6	78.6	77.6	76.7	75.7	74.7	73.7
				45.9	34.6	29.0	25.8	23.8	22.7	22.0	21.6	21.5	21.6	21.9	22.2	22.6	23.1	23.7
45	90.0	89.0	88.0	87.0	86.0	85.0	84.0	83.0	82.0	81.0	80.0	79.0	78.0	77.0	76.0	75.0	74.0	73.0
	-.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	11.0	12.0	13.0	14.0	15.0	16.0	17.0
44			89.5	87.6	86.4	85.2	84.1	83.0	82.0	80.9	79.8	78.8	77.7	76.7	75.6	74.6	73.6	72.5
			-74.7	-37.8	-25.4	-18.1	-13.1	-9.2	-6.1	-3.5	-1.2	.8	2.7	4.5	6.1	7.6	9.1	10.6
43					88.5	86.5	85.0	83.7	82.4	81.3	80.1	78.9	77.8	76.7	75.6	74.5	73.4	72.2
					-68.1	-45.1	-33.8	-26.3	-20.7	-16.2	-12.5	-9.3	-6.5	-4.0	-1.7	.4	2.4	4.3
42							87.2	85.2	83.5	82.1	80.8	79.5	78.2	77.0	75.8	74.6	73.4	72.2
							-62.8	-46.6	-36.9	-29.9	-24.4	-19.8	-15.9	-12.6	-9.5	-6.8	-4.3	-2.0

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
41									85.8 -58.2	83.7 -45.9	82.0 -37.6	80.4 -31.1	79.0 -25.9	77.6 -21.5	76.3 -17.6	74.9 -14.2	73.7 -11.1	72.4 -8.3
40										86.9 -70.2	84.0 -53.9	82.0 -44.2	80.2 -36.9	78.6 -31.0	77.1 -26.1	75.6 -21.9	74.2 -18.1	72.8 -14.8
39												84.7 -61.4	82.1 -50.0	80.1 -41.8	78.3 -35.4	76.6 -30.1	75.0 -25.5	73.5 -21.5
38													85.6 -68.9	82.4 -55.1	80.1 -46.1	78.0 -39.2	76.2 -33.5	74.4 -28.6
37														87.1 -77.5	82.8 -59.8	80.1 -49.9	77.8 -42.4	75.7 -36.4
36																83.3 -64.0	80.1 -53.1	77.5 -45.1
35																	83.8 -67.9	80.1 -55.8
34																		84.4 -71.5
33																		
32																		
31																		
30																		
29																		
28																		

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
55																		
54															87.8	85.4	83.9	82.7
															86.5	82.9	80.9	79.5
53								87.9	85.1	83.3	81.9	80.7	79.6	78.7	77.7	76.9	76.1	75.3
								85.5	79.9	76.8	74.7	73.1	71.9	70.9	70.1	69.5	69.1	68.8
52			86.2	85.2	83.4	81.9	80.6	79.5	78.4	77.4	76.5	75.6	74.7	73.9	73.1	72.3	71.5	70.8
			85.0	77.3	73.4	70.6	68.6	66.9	65.7	64.7	63.9	63.3	62.8	62.4	62.2	62.0	62.0	62.0
51	83.5	82.1	80.8	79.6	78.5	77.5	76.5	75.6	74.6	73.8	72.9	72.0	71.2	70.4	69.6	68.8	68.1	67.3
	69.7	66.4	64.0	62.0	60.5	59.4	58.4	57.7	57.1	56.7	56.4	56.2	56.1	56.1	56.1	56.2	56.4	56.6
50	79.6	78.5	77.5	76.5	75.5	74.6	73.7	72.8	71.9	71.0	70.1	69.3	68.5	67.7	66.9	66.1	65.3	64.5
	56.1	54.6	53.4	52.4	51.7	51.2	50.8	50.5	50.4	50.3	50.3	50.4	50.6	50.8	51.0	51.3	51.7	52.1
49	77.0	76.1	75.1	74.2	73.2	72.3	71.4	70.5	69.7	68.8	67.9	67.1	66.2	65.4	64.5	63.7	62.9	62.1
	46.3	45.6	45.1	44.7	44.5	44.4	44.4	44.4	44.6	44.8	45.0	45.3	45.7	46.1	46.5	47.0	47.5	48.0
48	75.2	74.3	73.3	72.4	71.5	70.6	69.7	68.8	67.9	67.0	66.1	65.2	64.3	63.4	62.6	61.7	60.8	60.0
	38.2	38.0	38.0	38.0	38.2	38.4	38.7	39.0	39.4	39.8	40.3	40.8	41.3	41.9	42.5	43.1	43.7	44.4
47	73.8	72.9	71.9	71.0	70.1	69.1	68.2	67.3	66.3	65.4	64.5	63.6	62.7	61.8	60.8	59.9	59.0	58.1
	31.0	31.2	31.6	32.0	32.4	32.9	33.5	34.0	34.6	35.3	35.9	36.6	37.3	38.0	38.8	39.5	40.3	41.1
46	72.8	71.8	70.8	69.9	68.9	68.0	67.0	66.0	65.1	64.1	63.2	62.2	61.2	60.3	59.3	58.4	57.4	56.5
	24.3	24.9	25.6	26.3	27.1	27.8	28.6	29.4	30.2	31.0	31.9	32.7	33.6	34.4	35.3	36.2	37.0	37.9
45	72.0	71.0	70.0	69.0	68.0	67.0	66.0	65.0	64.0	63.0	62.0	61.0	60.0	59.0	58.0	57.0	56.0	55.0
	18.0	19.0	20.0	21.0	22.0	23.0	24.0	25.0	26.0	27.0	28.0	29.0	30.0	31.0	32.0	33.0	34.0	35.0
44	71.5	70.4	69.4	68.3	67.3	66.2	65.2	64.2	63.1	62.1	61.0	60.0	58.9	57.9	56.8	55.8	54.7	53.7
	11.9	13.3	14.6	15.9	17.1	18.4	19.6	20.8	22.0	23.2	24.3	25.5	26.6	27.7	28.9	30.0	31.1	32.2
43	71.1	70.0	69.0	67.9	66.8	65.7	64.6	63.5	62.4	61.3	60.2	59.1	58.0	56.9	55.8	54.7	53.6	52.5
	6.0	7.7	9.3	10.9	12.4	13.9	15.3	16.7	18.1	19.4	20.8	22.1	23.3	24.6	25.9	27.1	28.3	29.5
42	71.0	69.9	68.7	67.6	66.4	65.2	64.1	62.9	61.8	60.6	59.5	58.3	57.2	56.0	54.9	53.7	52.6	51.4
	.2	2.2	4.2	6.0	7.8	9.5	11.1	12.7	14.3	15.8	17.3	18.8	20.2	21.6	23.0	24.3	25.7	27.0

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
41	71.1	69.9	68.7	67.4	66.2	65.0	63.8	62.6	61.4	60.1	58.9	57.7	56.5	55.3	54.1	52.9	51.6	50.4
	-5.7	-3.3	-1.0	1.2	3.2	5.2	7.0	8.8	10.6	12.3	13.9	15.5	17.1	18.6	20.1	21.6	23.1	24.5
40	71.4	70.1	68.8	67.5	66.2	64.9	63.6	62.3	61.0	59.8	58.5	57.2	55.9	54.7	53.4	52.1	50.8	49.5
	-11.7	-8.8	-6.2	-3.7	-1.4	.8	2.9	5.0	6.9	8.8	10.6	12.4	14.1	15.8	17.4	19.0	20.6	22.1
39	72.0	70.5	69.1	67.7	66.3	64.9	63.6	62.2	60.9	59.5	58.2	56.8	55.5	54.1	52.8	51.5	50.1	48.8
	-17.8	-14.5	-11.5	-8.7	-6.0	-3.5	-1.2	1.1	3.2	5.3	7.3	9.2	11.1	12.9	14.7	16.4	18.1	19.7
38	72.8	71.2	69.6	68.1	66.6	65.1	63.7	62.2	60.8	59.4	58.0	56.6	55.1	53.7	52.3	50.9	49.5	48.1
	-24.3	-20.4	-16.9	-13.7	-10.7	-7.9	-5.3	-2.8	-.4	1.8	4.0	6.1	8.1	10.1	12.0	13.8	15.7	17.4
37	73.9	72.1	70.4	68.7	67.1	65.5	64.0	62.4	60.9	59.4	57.9	56.4	54.9	53.4	51.9	50.4	49.0	47.5
	-31.2	-26.7	-22.6	-18.9	-15.6	-12.4	-9.5	-6.8	-4.2	-1.7	.7	3.0	5.2	7.3	9.3	11.3	13.3	15.1
36	75.3	73.3	71.4	69.5	67.8	66.1	64.4	62.7	61.1	59.5	57.9	56.3	54.8	53.2	51.6	50.1	48.5	47.0
	-38.8	-33.4	-28.6	-24.4	-20.6	-17.1	-13.8	-10.8	-7.9	-5.2	-2.6	-.2	2.2	4.5	6.7	8.8	10.9	12.9
35	77.3	74.9	72.7	70.6	68.7	66.8	65.0	63.2	61.5	59.8	58.1	56.4	54.7	53.1	51.4	49.8	48.2	46.5
	-47.4	-40.8	-35.2	-30.3	-25.9	-21.9	-18.3	-14.9	-11.8	-8.8	-6.0	-3.3	-.8	1.7	4.0	6.3	8.5	10.6
34	80.1	77.0	74.4	72.0	69.9	67.8	65.8	63.9	62.0	60.2	58.4	56.6	54.8	53.1	51.3	49.6	47.9	46.2
	-58.1	-49.3	-42.4	-35.6	-31.5	-27.0	-23.0	-19.2	-15.8	-12.5	-9.5	-6.6	-3.8	-1.2	1.3	3.8	6.1	8.4
33	85.1	80.1	76.7	73.9	71.4	69.0	66.8	64.7	62.7	60.7	58.8	56.9	55.0	53.2	51.3	49.5	47.7	45.9
	-74.8	-60.0	-50.8	-43.7	-37.7	-32.5	-27.9	-23.7	-19.9	-16.4	-13.0	-9.9	-6.9	-4.1	-1.4	1.2	3.7	6.1
32		85.8	80.0	76.3	73.3	70.6	68.1	65.8	63.6	61.5	59.4	57.4	55.4	53.4	51.4	49.5	47.6	45.7
		-77.7	-61.5	-52.0	-44.6	-38.5	-33.2	-28.5	-24.3	-20.4	-16.7	-13.3	-10.1	-7.1	-4.1	-1.4	1.3	3.9
31			86.4	79.8	75.8	72.6	69.8	67.2	64.7	62.4	60.1	58.0	55.8	53.7	51.7	49.6	47.6	45.6
			-80.2	-62.5	-52.7	-45.3	-39.1	-33.7	-28.9	-24.6	-20.6	-16.9	-13.4	-10.1	-7.0	-4.0	-1.2	1.6
30				86.9	79.5	75.3	71.9	68.9	66.1	63.5	61.1	58.7	56.4	54.2	52.0	49.8	47.7	45.6
				-82.0	-63.2	-53.2	-45.6	-39.4	-33.9	-29.0	-24.6	-20.6	-16.8	-13.2	-9.9	-6.7	-3.7	-.8
29					87.1	79.1	74.6	71.0	67.9	65.0	62.3	59.7	57.2	54.8	52.5	50.2	47.9	45.6
					-82.9	-63.4	-53.4	-45.7	-39.4	-33.9	-29.0	-24.5	-20.3	-16.5	-12.9	-9.5	-6.2	-3.1
28						86.8	78.5	73.8	70.0	66.7	63.7	60.9	58.2	55.6	53.1	50.6	48.2	45.6
						-82.5	-63.3	-53.2	-45.6	-39.2	-33.6	-28.7	-24.1	-19.9	-16.0	-12.4	-8.9	-5.6

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
27							86.1	77.7	72.9	69.0	65.5	62.4	59.4	56.6	53.9	51.2	48.7	46.1
							-81.1	-62.8	-52.8	-45.2	-38.8	-33.2	-28.1	-23.6	-19.3	-15.4	-11.6	-8.1
26								85.0	76.8	71.8	67.7	64.2	60.9	57.8	54.8	52.0	49.2	46.5
								-79.2	-62.1	-52.2	-44.6	-38.1	-32.5	-27.4	-22.8	-18.5	-14.5	-10.7
25									83.7	75.7	70.6	66.4	62.7	59.2	56.0	52.9	50.0	47.1
									-76.9	-61.0	-51.3	-43.7	-37.3	-31.6	-26.6	-21.9	-17.5	-13.5
24										82.2	74.4	69.2	64.9	61.0	57.5	54.1	50.9	47.8
										-74.5	-59.7	-50.2	-42.7	-36.3	-30.6	-25.5	-20.7	-16.4
23											80.5	73.0	67.7	63.2	59.2	55.5	52.0	48.7
											-72.0	-58.2	-48.9	-41.4	-35.0	-29.3	-24.2	-19.4
22												78.7	71.4	66.0	61.4	57.3	53.4	49.8
												-69.4	-56.5	-47.4	-40.0	-33.6	-27.9	-22.7
21													76.8	69.6	64.1	59.4	55.1	51.1
													-66.7	-54.6	-45.7	-38.3	-32.0	-26.2
20														74.7	67.6	62.0	57.2	52.7
														-64.0	-52.5	-43.8	-36.5	-30.1
19														85.0	72.5	65.4	59.7	54.7
														-81.6	-61.2	-50.2	-41.7	-34.5
18															80.8	70.1	63.0	57.2
															-75.1	-58.3	-47.8	-39.4
17																77.3	67.4	60.4
																-70.2	-55.3	-45.1
16																	73.9	64.6
																	-65.7	-52.2
15																		70.5
																		-61.5
14																		81.7
																		-78.1

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
55																			
54	81.7	80.8	80.0	79.3	78.7	78.2	77.7	77.3	77.0	76.7	76.5	76.4	76.3	76.4	76.5	76.8	77.2	77.7	78.5
	78.5	77.7	77.2	76.8	76.5	76.4	76.3	76.4	76.5	76.7	77.0	77.3	77.7	78.2	78.7	79.3	80.0	80.8	81.7
53	74.6	73.9	73.3	72.7	72.1	71.6	71.1	70.6	70.2	69.8	69.4	69.1	68.9	68.7	68.5	68.4	68.4	68.4	68.6
	68.6	68.4	68.4	68.4	68.5	68.7	68.9	69.1	69.4	69.8	70.2	70.6	71.1	71.6	72.1	72.7	73.3	73.9	74.6
52	70.1	69.4	68.8	68.1	67.5	66.9	66.4	65.8	65.3	64.8	64.4	64.0	63.6	63.2	62.9	62.6	62.4	62.2	62.1
	62.1	62.2	62.4	62.6	62.9	63.2	63.6	64.0	64.4	64.8	65.3	65.8	66.4	66.9	67.5	68.1	68.8	69.4	70.1
51	66.6	65.9	65.2	64.5	63.9	63.2	62.6	62.0	61.4	60.8	60.3	59.8	59.3	58.8	58.3	57.9	57.5	57.2	56.9
	56.9	57.2	57.5	57.9	58.3	58.8	59.3	59.8	60.3	60.8	61.4	62.0	62.6	63.2	63.9	64.5	65.2	65.9	66.6
50	63.7	63.0	62.2	61.5	60.8	60.1	59.4	58.7	58.1	57.4	56.8	56.2	55.6	55.0	54.5	53.9	53.4	52.9	52.5
	52.5	52.9	53.4	53.9	54.5	55.0	55.6	56.2	56.8	57.4	58.1	58.7	59.4	60.1	60.8	61.5	62.2	63.0	63.7
49	61.3	60.5	59.7	58.9	58.1	57.4	56.6	55.9	55.2	54.4	53.7	53.0	52.3	51.7	51.0	50.4	49.8	49.2	48.6
	48.6	49.2	49.8	50.4	51.0	51.7	52.3	53.0	53.7	54.4	55.2	55.9	56.6	57.4	58.1	58.9	59.7	60.5	61.3
48	59.1	58.3	57.4	56.6	55.8	55.0	54.2	53.3	52.5	51.8	51.0	50.2	49.4	48.7	47.9	47.2	46.5	45.8	45.1
	45.1	45.8	46.5	47.2	47.9	48.7	49.4	50.2	51.0	51.8	52.5	53.3	54.2	55.0	55.8	56.6	57.4	58.3	59.1
47	57.2	56.3	55.4	54.6	53.7	52.8	51.9	51.0	50.2	49.3	48.5	47.6	46.8	45.9	45.1	44.3	43.4	42.6	41.8
	41.8	42.6	43.4	44.3	45.1	45.9	46.8	47.6	48.5	49.3	50.2	51.0	51.9	52.8	53.7	54.6	55.4	56.3	57.2
46	55.5	54.6	53.6	52.7	51.8	50.8	49.9	48.9	48.0	47.1	46.1	45.2	44.3	43.4	42.5	41.5	40.6	39.7	38.8
	38.8	39.7	40.6	41.5	42.5	43.4	44.3	45.2	46.1	47.1	48.0	48.9	49.9	50.8	51.8	52.7	53.6	54.6	55.5
45	54.0	53.0	52.0	51.0	50.0	49.0	48.0	47.0	46.0	45.0	44.0	43.0	42.0	41.0	40.0	39.0	38.0	37.0	36.0
	36.0	37.0	38.0	39.0	40.0	41.0	42.0	43.0	44.0	45.0	46.0	47.0	48.0	49.0	50.0	51.0	52.0	53.0	54.0
44	52.6	51.6	50.5	49.4	48.4	47.3	46.3	45.2	44.1	43.1	42.0	40.9	39.8	38.8	37.7	36.6	35.5	34.4	33.3
	33.3	34.4	35.5	36.6	37.7	38.8	39.8	40.9	42.0	43.1	44.1	45.2	46.3	47.3	48.4	49.4	50.5	51.6	52.6
43	51.4	50.3	49.1	48.0	46.9	45.8	44.7	43.5	42.4	41.3	40.1	39.0	37.8	36.7	35.5	34.3	33.1	31.9	30.8
	30.8	31.9	33.1	34.3	35.5	36.7	37.8	39.0	40.1	41.3	42.4	43.5	44.7	45.8	46.9	48.0	49.1	50.3	51.4
42	50.2	49.1	47.9	46.7	45.5	44.3	43.2	42.0	40.8	39.5	38.3	37.1	35.9	34.6	33.4	32.1	30.9	29.6	28.3
	28.3	29.6	30.9	32.1	33.4	34.6	35.9	37.1	38.3	39.5	40.8	42.0	43.2	44.3	45.5	46.7	47.9	49.1	50.2

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
41	49.2	48.0	46.7	45.5	44.2	43.0	41.7	40.5	39.2	37.9	36.6	35.3	34.0	32.7	31.4	30.0	28.7	27.3	25.9
	25.9	27.3	28.7	30.0	31.4	32.7	34.0	35.3	36.6	37.9	39.2	40.5	41.7	43.0	44.2	45.5	46.7	48.0	49.2
40	48.3	47.0	45.7	44.4	43.1	41.7	40.4	39.1	37.7	36.4	35.0	33.7	32.3	30.9	29.5	28.0	26.6	25.1	23.6
	23.6	25.1	26.6	28.0	29.5	30.9	32.3	33.7	35.0	36.4	37.7	39.1	40.4	41.7	43.1	44.4	45.7	47.0	48.3
39	47.4	46.1	44.7	43.3	42.0	40.6	39.2	37.8	36.4	34.9	33.5	32.0	30.6	29.1	27.6	26.1	24.5	23.0	21.4
	21.4	23.0	24.5	26.1	27.6	29.1	30.6	32.0	33.5	34.9	36.4	37.8	39.2	40.6	42.0	43.3	44.7	46.1	47.4
38	46.7	45.2	43.8	42.4	40.9	39.5	38.0	36.5	35.0	33.5	32.0	30.5	28.9	27.4	25.8	24.2	22.5	20.9	19.2
	19.2	20.9	22.5	24.2	25.8	27.4	28.9	30.5	32.0	33.5	35.0	36.5	38.0	39.5	40.9	42.4	43.8	45.2	46.7
37	46.0	44.5	43.0	41.5	40.0	38.4	36.9	35.3	33.8	32.2	30.6	29.0	27.4	25.7	24.0	22.3	20.6	18.8	17.0
	17.0	18.8	20.6	22.3	24.0	25.7	27.4	29.0	30.6	32.2	33.8	35.3	36.9	38.4	40.0	41.5	43.0	44.5	46.0
36	45.4	43.8	42.2	40.7	39.1	37.5	35.8	34.2	32.6	30.9	29.2	27.5	25.8	24.1	22.3	20.5	18.6	16.8	14.8
	14.8	16.8	18.6	20.5	22.3	24.1	25.8	27.5	29.2	30.9	32.6	34.2	35.8	37.5	39.1	40.7	42.2	43.8	45.4
35	44.9	43.2	41.6	39.9	38.2	36.6	34.9	33.1	31.4	29.7	27.9	26.1	24.3	22.5	20.6	18.7	16.7	14.8	12.7
	12.7	14.8	16.7	18.7	20.6	22.5	24.3	26.1	27.9	29.7	31.4	33.1	34.9	36.6	38.2	39.9	41.6	43.2	44.9
34	44.4	42.7	41.0	39.2	37.5	35.7	33.9	32.1	30.3	28.5	26.6	24.8	22.8	20.9	18.9	16.9	14.9	12.8	10.6
	10.6	12.8	14.9	16.9	18.9	20.9	22.8	24.8	26.6	28.5	30.3	32.1	33.9	35.7	37.5	39.2	41.0	42.7	44.4
33	44.1	42.2	40.4	38.6	36.8	34.9	33.0	31.2	29.3	27.3	25.4	23.4	21.4	19.4	17.3	15.2	13.0	10.8	8.5
	8.5	10.8	13.0	15.2	17.3	19.4	21.4	23.4	25.4	27.3	29.3	31.2	33.0	34.9	36.8	38.6	40.4	42.2	44.1
32	43.8	41.9	40.0	38.0	36.1	34.2	32.2	30.2	28.2	26.2	24.2	22.1	20.0	17.8	15.7	13.4	11.1	8.8	6.4
	6.4	8.8	11.1	13.4	15.7	17.8	20.0	22.1	24.2	26.2	28.2	30.2	32.2	34.2	36.1	38.0	40.0	41.9	43.8
31	43.6	41.6	39.5	37.5	35.5	33.5	31.4	29.4	27.3	25.1	23.0	20.8	18.6	16.3	14.0	11.7	9.3	6.8	4.2
	4.2	6.8	9.3	11.7	14.0	16.3	18.6	20.8	23.0	25.1	27.3	29.4	31.4	33.5	35.5	37.5	39.5	41.6	43.6
30	43.4	41.3	39.2	37.1	35.0	32.8	30.7	28.5	26.3	24.1	21.8	19.6	17.2	14.9	12.4	9.9	7.4	4.8	2.1
	2.1	4.8	7.4	9.9	12.4	14.9	17.2	19.6	21.8	24.1	26.3	28.5	30.7	32.8	35.0	37.1	39.2	41.3	43.4
29	43.4	41.2	38.9	36.7	34.5	32.2	30.0	27.7	25.4	23.1	20.7	18.3	15.9	13.4	10.8	8.2	5.5	2.7	-.2
	-.2	2.7	5.5	8.2	10.8	13.4	15.9	18.3	20.7	23.1	25.4	27.7	30.0	32.2	34.5	36.7	38.9	41.2	43.4
28	43.5	41.1	38.8	36.4	34.1	31.7	29.3	26.9	24.5	22.1	19.6	17.1	14.5	11.9	9.2	6.4	3.6	.6	-2.4
	-2.4	.6	3.6	6.4	9.2	11.9	14.5	17.1	19.6	22.1	24.5	26.9	29.3	31.7	34.1	36.4	38.8	41.1	43.5

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
27	43.6	41.1	38.7	36.2	33.7	31.2	28.7	26.2	23.7	21.1	18.5	15.9	13.1	10.4	7.5	4.6	1.6	-1.5	-4.7
	-4.7	-1.5	1.6	4.6	7.5	10.4	13.1	15.9	18.5	21.1	23.7	26.2	28.7	31.2	33.7	36.2	38.7	41.1	43.6
26	43.9	41.2	38.6	36.0	33.4	30.8	28.2	25.5	22.9	20.2	17.4	14.6	11.8	8.9	5.9	2.8	-.4	-3.7	-7.1
	-7.1	-3.7	-.4	2.8	5.9	8.9	11.8	14.6	17.4	20.2	22.9	25.5	28.2	30.8	33.4	36.0	38.6	41.2	43.9
25	44.2	41.5	38.7	35.9	33.2	30.4	27.7	24.9	22.1	19.3	16.4	13.4	10.4	7.3	4.2	.9	-2.4	-6.0	-9.6
	-9.6	-6.0	-2.4	.9	4.2	7.3	10.4	13.4	16.4	19.3	22.1	24.9	27.7	30.4	33.2	35.9	38.7	41.5	44.2
24	44.8	41.8	38.8	35.9	33.0	30.1	27.2	24.3	21.3	18.4	15.3	12.2	9.0	5.8	2.5	-1.0	-4.6	-8.3	-12.2
	-12.2	-8.3	-4.6	-1.0	2.5	5.8	9.0	12.2	15.3	18.4	21.3	24.3	27.2	30.1	33.0	35.9	38.8	41.8	44.8
23	45.4	42.2	39.1	36.0	33.0	29.9	26.8	23.7	20.6	17.5	14.3	11.0	7.7	4.2	.7	-3.0	-6.8	-10.8	-15.0
	-15.0	-10.8	-6.8	-3.0	.7	4.2	7.7	11.0	14.3	17.5	20.6	23.7	26.8	29.9	33.0	36.0	39.1	42.2	45.4
22	46.3	42.8	39.5	36.2	33.0	29.7	26.5	23.2	19.9	16.6	13.2	9.8	6.2	2.6	-1.1	-5.0	-9.1	-13.4	-17.9
	-17.9	-13.4	-9.1	-5.0	-1.1	2.6	6.2	9.8	13.2	16.6	19.9	23.2	26.5	29.7	33.0	36.2	39.5	42.8	46.3
21	47.3	43.6	40.0	36.5	33.1	29.6	26.2	22.7	19.3	15.7	12.2	8.5	4.8	.9	-3.1	-7.2	-11.5	-16.1	-21.0
	-21.0	-16.1	-11.5	-7.2	-3.1	.9	4.8	8.5	12.2	15.7	19.3	22.7	26.2	29.6	33.1	36.5	40.0	43.6	47.3
20	48.6	44.6	40.7	37.0	33.3	29.6	26.0	22.3	18.6	14.9	11.1	7.3	3.3	-.8	-5.0	-9.5	-14.1	-19.1	-24.4
	-24.4	-19.1	-14.1	-9.5	-5.0	-.8	3.3	7.3	11.1	14.9	18.6	22.3	26.0	29.6	33.3	37.0	40.7	44.6	48.6
19	50.1	45.8	41.6	37.6	33.6	29.7	25.8	22.0	18.0	14.1	10.1	6.0	1.7	-2.6	-7.1	-11.9	-16.9	-22.3	-28.1
	-28.1	-22.3	-16.9	-11.9	-7.1	-2.6	1.7	6.0	10.1	14.1	18.0	22.0	25.8	29.7	33.6	37.6	41.6	45.8	50.1
18	52.1	47.3	42.7	38.4	34.1	29.9	25.8	21.6	17.5	13.3	9.0	4.6	.1	-4.5	-9.4	-14.5	-19.9	-25.8	-32.2
	-32.2	-25.8	-19.9	-14.5	-9.4	-4.5	.1	4.6	9.0	13.3	17.5	21.6	25.8	29.9	34.1	38.4	42.7	47.3	52.1
17	54.5	49.1	44.1	39.4	34.8	30.3	25.8	21.4	17.0	12.5	7.9	3.3	-1.5	-6.5	-11.8	-17.3	-23.2	-29.7	-36.9
	-36.9	-29.7	-23.2	-17.3	-11.8	-6.5	-1.5	3.3	7.9	12.5	17.0	21.4	25.8	30.3	34.8	39.4	44.1	49.1	54.5
16	57.5	51.4	45.8	40.6	35.6	30.8	26.0	21.2	16.5	11.7	6.8	1.8	-3.3	-8.7	-14.4	-20.4	-26.9	-34.1	-42.3
	-42.3	-34.1	-26.9	-20.4	-14.4	-8.7	-3.3	1.8	6.8	11.7	16.5	21.2	26.0	30.8	35.6	40.6	45.8	51.4	57.5
15	61.4	54.3	48.0	42.2	36.7	31.4	26.3	21.2	16.1	10.9	5.7	.3	-5.2	-11.0	-17.2	-23.8	-31.0	-39.2	-48.9
	-48.9	-39.2	-31.0	-23.8	-17.2	-11.0	-5.2	.3	5.7	10.9	16.1	21.2	26.3	31.4	36.7	42.2	48.0	54.3	61.4
14	66.9	58.0	50.7	44.2	38.1	32.3	26.7	21.2	15.7	10.2	4.5	-1.3	-7.3	-13.6	-20.4	-27.7	-35.8	-45.3	-57.4
	-57.4	-45.3	-35.8	-27.7	-20.4	-13.6	-7.3	-1.3	4.5	10.2	15.7	21.2	26.7	32.3	38.1	44.2	50.7	58.0	66.9

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
13	75.9	63.1	54.2	46.7	39.9	33.5	27.4	21.4	15.4	9.4	3.3	-3.0	-9.5	-16.5	-23.9	-32.1	-41.6	-53.1	-70.4
	-70.4	-53.1	-41.6	-32.1	-23.9	-16.5	-9.5	-3.0	3.3	9.4	15.4	21.4	27.4	33.5	39.9	46.7	54.2	63.1	75.9
12		70.9	59.0	50.0	42.2	35.1	28.3	21.7	15.2	8.6	2.0	-4.9	-12.0	-19.7	-28.0	-37.5	-48.7	-64.3	
		-64.3	-48.7	-37.5	-28.0	-19.7	-12.0	-4.9	2.0	8.6	15.2	21.7	28.3	35.1	42.2	50.0	59.0	70.9	
11			65.9	54.4	45.2	37.1	29.5	22.2	15.1	7.9	.6	-6.9	-14.9	-23.4	-32.9	-44.1	-58.5		
			-58.5	-44.1	-32.9	-23.4	-14.9	-6.9	.6	7.9	15.1	22.2	29.5	37.1	45.2	54.4	65.9		
10			79.2	60.7	49.3	39.8	31.2	23.0	15.1	7.2	-.9	-9.3	-18.1	-27.8	-39.0	-52.8	-76.2		
			-76.2	-52.8	-39.0	-27.8	-18.1	-9.3	-.9	7.2	15.1	23.0	31.2	39.8	49.3	60.7	79.2		
9				71.2	55.0	43.4	33.4	24.2	15.3	6.4	-2.6	-11.9	-22.0	-33.3	-46.9	-66.5			
				-66.5	-46.9	-33.3	-22.0	-11.9	-2.6	6.4	15.3	24.2	33.4	43.4	55.0	71.2			
8					63.9	48.5	36.5	25.8	15.7	5.7	-4.4	-15.1	-26.8	-40.4	-58.4				
					-58.4	-40.4	-26.8	-15.1	-4.4	5.7	15.7	25.8	36.5	48.5	63.9				
7						56.3	40.9	28.2	16.4	5.0	-6.7	-19.1	-33.0	-50.4					
						-50.4	-33.0	-19.1	-6.7	5.0	16.4	28.2	40.9	56.3					
6						71.2	47.7	31.7	17.7	4.3	-9.4	-24.2	-41.7	-68.3					
						-68.3	-41.7	-24.2	-9.4	4.3	17.7	31.7	47.7	71.2					
5							59.6	37.2	19.8	3.5	-13.0	-31.4	-55.8						
							-55.8	-31.4	-13.0	3.5	19.8	37.2	59.6						
4								46.9	23.3	2.8	-18.0	-42.9							
								-42.9	-18.0	2.8	23.3	46.9							
3								71.1	30.0	2.1	-26.2	-69.6							
								-69.6	-26.2	2.1	30.0	71.1							
2									46.0	1.4	-44.0								
									-44.0	1.4	46.0								
1										.7									
										.7									
0										.0									
										-.0									

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
55																		
54	79.5	80.9	82.9	86.5														
	82.7	83.9	85.4	87.8														
53	68.8	69.1	69.5	70.1	70.9	71.9	73.1	74.7	76.8	79.9	85.5							
	75.3	76.1	76.9	77.7	78.7	79.6	80.7	81.9	83.3	85.1	87.9							
52	62.0	62.0	62.0	62.2	62.4	62.8	63.3	63.9	64.7	65.7	66.9	68.6	70.6	73.4	77.3	85.0		
	70.8	71.5	72.3	73.1	73.9	74.7	75.6	76.5	77.4	78.4	79.5	80.6	81.9	83.4	85.2	88.2		
51	56.6	56.4	56.2	56.1	56.1	56.1	56.2	56.4	56.7	57.1	57.7	58.4	59.4	60.5	62.0	64.0	66.4	69.7
	67.3	68.1	68.8	69.6	70.4	71.2	72.0	72.9	73.8	74.6	75.6	76.5	77.5	78.5	79.6	80.8	82.1	83.5
50	52.1	51.7	51.3	51.0	50.8	50.6	50.4	50.3	50.3	50.4	50.5	50.8	51.2	51.7	52.4	53.4	54.6	56.1
	64.5	65.3	66.1	66.9	67.7	68.5	69.3	70.1	71.0	71.9	72.8	73.7	74.6	75.5	76.5	77.5	78.5	79.6
49	48.0	47.5	47.0	46.5	46.1	45.7	45.3	45.0	44.8	44.6	44.4	44.4	44.4	44.5	44.7	45.1	45.6	46.3
	62.1	62.9	63.7	64.5	65.4	66.2	67.1	67.9	68.8	69.7	70.5	71.4	72.3	73.2	74.2	75.1	76.1	77.0
48	44.4	43.7	43.1	42.5	41.9	41.3	40.8	40.3	39.8	39.4	39.0	38.7	38.4	38.2	38.0	38.0	38.0	38.2
	60.0	60.8	61.7	62.6	63.4	64.3	65.2	66.1	67.0	67.9	68.8	69.7	70.6	71.5	72.4	73.3	74.3	75.2
47	41.1	40.3	39.5	38.8	38.0	37.3	36.6	35.9	35.3	34.6	34.0	33.5	32.9	32.4	32.0	31.6	31.2	31.0
	58.1	59.0	59.9	60.8	61.8	62.7	63.6	64.5	65.4	66.3	67.3	68.2	69.1	70.1	71.0	71.9	72.9	73.8
46	37.9	37.0	36.2	35.3	34.4	33.6	32.7	31.9	31.0	30.2	29.4	28.6	27.8	27.1	26.3	25.6	24.9	24.3
	56.5	57.4	58.4	59.3	60.3	61.2	62.2	63.2	64.1	65.1	66.0	67.0	68.0	68.9	69.9	70.8	71.8	72.8
45	35.0	34.0	33.0	32.0	31.0	30.0	29.0	28.0	27.0	26.0	25.0	24.0	23.0	22.0	21.0	20.0	19.0	18.0
	55.0	56.0	57.0	58.0	59.0	60.0	61.0	62.0	63.0	64.0	65.0	66.0	67.0	68.0	69.0	70.0	71.0	72.0
44	32.2	31.1	30.0	28.9	27.7	26.6	25.5	24.3	23.2	22.0	20.8	19.6	18.4	17.1	15.9	14.6	13.3	11.9
	53.7	54.7	55.8	56.8	57.9	58.9	60.0	61.0	62.1	63.1	64.2	65.2	66.2	67.3	68.3	69.4	70.4	71.5
43	29.5	28.3	27.1	25.9	24.6	23.3	22.1	20.8	19.4	18.1	16.7	15.3	13.9	12.4	10.9	9.3	7.7	6.0
	52.5	53.6	54.7	55.8	56.9	58.0	59.1	60.2	61.3	62.4	63.5	64.6	65.7	66.8	67.9	69.0	70.0	71.1
42	27.0	25.7	24.3	23.0	21.6	20.2	18.8	17.3	15.8	14.3	12.7	11.1	9.5	7.8	6.0	4.2	2.2	.2
	51.4	52.6	53.7	54.9	56.0	57.2	58.3	59.5	60.6	61.8	62.9	64.1	65.2	66.4	67.6	68.7	69.9	71.0

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
41	24.5	23.1	21.6	20.1	18.6	17.1	15.5	13.9	12.3	10.6	8.8	7.0	5.2	3.2	1.2	-1.0	-3.3	-5.7
	50.4	51.6	52.9	54.1	55.3	56.5	57.7	58.9	60.1	61.4	62.6	63.8	65.0	66.2	67.4	68.7	69.9	71.1
40	22.1	20.6	19.0	17.4	15.8	14.1	12.4	10.6	8.8	6.9	5.0	2.9	.8	-1.4	-3.7	-6.2	-8.8	-11.7
	49.5	50.8	52.1	53.4	54.7	55.9	57.2	58.5	59.8	61.0	62.3	63.6	64.9	66.2	67.5	68.8	70.1	71.4
39	19.7	18.1	16.4	14.7	12.9	11.1	9.2	7.3	5.3	3.2	1.1	-1.2	-3.5	-6.0	-8.7	-11.5	-14.5	-17.8
	48.8	50.1	51.5	52.8	54.1	55.5	56.8	58.2	59.5	60.9	62.2	63.6	64.9	66.3	67.7	69.1	70.5	72.0
38	17.4	15.7	13.8	12.0	10.1	8.1	6.1	4.0	1.8	-.4	-2.8	-5.3	-7.9	-10.7	-13.7	-16.9	-20.4	-24.3
	48.1	49.5	50.9	52.3	53.7	55.1	56.6	58.0	59.4	60.8	62.2	63.7	65.1	66.6	68.1	69.6	71.2	72.8
37	15.1	13.3	11.3	9.3	7.3	5.2	3.0	.7	-1.7	-4.2	-6.8	-9.5	-12.4	-15.6	-18.9	-22.6	-26.7	-31.2
	47.5	49.0	50.4	51.9	53.4	54.9	56.4	57.9	59.4	60.9	62.4	64.0	65.5	67.1	68.7	70.4	72.1	73.9
36	12.9	10.9	8.8	6.7	4.5	2.2	-.2	-2.6	-5.2	-7.9	-10.8	-13.8	-17.1	-20.6	-24.4	-28.6	-33.4	-38.8
	47.0	48.5	50.1	51.6	53.2	54.8	56.3	57.9	59.5	61.1	62.7	64.4	66.1	67.8	69.5	71.4	73.3	75.3
35	10.6	8.5	6.3	4.0	1.7	-.8	-3.3	-6.0	-8.8	-11.8	-14.9	-18.3	-21.9	-25.9	-30.3	-35.2	-40.8	-47.4
	46.5	48.2	49.8	51.4	53.1	54.7	56.4	58.1	59.8	61.5	63.2	65.0	66.8	68.7	70.6	72.7	74.9	77.3
34	8.4	6.1	3.8	1.3	-1.2	-3.8	-6.6	-9.5	-12.5	-15.8	-19.2	-23.0	-27.0	-31.5	-36.6	-42.4	-49.3	-58.1
	46.2	47.9	49.6	51.3	53.1	54.8	56.6	58.4	60.2	62.0	63.9	65.8	67.8	69.9	72.0	74.4	77.0	80.1
33	6.1	3.7	1.2	-1.4	-4.1	-6.9	-9.9	-13.0	-16.4	-19.9	-23.7	-27.9	-32.5	-37.7	-43.7	-50.8	-60.0	-74.8
	45.9	47.7	49.5	51.3	53.2	55.0	56.9	58.8	60.7	62.7	64.7	66.8	69.0	71.4	73.9	76.7	80.1	85.1
32	3.9	1.3	-1.4	-4.1	-7.1	-10.1	-13.3	-16.7	-20.4	-24.3	-28.5	-33.2	-38.5	-44.6	-52.0	-61.5	-77.7	
	45.7	47.6	49.5	51.4	53.4	55.4	57.4	59.4	61.5	63.6	65.8	68.1	70.6	73.3	76.3	80.0	85.8	
31	1.6	-1.2	-4.0	-7.0	-10.1	-13.4	-16.9	-20.6	-24.6	-28.9	-33.7	-39.1	-45.3	-52.7	-62.5	-80.2		
	45.6	47.6	49.6	51.7	53.7	55.8	58.0	60.1	62.4	64.7	67.2	69.8	72.6	75.8	79.8	86.4		
30	-.8	-3.7	-6.7	-9.9	-13.2	-16.8	-20.6	-24.6	-29.0	-33.9	-39.4	-45.6	-53.2	-63.2	-82.0			
	45.6	47.7	49.8	52.0	54.2	56.4	58.7	61.1	63.5	66.1	68.9	71.9	75.3	79.5	86.9			
29	-3.1	-6.2	-9.5	-12.9	-16.5	-20.3	-24.5	-29.0	-33.9	-39.4	-45.7	-53.4	-63.4	-82.9				
	45.6	47.9	50.2	52.5	54.8	57.2	59.7	62.3	65.0	67.9	71.0	74.6	79.1	87.1				
28	-5.6	-8.9	-12.4	-16.0	-19.9	-24.1	-28.7	-33.6	-39.2	-45.6	-53.2	-63.3	-82.5					
	45.8	48.2	50.6	53.1	55.6	58.2	60.9	63.7	66.7	70.0	73.8	78.5	86.8					

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
27	-8.1	-11.6	-15.4	-19.3	-23.6	-28.1	-33.2	-38.8	-45.2	-52.8	-62.8	-81.1						
	46.1	48.7	51.2	53.9	56.6	59.4	62.4	65.5	69.0	72.9	77.7	86.1						
26	-10.7	-14.5	-18.5	-22.8	-27.4	-32.5	-38.1	-44.6	-52.2	-62.1	-79.2							
	46.5	49.2	52.0	54.8	57.8	60.9	64.2	67.7	71.8	76.8	85.0							
25	-13.5	-17.5	-21.9	-26.6	-31.6	-37.3	-43.7	-51.3	-61.0	-76.9								
	47.1	50.0	52.9	56.0	59.2	62.7	66.4	70.6	75.7	83.7								
24	-16.4	-20.7	-25.5	-30.6	-36.3	-42.7	-50.2	-59.7	-74.5									
	47.8	50.9	54.1	57.5	61.0	64.9	69.2	74.4	82.2									
23	-19.4	-24.2	-29.3	-35.0	-41.4	-48.9	-58.2	-72.0										
	48.7	52.0	55.5	59.2	63.2	67.7	73.0	80.5										
22	-22.7	-27.9	-33.6	-40.0	-47.4	-56.5	-69.4											
	49.8	53.4	57.3	61.4	66.0	71.4	78.7											
21	-26.2	-32.0	-38.3	-45.7	-54.6	-66.7												
	51.1	55.1	59.4	64.1	69.6	76.8												
20	-30.1	-36.5	-43.8	-52.5	-64.0													
	52.7	57.2	62.0	67.6	74.7													
19	-34.5	-41.7	-50.2	-61.2	-81.6													
	54.7	59.7	65.4	72.5	85.0													
18	-39.4	-47.8	-58.3	-75.1														
	57.2	63.0	70.1	80.8														
17	-45.1	-55.3	-70.2															
	60.4	67.4	77.3															
16	-52.2	-65.7																
	64.6	73.9																
15	-61.5																	
	70.5																	
14	-78.1																	
	81.7																	

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
55																		
54																		
53																		
52																		
51	74.5 65.3	83.9 88.3																
50	58.0 80.7	60.6 81.9	64.0 83.2	68.7 84.8	76.6 86.9													
49	47.3 78.0	48.5 79.1	50.2 80.1	52.3 81.2	55.1 82.4	58.9 83.7	64.5 85.2	73.8 87.2										
48	38.5 76.2	39.0 77.1	39.7 78.1	40.7 79.1	42.0 80.1	43.8 81.2	46.3 82.3	49.6 83.4	54.4 84.7	61.7 86.2	76.4 88.4							
47	30.8 74.8	30.7 75.7	30.8 76.7	31.0 77.7	31.4 78.6	32.0 79.6	33.0 80.6	34.4 81.6	36.4 82.7	39.2 83.7	43.4 84.9	50.0 86.1	62.1 87.7					
46	23.7 73.7	23.1 74.7	22.6 75.7	22.2 76.7	21.9 77.6	21.6 78.6	21.5 79.6	21.6 80.6	22.0 81.6	22.7 82.5	23.8 83.6	25.8 84.6	29.0 85.6	34.6 86.7	45.9 87.9			
45	17.0 73.0	16.0 74.0	15.0 75.0	14.0 76.0	13.0 77.0	12.0 78.0	11.0 79.0	10.0 80.0	9.0 81.0	8.0 82.0	7.0 83.0	6.0 84.0	5.0 85.0	4.0 86.0	3.0 87.0	2.0 88.0	1.0 89.0	-.0 90.0
44	10.6 72.5	9.1 73.6	7.6 74.6	6.1 75.6	4.5 76.7	2.7 77.7	.8 78.8	-1.2 79.8	-3.5 80.9	-6.1 82.0	-9.2 83.0	-13.1 84.1	-18.1 85.2	-25.4 86.4	-37.8 87.6	-74.7 89.5		
43	4.3 72.2	2.4 73.4	.4 74.5	-1.7 75.6	-4.0 76.7	-6.5 77.8	-9.3 78.9	-12.5 80.1	-16.2 81.3	-20.7 82.4	-26.3 83.7	-33.8 85.0	-45.1 86.5	-68.1 88.5				
42	-2.0 72.2	-4.3 73.4	-6.8 74.6	-9.5 75.8	-12.6 77.0	-15.9 78.2	-19.8 79.5	-24.4 80.8	-29.9 82.1	-36.9 83.5	-46.6 85.2	-62.8 87.2						
	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90

PSI

FIRST LINE, ALPHA1, ANGLE TO N-S LINE FROM HORIZONTAL
SECOND LINE, ALPHA2, ANGLE TO E-W LINE FROM HORIZONTAL

GAMMA \ PSI	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
41	-8.3	-11.1	-14.2	-17.6	-21.5	-25.9	-31.1	-37.6	-45.9	-58.2								
	72.4	73.7	74.9	76.3	77.6	79.0	80.4	82.0	83.7	85.8								
40	-14.8	-18.1	-21.9	-26.1	-31.0	-36.9	-44.2	-53.9	-70.2									
	72.8	74.2	75.6	77.1	78.6	80.2	82.0	84.0	86.9									
39	-21.5	-25.5	-30.1	-35.4	-41.8	-50.0	-61.4											
	73.5	75.0	76.6	78.3	80.1	82.1	84.7											
38	-28.6	-33.5	-39.2	-46.1	-55.1	-68.9												
	74.4	76.2	78.0	80.1	82.4	85.6												
37	-36.4	-42.4	-49.9	-59.8	-77.5													
	75.7	77.8	80.1	82.8	87.1													
36	-45.1	-53.1	-64.0															
	77.5	80.1	83.3															
35	-55.8	-67.9																
	80.1	83.8																
34	-71.5																	
	84.4																	
33																		
32																		
31																		
30																		
29																		
28																		

KANSAS GEOLOGICAL SURVEY COMPUTER PROGRAM
THE UNIVERSITY OF KANSAS, LAWRENCE

PROGRAM ABSTRACT

Title (If subroutine state in title):

FORTRAN IV CDC 6400 computer program for constructing isometric diagrams.

Date: 29 August 1969

Author, organization: William B. Wray, Jr.

Dept. of Geology and Geophysics, Univ. of California, Berkeley

Direct inquiries to: Author

Name: Address:

Purpose/description: Subroutines to provide a method of displaying a surface in isometric pictorial representation.

Mathematical method: Point translation from (x, y, z) space to (u, v) space.

Restrictions, range: No restrictions except disc or other fast-access storage unit capacity.

Computer manufacturer: CDC Model: 6400

Programming language: FORTRAN IV

Memory required: * K Approximate running time: variable; linear with number of points defining surface.

Special peripheral equipment required: digital plotter

Remarks (special compilers or operating systems, required word lengths, number of successful runs, other machine versions, additional information useful for operation or modification of program)

Run successfully many times.

* 7K + work array + user general plotter routines (see text)

(continued from inside front cover)

12. Computer applications in the earth sciences; Colloquium on trend analysis, edited by D.F. Merriam and N.C. Cocke, 1967 . . . . $1.00
13. FORTRAN IV computer programs for Markov chain experiments in geology, by W.C. Krumbein, 1967 . . . . $1.00
14. FORTRAN IV programs to determine surface roughness in topography for the CDC 3400 computer, by R.D. Hobson, 1967 . . . . $1.00
15. FORTRAN II program for progressive linear fit of surfaces on a quadratic base using an IBM 1620 computer, by A.J. Cole, C. Jordan, and D.F. Merriam, 1967 . . . . . $1.00
16. FORTRAN IV program for the GE 625 to compute the power spectrum of geological surfaces, by J.E. Esler and F.W. Preston, 1967 . . . . $0.75
17. FORTRAN IV program for Q-mode cluster analysis of nonquantitative data using IBM 7090/7094 computers, by G.F. Bonham-Carter, 1967 . . . . $1.00
18. Computer applications in the earth sciences: Colloquium on time-series analysis, edited by D.F. Merriam, 1967. . . . . $1.00
19. FORTRAN II time-trend package for the IBM 1620 computer, by J.C. Davis and R.J. Sampson, 1967 . . . . $1.00
20. Computer programs for multivariate analysis in geology, edited by D.F. Merriam, 1968 . $1.00
21. FORTRAN IV program for computation and display of principal components, by W.J. Wahlstedt and J.C. Davis, 1968 . . . . $1.00
22. Computer applications in the earth sciences: Colloquium on simulation, edited by D.F. Merriam and N.C. Cocke, 1968 . . . . $1.00
23. Computer programs for automatic contouring, by D.B. McIntyre, D.D. Pollard, and R. Smith, 1968 . . . . $1.50
24. Mathematical model and FORTRAN IV program for computer simulation of deltaic sedimentation, by G.F. Bonham-Carter and A.J. Sutherland, 1968 . . . . $1.00
25. FORTRAN IV CDC 6400 computer program to analyze subsurface fold geometry, by E.H.T. Whitten, 1968 . . . . $1.00
26. FORTRAN IV computer program for simulation of transgression and regression with continuous-time Markov models, by W.C. Krumbein, 1968 . . . . $1.00
27. Stepwise regression and nonpolynomial models in trend analysis, by A.T. Miesch and J.J. Connor, 1968 . . . . $1.00
28. KWIKR8 a FORTRAN IV program for multiple regression and geologic trend analysis, by J.E. Esler, P.F. Smith, and J.C. Davis, 1968 . . . . $1.00
29. FORTRAN IV program for harmonic trend analysis using double Fourier series and regularly gridded data for the GE 625 computer, by J.W. Harbaugh and M.J. Sackin, 1968 . . $1.00
30. Sampling a geological population, by J.C. Griffiths and C.W. Ondrick, 1968 . . . $1.00
31. Multivariate procedures and FORTRAN IV program for evaluation and improvement of classifications, by Ferruh Demirmen, 1969 . . . . $1.00
32. FORTRAN IV programs for canonical correlation and canonical trend-surface analysis, by P.J. Lee, 1969 . . . . $1.00
33. FORTRAN IV program for construction of Pi diagrams with the Univac 1108 computer, by Jeffrey Warner, 1969 . . . . $1.00
34. FORTRAN IV program for nonlinear estimation, by R.B. McCammon, 1969 . . . . . $0.75
35. FORTRAN IV computer program for fitting observed count data to discrete distribution models of binomial, Poisson and negative binomial, by C.W. Ondrick and J.C. Griffiths, 1969 . $0.75
36. GRAFPAC, graphic output subroutines for the GE 635 computer, by F.J. Rohlf, 1969 . . $1.00
37. An iterative approach to the fitting of trend surfaces, by A.J. Cole, 1969 . . . . . $1.00
38. FORTRAN II programs for 8 methods of cluster analysis (CLUSTAN I), by David Wishart, 1969 $1.50
39. FORTRAN IV program for the generalized statistical distance and analysis of covariance matrices for the CDC 3600 computer, by R.A. Reyment, Hans-Ake Ramden, and W.J. Wahlstedt, 1969 . . . . $1.00
40. Symposium on computer applications in petroleum exploration, edited by D.F. Merriam, 1969 $1.00
41. FORTRAN IV program for sample normality tests, by D.A. Preston, 1970 . . . . . $1.00
42. CORFAN-FORTRAN IV computer program for correlation, factor analysis (R- and Q-mode) and varimax rotation, by C.W. Ondrick and G.S. Srivastava, 1970 . . . . . $1.50
43. Minimum entropy criterion for analytic rotation, by R.B. McCammon, 1970 . . . . $1.00
44. FORTRAN IV CDC 6400 computer program for constructing isometric diagrams, by W.B. Wray, Jr, 1970 . . . . $1.00