COMPUTER DRAFTED SHAFTS

A BASIC SYSTEM

FOR

COMPUTERIZED DRAFTING

OF

POWER TRAIN SHAFTS

By

JOHN K. NEWLIN, B.Sc.

A Thesis

Submitted to the Faculty of Graduate Studies

in Partial Fulfillment of the Requirements

for the Degree

Master of Engineering

McMaster University

May 1969

MASTER OF DESIGN (1969)
(Mechanical Engineering)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: A Basic System for Computerized Drafting of Power Train Shafts

AUTHOR: John Kenton Newlin, B. Sc. (Northwestern University)

SUPERVISORS: Professor G. Kardos and Professor W. R. Newcombe

NUMBER OF PAGES: viii, 177

SCOPE AND CONTENTS:

A set of parameters is described by which a designer can easily express a shaft to a computer numerically. Relationships of these parameters to different shaft geometries are described. Logic is provided for translation from the designer's parameters into parameters convenient for manipulation by the computer and for detecting inconsistencies in the designer's description of a shaft.

Details are given of logic by which a computer can plan locations for shaft dimensioning, along with logic for the drawing and dimensioning of shafts by a digital plotter.

The full listing is given in FORTRAN for a computer program and ten subprograms embodying the logic. Shafts drawn by the

computer appear, and the economic feasibility of computer drafting of shafts is discussed.

ACKNOWLEDGEMENTS

The author would like to thank International Harvester of Hamilton, whose cooperation made the arrangement of this work possible.

The author would also like to thank John O. Nicholson of International Harvester Engineering Research, Chicago, who proposed the subject of this thesis, and who provided the definition of the problem.

He would like to thank his advisors at McMaster, Professor W. R. Newcombe and Dr. G. Kardos, who were able to draw upon their experience in design and graphics to give him the guidance he often needed.

Finally, the author would like to acknowledge his great indebtedness to his wife, Barbara, who helped and encouraged him at every stage, keypunching the program and editing and typing the manuscript.

TABLE OF CONTENTS

Appendices

LIST OF ILLUSTRATIONS

I ABSTRACT

The basic program has been developed through which a designer can transform his concept of a power train shaft design into an engineering drawing by means of computer graphics.

One view including dimensioning is drawn of a shaft composed of cylindrical and tapered sections with fillets. The input is in the designer's terms. It consists of the dimensions immediately available to him, and it minimizes special codes and calculations. Steps have been taken to make the system convenient and flexible so that drawings can be changed easily as designs develop or are modified. The computer both embellishes the shaft description and translates it into terms convenient for its use. At the same time, it checks the consistency and uniqueness of the input.

For the task of dimensioning, a set of standardized formats were selected: 3 for angles, 4 for lengths, 2 for diameters, 1 for fillet radii, 2 for chamfers and 120 for the expression of numerical dimensions. The decision process by which the draftsman normally arranges the dimensioning has been formalized and, insofar as possible, imitated by the computer logic. Limitations and shortcomings of this logic have been noted.

The computer program embodying the logic accepts card input and produces drawings on the digital plotter. Observation of these drawings

has produced a basis for future improvement of the logic processes. The economic practicality of computer graphics in this application has also been discussed.

II INTRODUCTION

This work was undertaken with the sponsorship of International Harvester Company, and as such has certain implicit working guidelines in addition to the simple goal of producing a computer program and logic for drawing shafts.

It is, in fact, the industry-oriented framework which renders this work a design project rather than merely an exercise.

International Harvester is developing design-aid systems which will be available corporation-wide. These systems center around the high speed digital computer, and a portion of the man-machine communication and the documatation utilizes the digital plotter. In the future, International Harvester will also have CRT input-output facilities.

One of the design-aid systems now being developed is for power train design. Several parts of this system, such as gear design programs and a shaft bending and fatigue analysis program, are already in existence. The subject of this thesis is the development of another section of the eventual system--one that will produce shaft drawings. Because McMaster and International Harvester have different computers and different plotters, the essential fruit of this thesis is the logic of the computer graphics program. A program was written for McMaster's IBM 7040 computer and Benson Lehner plotter to test and verify the underlying logic.

The program's somewhat limited scope is justified by the "cut and paste" philosophy. The single view of the shaft which the program produces does not pretend to be an entire engineering drawing. Since all engineering drawings at International Harvester are stored and reproduced through the medium of microfilm, a complete drawing can be assembled out of separate pieces and unified photographically. In this way, the computer drawing which does not include a border, for instance, can be pasted on a standard piece of drawing paper with a border. Because it would not be practical to assemble a logic anticipating all the myriad possible features of a shaft, this position was adopted: at all stages of refinement the program should be able to draw as many shafts as possible. Those shafts which possess features not encompassed by the program should at least be expressible to the program so that it can partially complete their drawing, allowing a draftsman to add the other features.

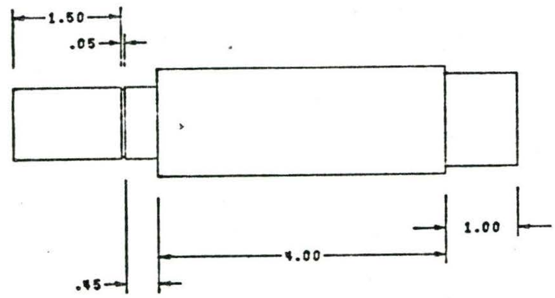
Throughout the work, the value of including allowance for various features was estimated by surveying sample International Harvester shaft drawings. Further guidance on this matter was provided by John Nicholson, Dr. Kardos and Professor Newcombe.

III DESCRIPTION OF THE INPUT

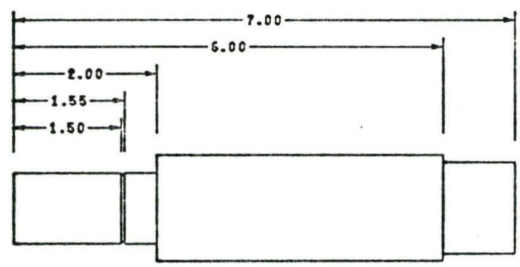
A power train shaft can be considered as an assemblage of cylindrical and tapered elements joined end-to-end with fillets at the joints.

In devising a system of parameters to describe a shaft, the first attempt was to describe the sections individually. Each section has a length, and one diameter if it is cylindrical; or, if it is a tapered section, two diameters or a diameter and taper angle. Furthermore, a section can have one or two fillets at its ends. The result of this approach is a shaft with serial dimensioning (see Fig. I-A). But because of manufacturing tolerances, shafts are not always dimensioned serially. Nor is baseline dimensioning always used (see Fig. I-B). The designer requires the order of dimensioning which best fits his purposes (see Fig. I-C). Any order is valid that is definitive.

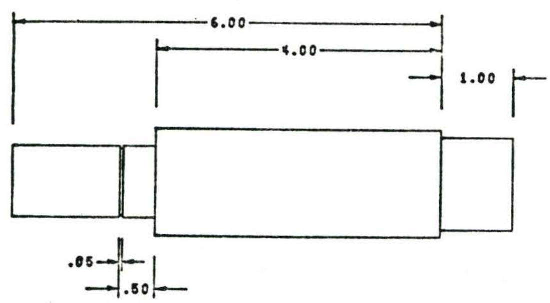
Because of its weaknesses, the first method was discarded in favor of a second, which associates all parameters with planes rather than segments. The planes are those which are perpendicular to the center line of the shaft and appear as vertical lines on the shaft in the drawing. The length dimensions are placed between these planes. In serial dimensioning (Fig. I-A), every plane is located with respect to its neighbor(s). In baseline dimensioning (Fig. I-B), all planes are located with respect to the leftmost plane.



A -- Serial Dimensioning



B -- Baseline Dimensioning



C -- Mixed Dimensioning

Figure 1 Alternate Methods of Parameterization

In the generalized shaft dimensioning system (Fig. I-C), any plane may be located with respect to any other as long as the location of all planes is achieved.

In general, for a shaft of n planes the location of all planes can be achieved with n-1 length dimensions. If there are more than n-1 length dimensions, then there is either an inconsistency or a redundancy. Redundant dimensioning is allowed as long as only n-1 of the dimensions are locating dimensions, the remainder being reference dimensions. A shaft with n planes is also composed of n-1 segments in the sense of the first approach, and it is therefore possible to express a shaft of n planes in n-1 groups of parameters.

III. 1 Basic Parameters

The leftmost plane in the drawing was arbitrarily chosen to be without a group of parameters, and it is numbered zero. Each of the remaining planes has associated with it: I, a unique, non-zero number; D, a diameter; FR, a fillet radius; TANG, a taper angle; IREF, the number of the plane with respect to which it is located; and R, the locating length dimension. The way in which the parameters represent different configurations can be seen in Fig. II. The diameter associated with plane I--D(I)--is always to the left of plane I. TANG is always positive when the diameter is increasing to the right. When one of the n-1 locating dimensions is given between two planes,

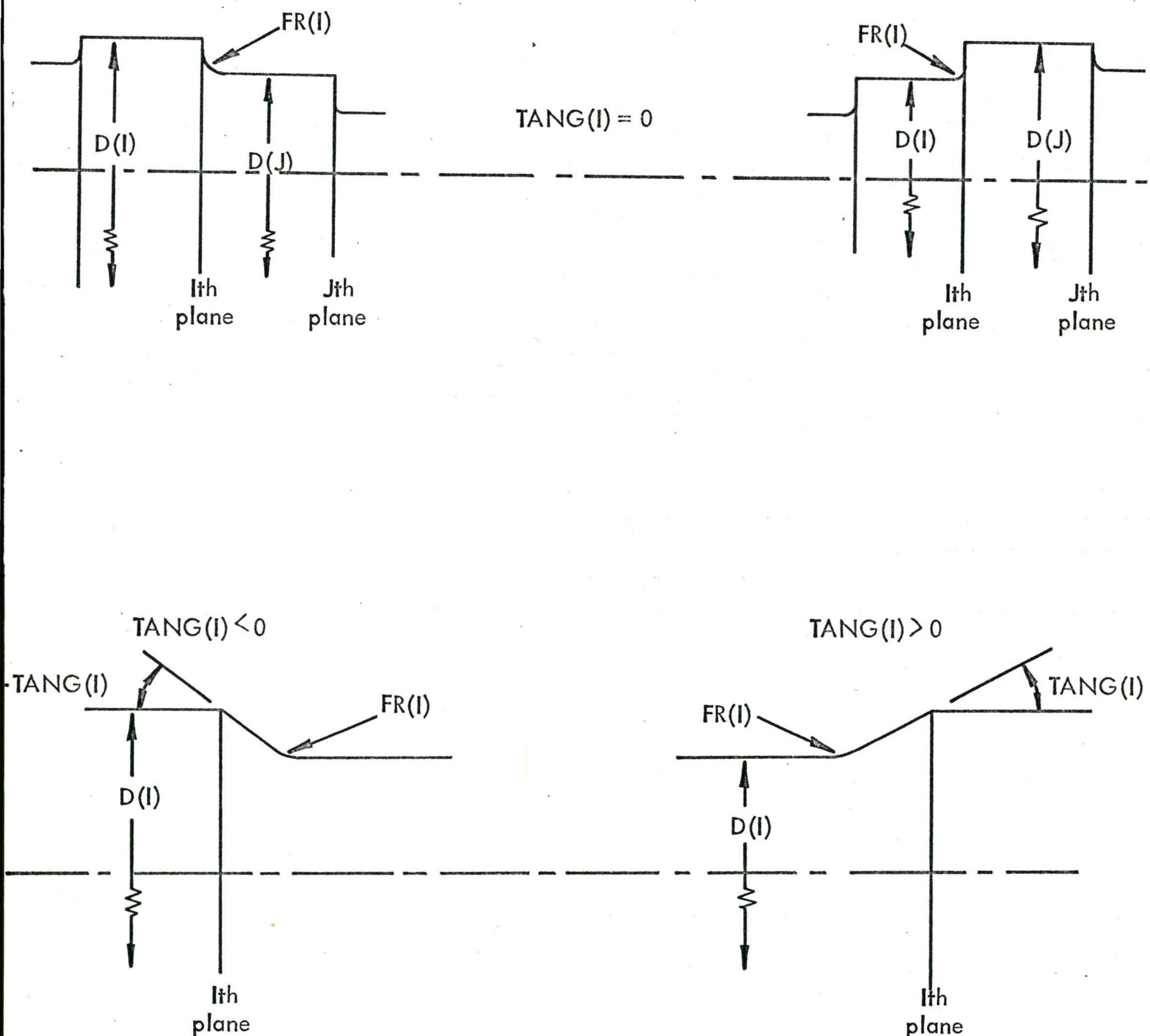


Figure II Legal Configurations

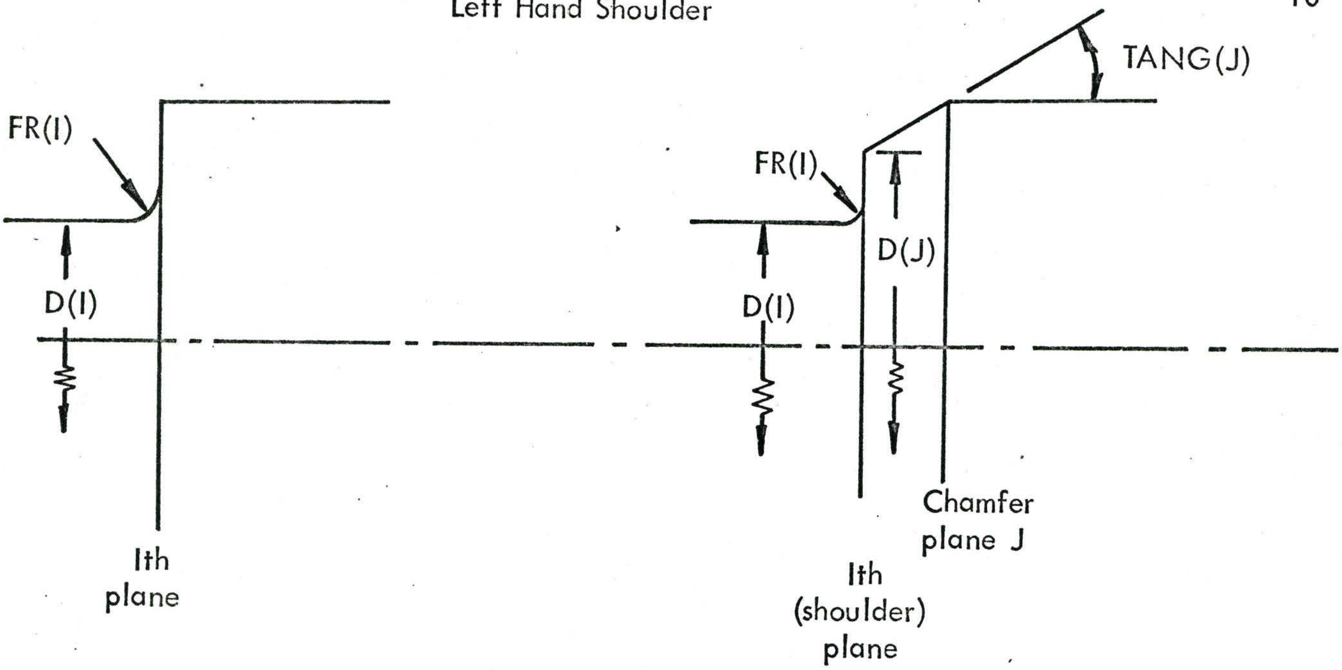
one of them (and only one of them) must have the number of the other as its IREF, and its R equal to the locating dimension. If plane IREF (I) is to the right of plane I, then R must be negative. The assignment of the n-1 locating dimensions to the n-1 planes is unique.

One further parameter, IREFQ, provides for up to n-1 reference dimensions. IREFQ (k) = n indicates that a dimension is desired between planes k and n. When a reference dimension is desired in the output, its length need not be given as input, for the other parameters contain enough information to derive the distance between any two planes. Although this system has a large capacity, it is not capable of expressing all schemes of reference dimensioning. For convenience, IREFQ = 0 (left blank on a FORTRAN input card) means that no reference dimension is intended. To distinguish the case of a reference dimension to the leftmost plane, a minus one (-1) must be used where one is intended.

III. 2 Tolerances

Three alternate methods of expressing tolerances were considered: 1) minimum and maximum metal conditions, 2) nominal dimension with equal plus and minus tolerances, 3) nominal dimension with plus and minus tolerances not necessarily equal. Although the third has the greatest flexibility, the first two have the advantage of two rather than three parameters for their expression. Based on new International Harvester standards, the second alternative was chosen. Lengths, diameters and fillet radii are allowed

Left Hand Shoulder



Right Hand Shoulder

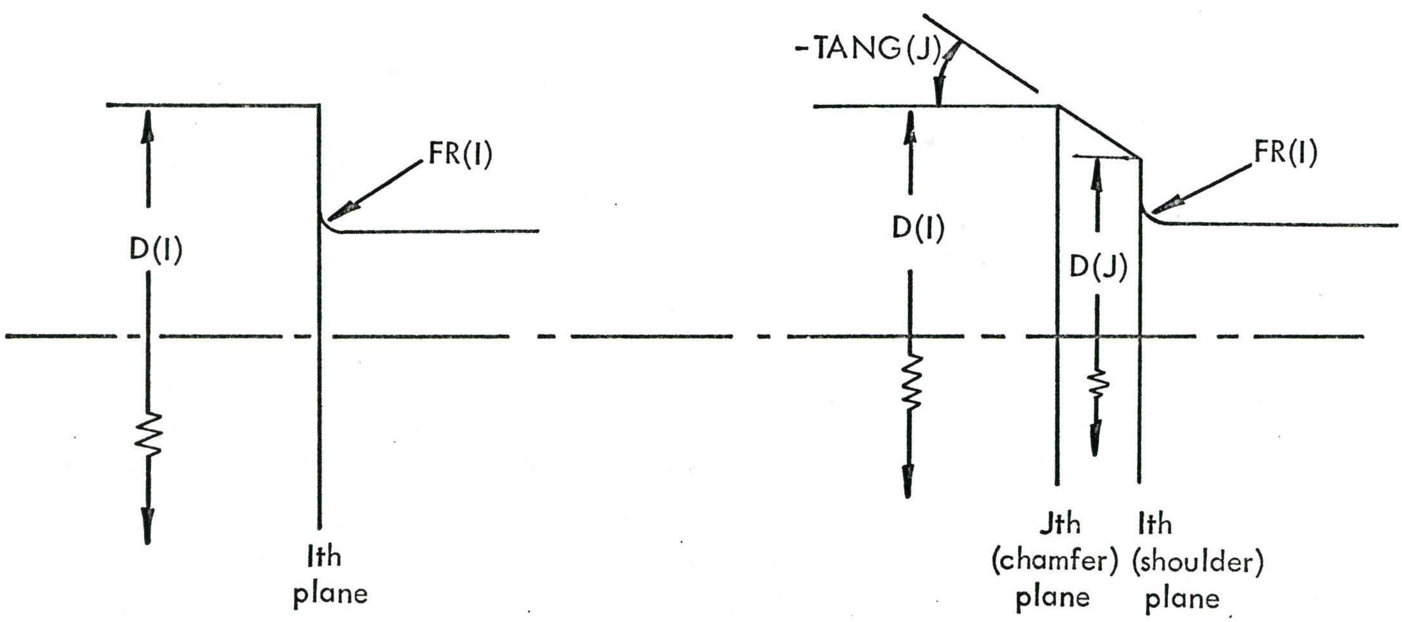


Figure III   Chamfer Plane Parameters

tolerances, although angles are expressed only to the nearest degree without tolerance. The convention of implied tolerancing is recognized in the input for its convenience, and in the output because it is accepted by International Harvester standards. By this convention, one figure written after the decimal implies a tolerance of $\pm$ .060, two figures imply a tolerance of $\pm$ .030 and three imply a tolerance of $\pm$ .010, where all dimensions are in inches. Because FORTRAN does not distinguish between trailing zeroes and blanks, a code is used to show which is intended. To indicate n figures written after the decimal of the nominal dimension, -n is given for the tolerance.

III. 3 The Chamfer Plane

Plane I is a chamfer plane if FR (I) = 0, and TANG $\neq$ 0. The chamfer has been treated as an independent unit. One can be added to or removed from a shaft as a unit, without altering the parameters associated with other planes. The shoulder plane is the plane to the left of a chamfer plane I if TANG (I) $>$ 0, or to the right of I if TANG (I) $<$ 0. The case of the right hand shoulder requires one exception to the rule that D is the diameter to the left of the plane with which it is associated. In this case when the chamfer is present, D of the shoulder plane is the diameter to the left of the chamfer plane (see Fig. III).

Two methods of dimensioning a chamfer have been provided. If the shoulder diameter (seen as D(J) in Fig. III) is to be used, then R must be zero in the input. If the chamfer plane is to be located by giving R, then the diameter D must be left as zero. Either the diameter or the plane location with angle fully describes the chamfer, and to give both would be redundant.

If a chamfer is dimensioned by locating the chamfer plane with respect to the shoulder plane, if the distance is less then .5 inches and if the angle is 45°, the chamfer is indicated by a note in the output according to International Harvester standards.

III. 4 The Ditto

In some cases, a diameter or a fillet radius may be common to several sections. For convenience in these cases, a "ditto" feature has been included. After the dimension has been given for some section, it can be applied to later sections by giving zero for the nominal dimension and giving the number of the plane where it previously occured for the tolerance. A length dimension ditto has not been provided. Wherever a dimension is referred to by a ditto, it is labelled a typical dimension in the output and the other identical dimensions do not appear.

III. 5 Other Features

Dimensions other than dittoing ones can be suppressed in a drawing by using -99 in place of their tolerance in the input. This feature can be used to produce shaft drawings which incorporate features not provided for in the basic parameters. Although dimensions should not be set equal to zero (left blank) in an attempt to prevent their being dimensioned, they can be made small enough (.0001") so that they virtually disappear in a drawing. (See Appendix III for examples.)

Since fillet radii are often expressed as a maximum only, the program includes a means for doing so. If the tolerance is equal to the respective nominal fillet radius in the input, the total of the two will appear as one figure labelled MAX in the drawing. That is, rather than dimensioning a fillet radius .05 ± .05, the program will dimension it .10 MAX.

III. 6 The Drawing Data

The program provides for any scale to be used in the drawing. Expressed as a decimal, the scale is the (linear) size of the drawn shaft divided by the (linear) size of the actual shaft.

A twelve-character (maximum) name is associated with each shaft. Provision is made for plotting on any width of paper by including paper width as an input parameter.

III. 7 Input Format

For each shaft, the first card contains the drawing data as follows:

Data	Columns	Form
Shaft Name	1 - 12	alphanumeric
Scale	13 - 23	floating point
Paper Width	24 - 31	" "
PLTO	80	T or F

If a plot is desired, PLTO = T. If no plot is to be produced, PLTO = F or is blank. Following the leader card, there is one card for each plane but the leftmost. These are all on the same format:

Data	Columns	Form
I	1 - 2	integer
D(I) (nominal)	5 - 12	floating point
D(I) (tolerance)	15 - 22	" "
IREF(I)	25 - 26	integer
R(I) (nominal)	29 - 37	floating point
R(I) (tolerance)	40 - 48	" "
FR(I) (nominal)	51 - 57	" "
FR(I) (tolerance)	60 - 66	" "
IREFQ(I)	69 - 70	integer
TANG(I)	73 - 77	floating point
LAST	80	integer

LAST is zero except on the last card for a shaft. If there are more shafts to follow, LAST = 1; on the last card of the last shaft, LAST = 2. Blanks can be used for zeroes throughout.

IV DESCRIPTION OF THE PROGRAM

On the macro scale, the program has a series layout, seen in Fig. V. Locations for all dimensions are found and reserved in a series of set-up sections before any plotting is done. Dimensions are divided into four types: angular dimensions, length dimensions, diametral dimensions and dimensions noted with leaders (fillet radii and some chamfers). These four groups represent the best compromise between generality of format and simplicity of programming logic. If after all dimension locations have been planned the computer finds the paper to be too narrow, the plotting is bypassed completely.

A coordinate system for plotting has been adopted with the x-axis on the center line of the shaft and the y-axis on the line of the leftmost plane. This will be called the "data units" coordinate system in accordance with the Benson-Lehner literature. The "plotter"coordinate system has its x-axis at the lower edge of the paper strip and its y-axis at the left hand end. Plotter unit coordinates are plotted as given directly in inches, but those in data units are scaled by the plotter to achieve the desired drawing scale.

The term "above" will be used to refer to the area above the data units x-axis and "below" to refer to the area below it. "Outside" will mean farthest away from the shaft, above or below, and "inside" will mean closest to the shaft above or below but should not be taken to mean inside the outline of the shaft.

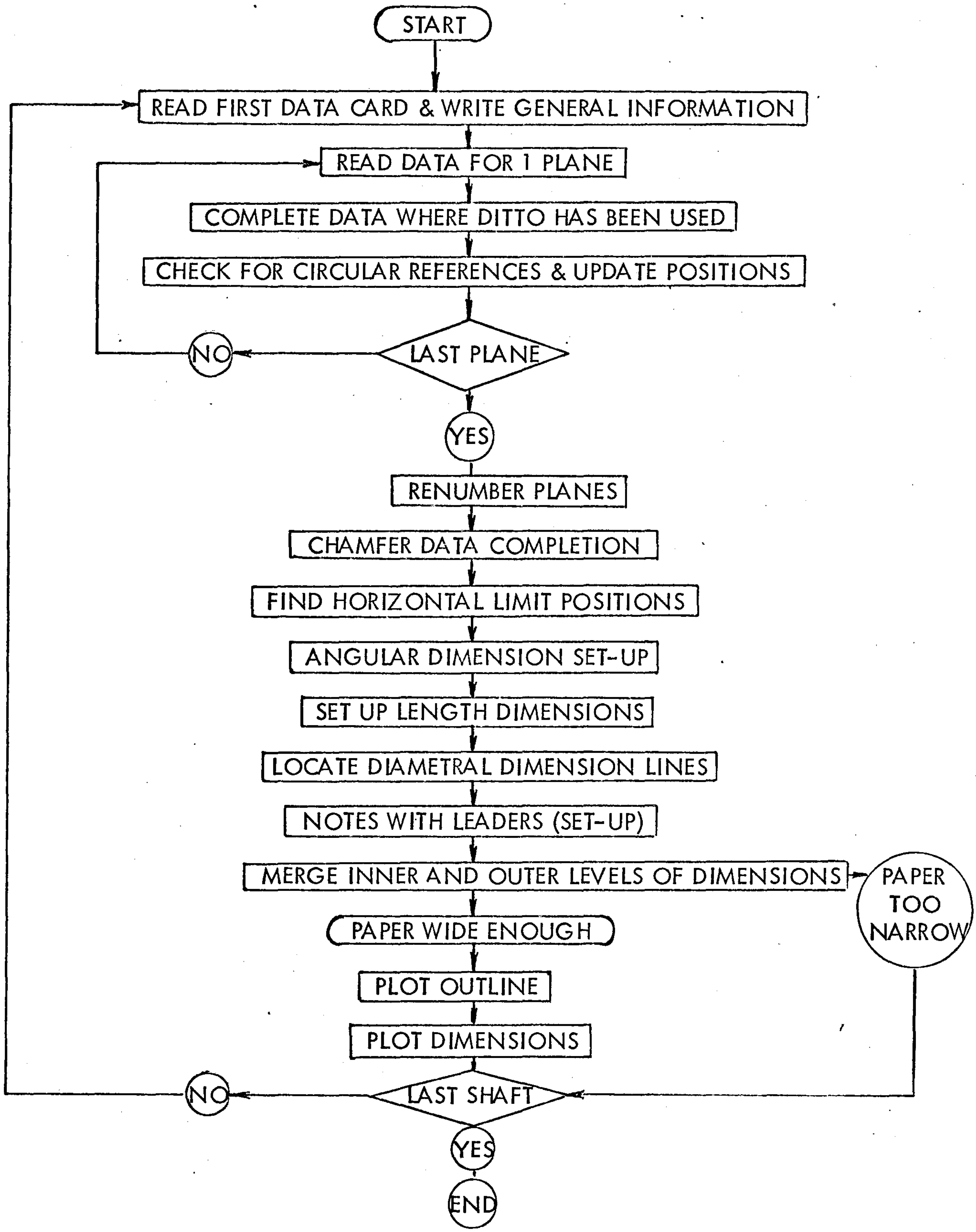


Figure IV Overall Flowchart

IV. 1 Variables Defined

To aid readers of the program listing (Appendix II) and to facilitate further explanation of the program, the program variables are here defined.

DNOM(I) -- the nominal diameter left of plane I

DTOL(I) -- the tolerance on DNOM(I)

RNOM(I) -- the (signed) locating length dimension to plane I

RTOL(I) -- the tolerance on RNOM(I)

FRNOM(I) -- the nominal fillet radius at plane I

FRTOL(I) -- the tolerance on FRNOM(I)

TANG(I) -- the taper angle associated with plane I

A1, A2, A3
A4, A5, A6 -- respectively the vectors in which the above seven input parameters are stored from input to the point in the program at which planes are sorted by their position.
A7

NUM(I) -- the designer's number for the Ith plane read, i.e., the number in the first two columns of the Ith card after the first general data card**

** There are three numbering systems for the planes. The first can be called the designer's numbering system. This is the system of numbers assigned to the planes on the input cards. It may be arbitrary but must be consistent. The second set of numbers arises from the computer's numbering the planes as they are read. These may be called the computer's input numbers. The first plane read is number one in this system, the second, number two, etc. NUM(I) gives the designer's number for a plane as a function of its input number. The third, or working numbers, are assigned by the computer upon the sorting of the planes' positions. The planes are numbered in consecutive ascending order from left to right; the leftmost plane is number one, etc. II(K) gives the working number of a plane as a function of its designer's number.

I ROOT -- vectors used in the sorting process of checking for redundant systems
LEVEL
of length dimensioning

POSIT(I) -- the total distance from plane I to the leftmost plane

IREF(I) -- indicator that RNOM(I) locates plane I with respect to plane IREF(I)

IREFQ(I) -- indicator that a reference dimension is desired between plane I and plane IREFQ(I)

A8 -- vector used temporarily in the renumbering section

DTYP(I) -- true only if DNOM(I) and DTOL(I) are typical dimensions

FRTYP(I) -- true only if FRNOM(I) and FRTOL(I) are typical dimensions

LAB -- logical array used as temporary storage for DTYP and FRTYP before renumbering

LAB(J,I) -- logical record, true if there is a witness line planned at plane I. J = 1 for above the shaft in the drawing or 2 for below it.

II(K) -- a service vector used in sorting. In the renumber section, II relates the designer's number to the final computer's number--II(K)--for each plane. In the set up sections for diameters and lengths, it holds the plane numbers while the respective dimensions associated with the planes are ranked from largest to smallest

JJ -- utility vectors used as temporary storage areas during sort routines in sections
KK
"RENUMBER" and "SET UP LENGTH DIMENSIONS"

PL(I) -- respective left and right horizontal limit positions to the left of plane I
PR(I)

YACL(I) -- the y coordinate (in data units) for the plotting of the angular dimension associated with plane I. YACL(I) = 0.0 if there is no angle to be plotted at plane I.

PABK(J, K) -- records the right hand end of a plotted angular dimension. J = 1 for above, 2 for below. K = 1 is the leftmost entry, all entries being ordered.

WL(J, K, I) -- record of vertical witness lines, including breaks. Recorded in the form of the y value (in data units measured from the outside of the plotting) of the beginning (K is odd) and ending (K is even) of segments of the witness lines. I is the number of the plane and K increases for increasing values of WL.

KCF(J, I) -- the inside-most level of dimensioning already planned at any stage of the program in the corridor to the left of plane I (between I and I-1). J = 1 for above, 2 for below. There is one more column of this vector in use than there are planes for any shaft; the last column applies to the space to the <u>right</u> of the right-most plane.

IH(I, L) -- the level assigned to a length dimension associated with plane I. L = 1 for the toleranced dimension R(I). L=2 if it is a reference dimension. Levels are numbered from the outside inward. Levels below have negative numbers.

LOH(I, L) -- code variable = 1 if length dimension writing is on right hand side or = -1 if on left side. L is as for IH.

LCODE(I, L) -- integer code for the type of arrow format to be used for length dimension associated with plane I. L is as for IH.

DL(I, N) -- record of the placement for writing D(I). N = 1 for the x location of the arrow. N = 2 for a floating point record of the level of writing (by the same convention as IH), or a 0.0 for no diameter to be written. N = 3 is 1.0 for writing to the right of the arrow or -1.0 for writing to the left of the arrow. This array also serves temporarily in "ANGULAR DIMENSION SET-UP".

WLH(K, N) -- a record of the Kth horizontal witness line generated by the set-up. N = 1 for the y coordinate (data units) at which the line is to be plotted, N = 2 for the x coordinate of one end and N = 3 for the x coordinate of the other end (in data units).

ABC(J, I) -- like KCF(J, I) but applying to the inside contour--that of the shaft plus notes and angle dimensions as they are added. A floating point variable (in data units).

DN(I, N) -- record of a note associated with the Ith plane. If N = 4, then DN = 0.0 for no note, 1.0 for a chamfer or 2.0 for a fillet. N = 1 for the x coordinate (signed positive for a right-pointing leader) of the arrow tip and N = 2 for the y coordinate of the arrow tip. If N = 3, DN is the y coordinate of the level of writing. All coordinates are in data units.

WLY(I) -- the radius of a perpendicular section through the shaft at plane I (in data units)

IV. 2 Designer/Machine Translation

The previously described input to the program was designed for maximum convenience to the program user. It is in a condensed form and avoids any redundant statements of information. The data is in an efficient form for expression to the computer, but not efficient for the program's use; and it is therefore restated and expanded before the program uses it. The program reads the data one plane (one card) at a time and makes these modifications to the data before the next plane is read: the angle is converted from degrees to radians for use in all computer library routines; planes other than the first one are examined for dittos, and where dittos have been used, the intended information is inserted; a running account is kept of all plane positions, and a check is made for redundant references.

IV. 2. 1 "COMPLETE DATA WHERE DITTO HAS BEEN USED" Section

This section is skipped after the first plane data has been read, for it should contain no dittos. No test has been provided to be sure that dittos have not been used on the first card, but one's presence would simply be interpreted as bona fide data throughout the program, causing no real harm but making itself evident in the plot.

It is because this section is in the input loop that all dittos must refer to planes read previously. The section was placed there in spite of this disadvantage (see discussion in Suggestions) to limit the number of planes to be checked for a number matching the ditto number. The checking must be done indirectly because the designer's numbers are different from the computer's input numbers.

The diameter or fillet radius is considered to be dittoed if the nominal dimension ≤ 0.0 and the tolerance $\neq 0.0$. The tolerance is rounded off to the nearest integer and a search is made for an earlier plane bearing the designer's number equal to this rounded off value. An error message is written if the search finds none, and the program stops without further execution. If the search is successful, the nominal dimension is set equal to the intended value, and A2 is set equal to -99 because this dimension will not be plotted. Only the nominal dimension is used for drawing the shaft. A position in LAB is switched to "true" for the dimension copied, indicating that this is a typical dimension and will be plotted as such.

IV. 2. 2 "CHECK FOR CIRCULAR REFERENCES AND UPDATE POSITION" Section

A valid system of n-1 dimensions locating n parallel planes with respect to one another can have no redundancies. A redundant dimension is wasted, leaving only n-2 effective dimensions, and at least one plane cannot be located with respect to the others.

In searching for the distinguishing features of a valid dimensioning system, it was discovered that any valid system can be thought of as a tree when the dimensions have the one-way sense that they do in this program. If plane I has IREF(I) = K, then R(I) locates plane I with respect to plane K; and in the "tree" sense, I must be thought of as a "branch" emanating from K. Then I is one level "higher" than K; and if K is not referred to anything else (as is the case if K = 0), then K can be said to be the "root" of I, and I is on the first level of "K's tree." If K is still at the zero level and some other plane is referred to I, then that other plane is in the second level of K's tree. The tree always branches upward because although a plane can refer to only one other plane, it can have any number of planes referring to itself.

Thus, any system of valid dimensioning can be graphically visualized as a tree and can be assembled one reference at a time. In the assembling process, each plane number is added to a tree as a new branch when that plane's data are read in. There may be several trees growing at once. If, for example, the third plane read is referred to the fifth plane then a tree is started with the fifth plane as its root, while the second plane might be rooted in the first. If later the fifth plane were referred to the second, it would be "moved" (with the third plane and any other branches) into the tree rooted in the first plane.

The reason for using this tree system is that improper dimension relationships become apparent as soon as they arise. If the root of a developing tree is referred to a plane which is already related to it (in its tree), a circular or redundant referral is present and the program execution is terminated after an error message is written. For example, if B is referred to C which is referred to B, a circular reference has been made.

One other type of error can be detected after all the planes have been read. At this point, there must be only one tree (with plane zero at its root). Any other tree will have an undefined plane at its root, and this is cause for program execution to be terminated with the writing of the proper error message.

After the last plane has been read, NPLANE is set equal to the total number of planes (recall that this is the number of planes read plus one) and three other stages of data alteration occur--the "RENUMBER", "CHAMFER DATA COMPLETION" and "FIND HORIZONTAL LIMIT POSITIONS" sections.

IV. 2. 3 "RENUMBER" Section

In this section, the planes are given their working numbers and all data are transformed to the new variables using the new subscripts. The number "1" is reserved for the leftmost plane, although it has no input data associated with it, for convenience in all later routines. For example, since no zero subscript is allowed in FORTRAN, the only way for the position of the leftmost plane to be indicated in the POSIT vector is to number it 1, making POSIT(1) = 0.0.

Going into "RENUMBER", POSIT for all planes has a negative sign affixed. The rightmost plane is found by searching for the algebraically least value of POSIT. After the data have been transferred for this plane, then its position can be set positive, putting it out of contention in the succeeding searches. In the case of a chamfer to be dimensioned by its shoulder diameter, the values of POSIT for the chamfer plane and shoulder plane will be the same at this point because R of the chamfer plane is zero. For that reason, they are sorted by their values of TANG. The shoulder plane has TANG = 0 and the chamfer plane has TANG$>$0 if it is on the right of the shoulder plane or TANG$<$0 if it is on the left. Thus, the plane having the algebraically larger value of TANG is always on the right. Note that a chamfer plane must be referred to its shoulder plane if it is to have a shoulder diameter dimension.

In this section, the values recorded for IREFQ are converted to their working form. As mentioned earlier, wherever no reference dimension is intended, IREFQ is left blank on the input card. This results in zeroes in memory except when a reference dimension is desired. At this stage of the program, a new convention is adopted: wherever a reference dimension is desired, IREFQ of one plane is set equal to the working number of the other plane involved. This can now include references to the leftmost plane, formerly indicated by -1's to distinguish them from blanks. Where no reference dimension is required, IREFQ(I) is set equal to I, the plane's own number. This convention allows a uniform treatment of all length dimensions in the "SET UP LENGTH DIMENSIONS" section.

IV. 2. 4 "CHAMFER DATA COMPLETION" Section

This section operates only on parameters associated with chamfer planes, as defined earlier. If the rightmost plane is a chamfer plane, an error message is printed and the program is stopped. If RNOM$>$0, DNOM (the shoulder diameter) is calculated and DTOL is set equal to -99.0, signalling that it will not be dimensioned in the drawing. This calculation avoids using RNOM because occasionally this might not be the distance between the chamfer and shoulder planes. No provision is made for a shoulder diameter which is calculated to be smaller than the diameter adjoining the shoulder plane, as this is a non-fatal error.

If RNOM = 0, this signals that the chamfer is to be dimensioned by the shoulder diameter. A test is provided to be sure that the shoulder diameter given is not larger than the diameter adjacent to the chamfer plane because this is a serious error. The location of the chamfer plane and its POSIT is corrected. Its RTOL is set equal to -99.0 to suppress the drawing of the dimension. Finally, this section corrects the right shoulder anomaly seen in Fig. III by switching the diameters of the chamfer and shoulder planes. The switching is done for consistency in the plotting phase of the program.

IV. 2. 5 "FIND HORIZONTAL LIMIT POSITIONS" Section

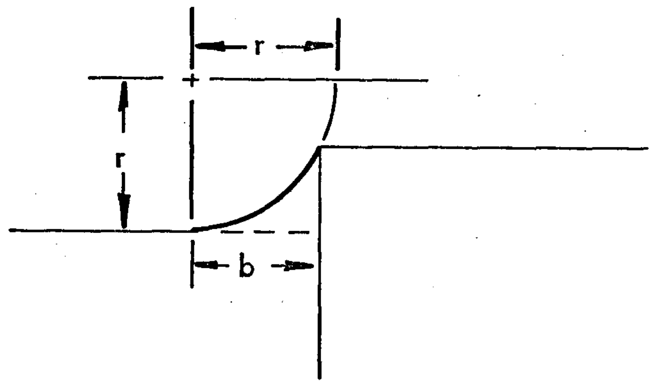
Between any two planes, except a chamfer plane and its shoulder plane, some section of the shaft profile must be horizontal. This section may be so short that it is only a point, but it is within the limits of this horizontal section that the diameter exists, and this is where it must be indicated. In a section where there are no fillets or tapers, the horizontal limits are at the planes which define the section. When a taper or fillet is at one or both ends, the length of the horizontal section is reduced. The geometry used to calculate the amount of reduction is shown in Fig. VI.

Because PL and PR are used in drawing the shaft outline, as well as locating diametral dimensions, they are given functionally consistent values for chamfer sections, even though no horizontal is involved.

In this section of the program, two checks are made on the consistency of the input data. Where a taper and fillet exist together, the program checks that the taper angle is positive when the larger diameter is on the right and negative when it is on the left. It also checks that the calculated horizontal section does not have a negative length. The calculations could erroneously imply a negative length in certain cases of inconsistent input--for instance, a fillet diameter between two planes which is larger than the distance between them but less than the difference in their diameters.

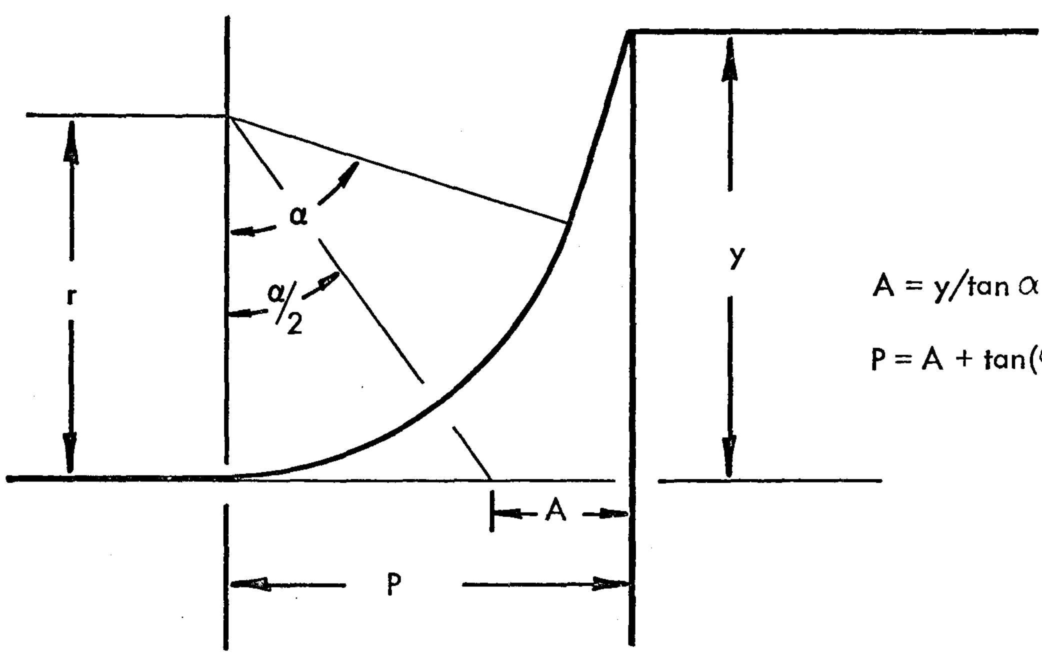
IV. 3 Dimension Set-Ups

The largest part of the program is in the set-up sections for the four types of dimensions. The sections are ordered according to the priority of dimension placement. Angular dimensions were given top priority because for any angle there are only two places that the dimensioning indication can be placed--above or below the shaft, against its outline. They compete for space with all the other dimensions and interferences between angular dimensions and others were judged to be the worst type. Placing them first gives the best chance for



$$a^2 + b^2 = r^2$$

$$b = \sqrt{r^2 - a^2}$$



$$A = y/\tan \alpha$$

$$P = A + \tan(\alpha/2) * r$$

Figure V   Horizontal Limits Geometry

finding them an open location and allowing the later dimensions to avoid them. Length dimensions are placed second because their witness lines have only the above-below option for placement. Furthermore, length dimensions are given outside placement to minimize interference, and the system for outside reservation of space works from the outside to the inside. Arbitrarily, diameters were placed before notes. The placement of either was not determined to avoid the other and the reversal of their order in the program should make no difference in the resulting plot.

IV. 3. 1 "ANGULAR DIMENSION SET-UP" Section

ABC keeps a record of the limits of the space occupied at the center of the drawing (see Fig. IX). As angles and notes are planned, they are added just outside the limits and ABC is adjusted to record that their space is reserved. Before any dimensions are planned, ABC receives its initial values which record the shape of the shaft. The drawing is divided into "corridors" by the planes of the shaft. ABC(L,I) applies to the Ith corridor between plane I-1 and plane I. The upper limit of occupied space is designated by L = 1 and the lower limit by L = 2. This system is sensitive only to the outside-most plotted point in any corridor. Thus, if the shaft has a chamfer between planes four and five where the diameter changes from 2.5 to 2.7 inches, ABC(1,5) and ABC(2,5) must be equal to 2.7.

The array which records occupied space can be called a grid system. Its sensitivity to details of the contour of the occupied space is a function of the fineness of its mesh. ABC could be more sensitive if the drawing area were divided into more corridors; but dividing the area by planes was consistent with the parameterization, and the most convenient compromise between insensitivity and complexity. In the vertical direction, it was decided that ABC must be a floating point variable to best record the shaft profile.

An attempt is made to dimension all angles except those associated with a noted chamfer. A chamfer is indicated by a note if: 1) the angle is 45° 2) the distance between the chamfer plane and the shoulder plane is less than 1/2" 3) the chamfer plane is located with respect to its neighboring plane 4) the chamfer is not to be dimensioned by the shoulder diameter. The first condition is given by International Harvester standards; the second and third are intended to eliminate a 45° taper which is not to be treated as a chamfer.

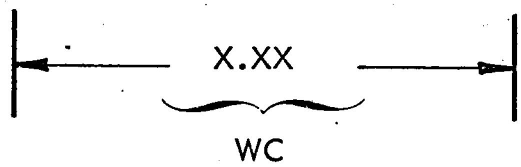
The dimensioning for an angle is allocated an area .8" x .8" (plotter units). An angle is not dimensioned if this area interferes with the outline of the shaft, recorded in ABC; but a message is printed out to say that the dimension is being omitted. Angles are dealt with from left to right. An angle dimension is placed above the shaft unless it interferes with a previously placed dimension. In that case, it is placed below unless there is interference there too, causing it to be omitted and a suitable message printed.

The PABK array, which is zeroed for each new shaft, is the means of checking for interferences between angle dimensions. Because the angles are handled in a left to right order and a new entry is always inserted in the first zero spot in PABK, all entries in PABK are in order, left to right. A new entry is not made in PABK if it would be within .8" of the previous entry, for this proximity indicates interference. It was envisioned that in one case this test would be too severe. Because the y-coordinate of the angle dimension is not recorded in PABK, no simple check exists to ascertain that the y-coordinates of two supposedly interfering angle fields are not separated by more than .8". If that is the case, one angle dimension would be sufficiently above the other so that no interference would occur. However, this was judged to be a rare occurence.

DL is used to temporarily store the modifications to be made to ABC; if ABC were updated before all angles had been located, it would be difficult or impossible to distinguish an outline interference from an interference with another angle.

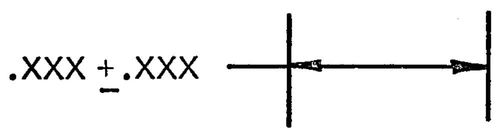
IV. 3. 2 "SET UP LENGTH DIMENSIONS" Section

Four different formats are provided for the arrangement of arrows and length dimension writing (see Fig. VI). Type 1, which has arrows and writing between the witness lines, is used unless the length between witness lines (in inches) is less than WC + S4L, where WC is the calculated length of the writing and S4L is the minimum allowable length for the two internal arrows. Type 2, with arrows outside the witness lines and writing inside, is used unless too little room exists for the writing. Type 3 is used if the length is too short for the

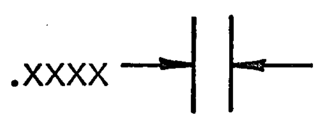


Type 1

Type 2



Type 3



Type 4

Figure VI Length Dimension Formats

writing but is greater than S5L, the minimum limit for the length of a single internal arrow. Type 4 is used in all remaining cases.

The highest priority in placing length dimensions is avoiding interference with the angular dimensions, which have their locations planned beforehand. If an angular dimension is placed in line with a plane, that plane is considered "blocked above" if the angular dimension is planned above it, or "blocked below" if it is planned below. The placement of a length dimension between two planes is considered impossible if at least one of them is blocked above and at least one of them is blocked below. When the program encounters such a case, it omits the dimension, printing a notice that it has done so.

The second priority in locating a length dimension is to avoid interference between the dimension and those previously placed. These interferences take the form of intersections between the vertical witness lines and the horizontal writing and arrows. It was found that these intersections can be minimized, but not avoided completely, by placing on the outside those length dimensions which span the most planes. Therefore, the program sorts all the locating and reference length dimensions with respect to the number of planes they span, so that the largest spans can be placed first. Any length dimension may be placed above or below to minimize the number of interferences, providing that neither plane concerned is blocked.

The third criterion for locating a length dimension is minimizing the number of new witness lines which must be drawn. Minimizing the number of witness lines gives the least chance of interfering with later dimensions.

The fourth and lowest criterion in dimension placement is that of space. Because the space for dimensions is reserved from the outside inward in discrete levels, each new dimension is placed at the outside-most open level. This criterion favors the alternative location which is farthest outside.

Type 3 and type 4 length dimension formats are asymetrical, with the possibility of placing the writing to the right or the left of the witness lines. Therefore, to simplify programming, all length dimensions are treated as if they have four possible locations open to them: above-right, above-left, below-right and below-left. For each length dimension, all four possible locations are compared with respect to all four criteria described above. The location decision is made, based on the highest possible criterion. If two or more options are equal with respect to all four criteria, of those remaining the program takes first the topmost and then the leftmost.

A separate section of logic deals with type 1 length dimensions between neighboring planes. This logic provides a shortcut in program execution, bypassing the much more lengthy generalized logic which deals with all other cases. A type 1 length dimension is so simple (because there can be no interferences) and common that providing the logic needed to treat it separately was justified by the increase in program efficiency.

IV. 3. 3 "LOCATE DIAMETRAL DIMENSION LINES" Section

A single format was selected for diametral dimensioning to make all writing horizontal, as prescribed by International Harvester standards, and to avoid writing within the outline. Examples of the format may be seen in Fig. VII. This format alone is not suitable for all situations. If a section is too narrow, it was decided that horizontal witness lines must be used to keep from crowding the arrow. For a chamfer dimensioned by its shoulder diameter, horizontal witness lines are always used. If a diameter is too small, the two arrowheads overlap, but the limit for this condition is about .30" diameter (see ARROW Subroutines). No provision was made for avoiding the overlap because the condition is rare and results only in a minor flaw in the drawing.

At most, four possible locations are considered for each diameter which is to be dimensioned. (Those which are given in the input with dittos or with -99.0 for the tolerance and those associated with regular chamfers are not dimensioned.) The four locations are above with leader turned to the right, above left, below right and below left.

It was decided to provide two possible locations for the diameter arrow--giving a greater chance of avoiding interference with angular dimensions--rather than consistently placing the arrow midway between the horizontal limit positions of a section. The left-hand arrow position is always associated with the right-hand-turning leader and the right-hand arrow position with the

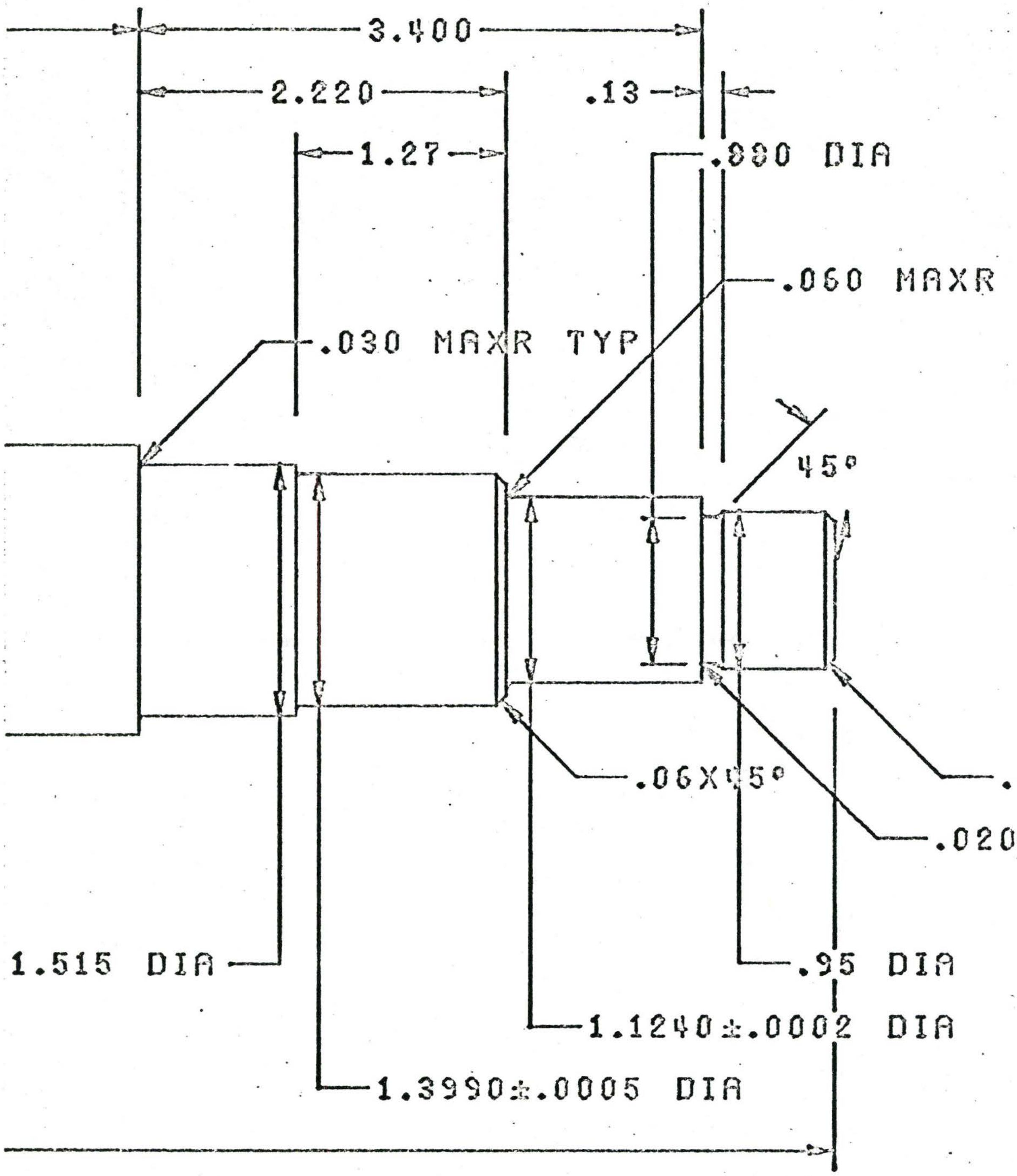


Figure VII Product of Corrected Diametral Dimension Logic left-turning leader to reduce the number of interferences between diameter writing and vertical witness lines.

With the exception of minimizing vertical witness lines, the previously described criteria for locating length dimensions are used for locating diametral dimensions. They are as follows: 1) avoidance of interference with angle dimensions 2) avoidance of interference with vertical witness lines 3) maximization of space. While these criteria work very well for length dimensions, they proved to be incomplete for diametral dimensions for several reasons.

Diametral dimensions should also avoid interferences among themselves. These interferences are more serious than those between length dimensions because no provision has been made for breaks in diametral leaders. It had been decided that these interferences were too difficult to detect for the purpose of providing leader line breaks because the x-location of the leader lines were less readily accessible than those of the planes. The interference problem was intensified because a poor choice had been made of the order in which to locate diametral dimensions. The original choice had been to place the dimensions for largest diameters first, thinking that because these were on the outside, dimensions were less likely to be crowded over the tallest sections. The results of this original dimension placement logic, an example of which can be seen in Fig. VIII, were considered to be unsatisfactory.

The program was written with the idea that it be only a first attempt at computerizing the drafting of turned shafts. Logical shortcomings are therefore

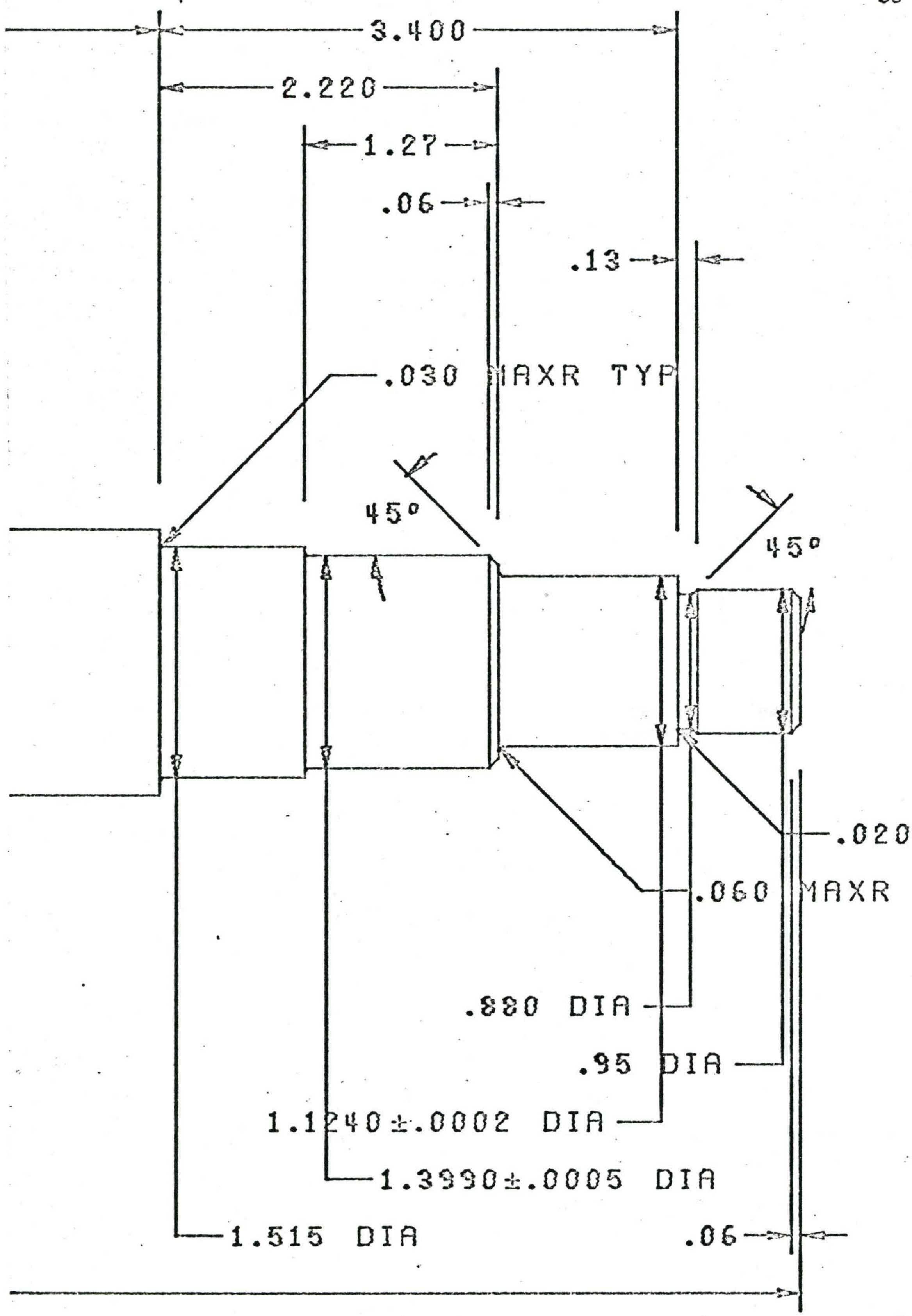


Figure VIII Product of Original Diametral Dimension Logic pointed out but have not been corrected; this one is the exception. It was decided that the output of the program could be vastly improved by making two logical changes. The first diameters to be located are those at the center of the shaft; thus their writing is outside that of the other diametral dimensions. After the first diametral dimension is placed, all those to the right of it are given right-hand-turning leaders, and those to the left of it are given left-hand-turning leaders. Without the program's either counting interferences or providing leader breaks for them, the revised logic gives results much improved over those of the earlier logic (see Fig. VII).

The rest of the logical apparatus has been left intact, but chooses between only two alternatives except in the case of the first diametral dimension or ones for which the writing of all four alternatives falls between the planes bounding the section.

Logic is provided to decide whether horizontal witness lines should lie on the left or right of the diameter which they indicate. They are placed on the left of a left chamfer or on the right of a right chamfer unless there is interference with angular dimensions both above and below. For a groove or ridge, they are placed on the side which has fewer options "blocked" by angular dimensions. If both sides are equal in that regard, they are placed on the side which has the longer adjoining section. The placement of horizontal witness lines modifies the ABC array so that notes with leaders may avoid interfering with them.

Any diametral dimension which has all options blocked is omitted and a suitable message printed.

IV. 3. 4 "SET UP NOTES WITH LEADERS" Section

There is at most one note for each plane, either a fillet radius or a noted chamfer. Only one format is used, a 45° leader line with an arrowhead at the end which contacts the shaft outline and a short horizontal segment at the other end. Locations for all the arrow points are determined without distinguishing between the top and bottom of the shaft. Fillet leaders contact the 45° point of fillets where the arc is at least 45°. Chamfer leaders are directed to the mid-point of the chamfer. Right-pointing leaders are used for left chamfers and fillets when the smaller diameter is on the left, to minimize the chance of the leader line intersecting the outline. Writing locations are reserved for notes in distinct levels, inner levels first, adjusting ABC to record their presence as the reserving process proceeds. Right-pointing leaders are handled in a left-to-right order so that a later one with writing outside of earlier ones will not cut the writing of earlier ones with its leader. For the same reason and at the same time, left-pointing leaders are reserved locations in a right-to-left order. Each note has its writing placed at the closest possible level to the shaft, be it above or below. The location decision is based on a survey of ABC, insuring that notes will not be written on top of one another and that notes will not be written with .4" vertically of the shaft outline.

No breaks are made in vertical witness lines for notes because the necessary logic was not justified at this stage of development. The complication arises because the witness line breaks are recorded in a different frame of reference than the notes. The relationships between the shaft's frame of reference (the data axes) and the outside frames of reference, in which length and diameter dimensions are located, is not set until the "MERGE" Section.

IV. 3. 5 "MERGE" Section

In this section the distances between four horizontal reference lines are determined in order to bring the outside dimensions (lengths and diameters) as close to the shaft as possible and to fit the shaft and all dimensioning within the height specified by PWIDTH. The four horizontal reference lines are:
1) the x-axis of the plotter coordinates, fixed at the bottom of the paper
2) the x-axis of the data coordinates, upon which the center line of the plotted shaft falls 3) the upper baseline at the upper edge of dimensioning from which all upper diametral and length dimensions are located 4) the lower base line at the lower edge of all dimensioning from which all lower diametral and length dimensions are located. (See Fig. IX.)

The levels of dimensioning recorded in KCF are allotted .375" each, exceeding the International Harvester minimum of .25" between levels of dimension writing. Between planes I-1 and I, ABC(1,I) and KCF(1,I) dictate some minimum distance between the data units x-axis and the upper base line which will maintain the .4" minimum approach of outer dimensioning to the

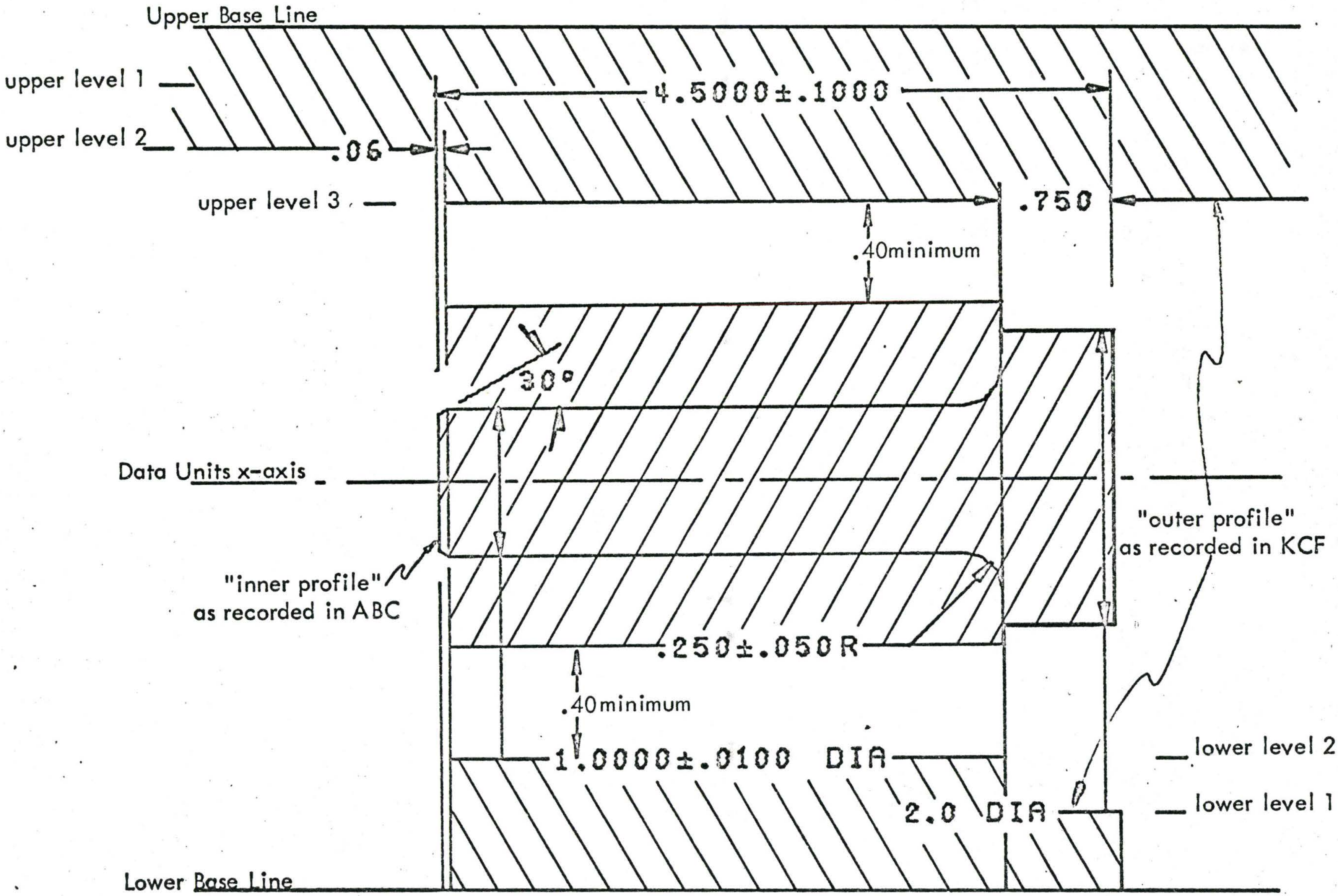


Figure IX The "MERGE" Process

shaft outline. Taking the maximum of such figures calculated for all NPLANE + 1 corridors gives the distance between the upper base line and the data units x-axis. If this distance is greater than PWIDTH/2, the data units x-axis is placed so that the distance from the plotter units x-axis to the upper base line is PWIDTH.

In the same way, the distance can be calculated between the data units x-axis and the lower base line. After that calculation is made, if the distance from the lower base line to the upper base line is more than PWIDTH, the plotting is deleted and a message is printed telling how much too narrow the paper is. If the distance from the data units x-axis to the lower base line is more than PWIDTH/2, then the lower base line is made to coincide with the plotter units x-axis. If the location of the data units x-axis is not determined by either of the distances of base lines being greater than PWIDTH/2, then it is located so that the distance between it and the plotter units x-axis is PWIDTH/2.

An alternative to moving the shaft up or down was considered as a means of enabling the program to fit drawing and dimensioning on the paper-- allowing a variable distance between levels of dimensioning. This plan has two major disadvantages. The first is that in planning notes on discrete levels, the distance between the levels must be known in advance. Notes must be written on discrete levels to maintain the minimum vertical spacing between notes. The second is that the witness line array, WL, records vertical distances to the beginning and ending points of gaps, based on a set distance between levels of dimensioning. If the values of WL were scaled to adjust for changes in the distance between levels, the size of each gap would be distorted.

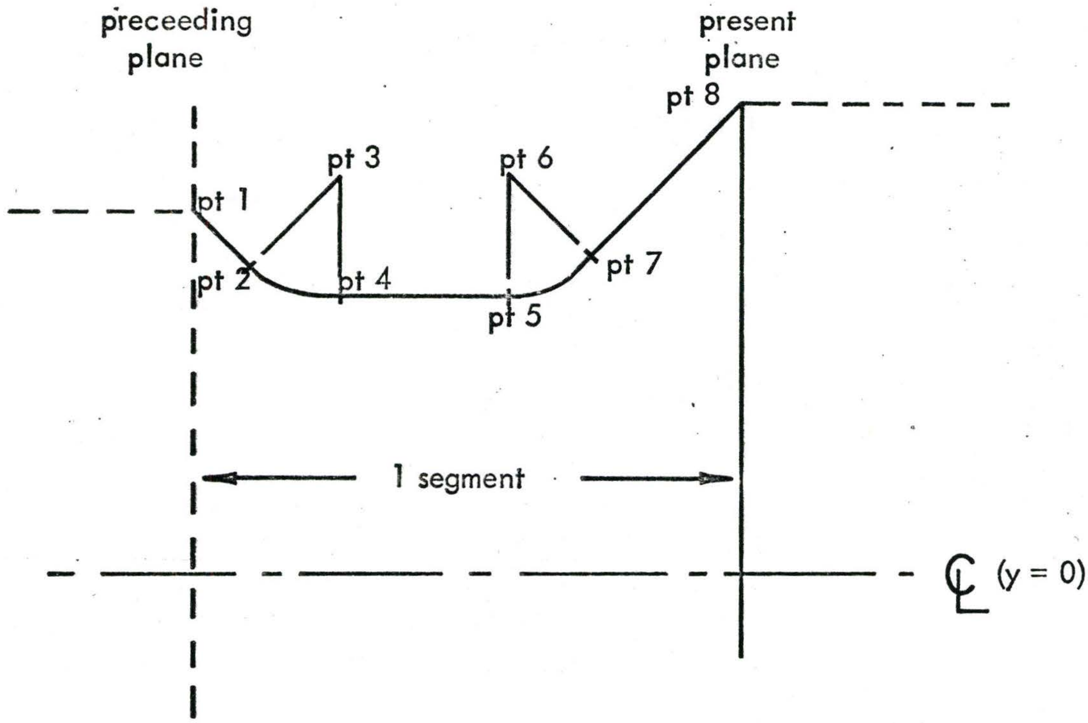
IV 4. 0 Plotting

Although a major part of the program is outside the plotting section, this segment uses about 80% of the execution time. The Benson-Lehner subroutines for plotting are probably comparatively time-consuming.

IV 4. 1 Plotting the Shaft Outline

The shaft outline is plotted one section at a time, left to right. A set of 16 parameters was conceived which describes any section recognized by this program. Fig. X illustrates their geometrical meaning. Only the top of one segment is shown in Fig. X; but because the x-axis is the center line, the same coordinates describe the bottom of each segment when the y-coordinates are made negative.

Few segments have all the allowed features and need all parameters to describe them. For simplicity there is only one set of plotting statements to plot all segments, and codes are used to omit plotting of any feature not present in a particular segment. The line from point 1 to point 2 is omitted if Y1 is set equal to zero. The arc from point 2 to point 4 is omitted if Y3 is set equal to zero. The line from point 4 to point 5 is not drawn unless X5 > X4. The arc from point 5 to point 7 is omitted if Y6 is set equal to zero. Y8 cannot be set equal to zero because point 8 is used for drawing the vertical line at the



X1, Y1 -- coordinates of point 1		
X2, Y2 -- coordinates of point 2		
X3, Y3 -- coordinates of point 3		
X4, Y4 -- coordinates of point 4		
X5, Y5 -- coordinates of point 5	}	coordinates in data units
X6, Y6 -- coordinates of point 6		
X7, Y7 -- coordinates of point 7		
X8, Y8 -- coordinates of point 8		

Figure X Outline Plotting Parameters

preset plane, but the line from point 7 to point 8 is not drawn if their x-coordinates are the same.

Chamfer segments are drawn as lines between point 4 and point 5, omitting all other sections. They are dealt with by separate logic (see statements 5100 to 5150 in the program listing) because their case is very simple.

IV. 4. 2 Plotting the Dimensions

The plotting of all dimensions is very simple, involving the repetitive use of plotting subroutines described in Appendix I. The main program logic has only to separate the different formats for each type of dimension and supply the correct arguments to the subroutines.

V ECONOMIC FEASIBILITY

Appendix III shows several shafts plotted at McMaster by the program. Each example gives a price, which is the cost of program execution on the IBM 7040 at $4.51/minute. This charge for the computer includes all overhead plus amortization of the machine. Further, the average shaft takes about five minutes to plot on the Benson-Lehner plotter at $3.00/hour for a cost of $.25. The cost of punching the input cards is less than one cent per card. The largest cost factor at present, the cost of loading the program into the computer from cards, is $9.90 due to the two minutes, twenty seconds required. At present, the program must be loaded each time it is run; and although the effective cost can be reduced by running several shafts at one time, this is not always practical. This problem can be virtually eliminated by storing the program on tape or on disc. A loading from disc requires only three seconds.

The same shafts which appear in the Appendix would require on the average of at least two hours if done by a draftsman. In a design office, drafting time costs about $7.00/hour, resulting in a cost of $14.00 per shaft.

Comparing the cost of the two methods--manual drafting and computer drafting--the previous figures indicate that the computer can do the job for 1/4 the cost.

VI CONCLUSIONS

The sample shaft drawings in Appendix III give ample evidence of the high degree to which this program has achieved the original aims of the thesis, "to test and verify the underlying logic." The drawings produced by the logic provided the basis for one improvement which has already been made--that in the logic for locating diametral dimensions. The program and drawings have also been the source of the many other improvements suggested in Section V.

The most important question that has been answered is whether computerized drafting of power train shafts is economically feasible at the present time. When the first attempted program is able to produce the basic drawing for an average shaft for about $3.50, the answer is certainly "yes." Further work on the program will make it still more economical; and the advantages will multiply as it is linked to other programs to form a design system. In addition to direct economic advantages, less tangible ones exist such as shortened time from design conception to finished drawing, more readable drawings, more uniform adherence to standards and time saving for the designer. These advantages are present with the drafting program, and will multiply even faster than the tangible advantages as a total system is developed.

One question must now be raised for consideration by both International Harvester and McMaster: what to do next? Whether to expand the capabilities of the drafting program to include such things as enlargements, sections, gears, splines and keyways; or to begin adding other sections of a design system such as CRT input, strength and fatigue analysis, programs and output for numerically-controlled machines? This question would bear some study, and the answers depend on what facilities are available and what types of shafts are being dealt with.

No matter how general a program is, it can deal with only a limited number of cases; and man must decide where to draw those limits to divide the work which he can best do from that which can best be done by the computer.

VII RECOMMENDED CHANGES AND ADDITIONS

The complete review of the program necessary to write the preceeding sections has suggested several ways to improve the program with minor changes and several expansions of the logic which could improve the results and usefulness of the program.

At present, a plane can ditto a dimension only from a card which is read earlier by the program (which preceeds it in the input). This limits the ease with which planes can be added to or removed from a shaft, and prevents the designer from planning a distribution of dimensioning which will minimize crowding. For instance, if two sections share the same diameter and the same fillet radius, the designer cannot make the diameter dimension appear on one section and the fillet dimension on the other.

If the ditto completion section (statements 100 to 199) were modified slightly so that it could be moved outside the input loop (to about statement 400), then planes could bear dimensions dittoed from planes read either before or after them. This constitutes the first suggestion for improvement.

The second suggested improvement would be a revision in the size and proportions of the dimensions of array WL and WLH to better correspond to the 49 plane capacity of the program. The present capacity of WL for 8 breaks in every vertical witness line and the capacity in WLH for 30 horizontal witness lines seems disproportionately large. At present, the 11,300 words used by the program are only 63% of the storage capacity of the 32K McMaster machine, considering the 10,200 words always occupied by the system software. If the program is to be expanded, however, storage space may be at a premium; and perhaps even the 49 plane capacity of the program should be reduced to make room for additional logic.

Several improvement could be made in the "LOCATE DIAMETRAL DIMENSION LINES" section. It is poorly put together at present because the original concept did not work out as planned; modifications were made to repair its most obvious weakness, while retaining the original logical framework. An entirely new replacement for this section could be written which would be simpler, execute faster and produce better results. The present logic which considers two x locations for the arrows should be eliminated in favor of one which finds one x location by considering locations of angular dimensions. Then in each case four locations could be considered for the writing, taking into account not only space and interferences with length dimensions (the two things now considered) but also interference with diametral dimensions. In order to test for interference with diametral dimensions, a new array should be established containing the x locations of all vertical lines in order--both witness lines and leader lines.

A suggested addition would be a subroutine to determine the number of decimal places intended for a nominal dimension given as input. This program would be written in assembly language because it is impossible for FORTRAN to distinguish between zeroes and blanks when a card is read. It would supply the same code for the number of significant figures as is now used, but would relieve the designer of the bother of a code and allow him to show his intentions directly. The flowchart for such a subroutine is shown in Fig. XI.

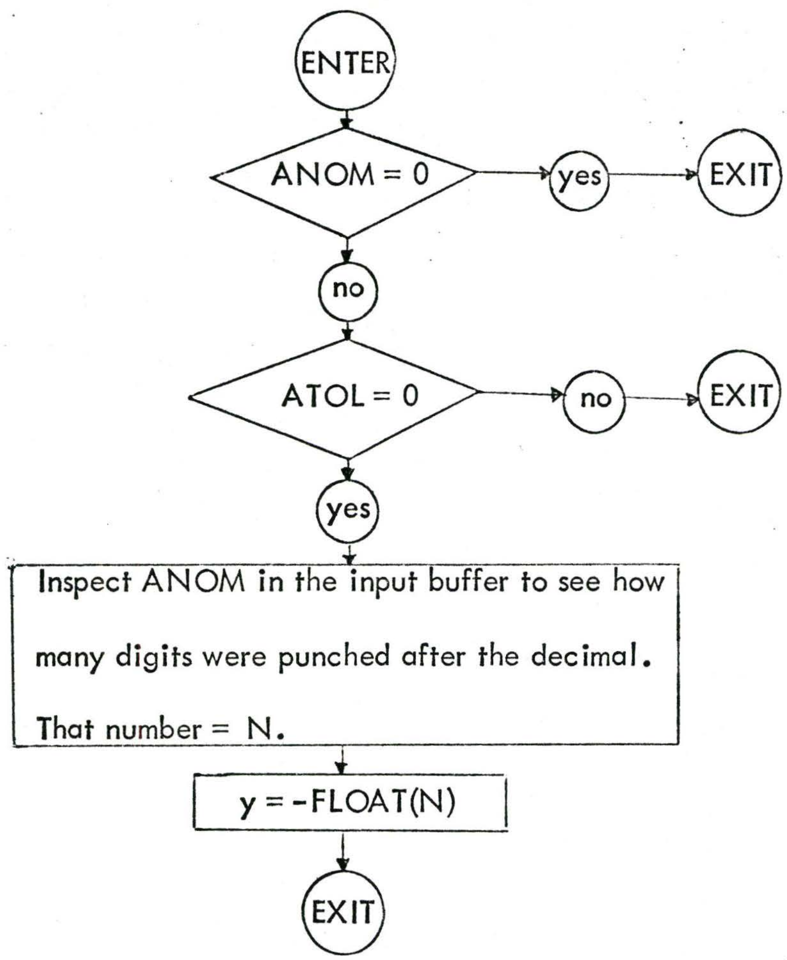


Figure XI   Subroutine Inspect (ANOM, ATOL)

The program could be greatly improved by adding a skip capability for diametral leader lines. This addition would require a logical apparatus, not now present, similar to that which provides breaks in length witness lines. At the same time, provision should be made to allow breaks in both types of vertical lines for the writing of notes.

A final suggested addition would be more sophisticated logic for locating notes with leaders. The decision to place a note above or below the shaft should be based primarily on avoidance of interference between the leader and angular dimensions, and between the leader and horizontal witness lines. The present logic which uses the criterion of closeness to the shaft could be retained in a secondary role.

VIII REFERENCES

1. International Harvester Proposed Drafting Standards

2. International Harvester Standards for Lines and Line Work

3. Bensen Lehner Plotter Instructions

APPENDIX I

DESCRIPTION OF SUBPROGRAMS

APPENDIX I DESCRIPTION OF SUBPROGRAMS

Ten subprograms were written in conjunction with the main program to perform the most repetitive tasks. Most of them are plotting subroutines which plot certain features several times and at several locations on the page for each shaft.

I. 1 Function CLEN (ANOM, ATOL, I, J, TYP) and Subroutine ALLPLT (X, Y, ABOM, ATOL, I, J, TYP, XF)

CLEN gives the length (in inches) of a dimension written out by subroutine ALLPLT, using the same arguments. The Benson-Lehner writing subroutines used by ALLPLT produce writing the length of which is difficult to predict. Benson-Lehner allows for different sizes of writing; the programmer chooses one by giving a scale factor. The difficulty arises not only because different letters and characters have different widths, but also because as the letter size is changed by scaling, the width of the spaces between letters does not change proportionately.

Lettering height is .12" for all dimensions, in accordance with International Harvester standards. This corresponds to the Benson-Lehner scale factor of 2 and yields a space between characters of .08". The letters written in dimensions by ALLPLT are also .08" wide and the numbers are .06" wide except for the number 1, but no allowance is made for that exception in CLEN.

CLEN works by adding up "units" for each of five possible parts of the dimension (see Fig. XII). A "unit" has a width equal to .01" x the scale factor. This is true for any Benson-Lehner lettering, but in this case means that a unit is .02" wide.

If a tolerance is written out, note that it is always written with four figures to the right of the decimal.

I. 2 Subroutine SKIP (JAB, I, K, GAP)

When a new witness line is first planned at plane I, WL(JAB, 1, I) is set equal to the distance below the upper baseline (if JAB = 1, or above the lower baseline if JAB = 2) at which the witness line begins. If no breaks in the witness line have been called for by the time the set-up is complete, it is plotted from this beginning point to its end point, a fixed distance from the shaft. If, however, a dimension on level K interferes with the witness line, subroutine SKIP is used. This subroutine provides a break at level K in the vertical witness line at plane I by introducing new intermediate points. The width of the break is two times GAP. The plotting of the witness line then proceeds from the beginning point to the first intermediate point with the pen down, to the second with the pen up and to the end point with the pen down. This works for any number of breaks in a vertical witness line, as long as 1) each call to SKIP has a higher value of K than the last one for that plane and 2) the gaps do not overlap with one another. Both of these conditions are satisfied by the present logic of the main program.

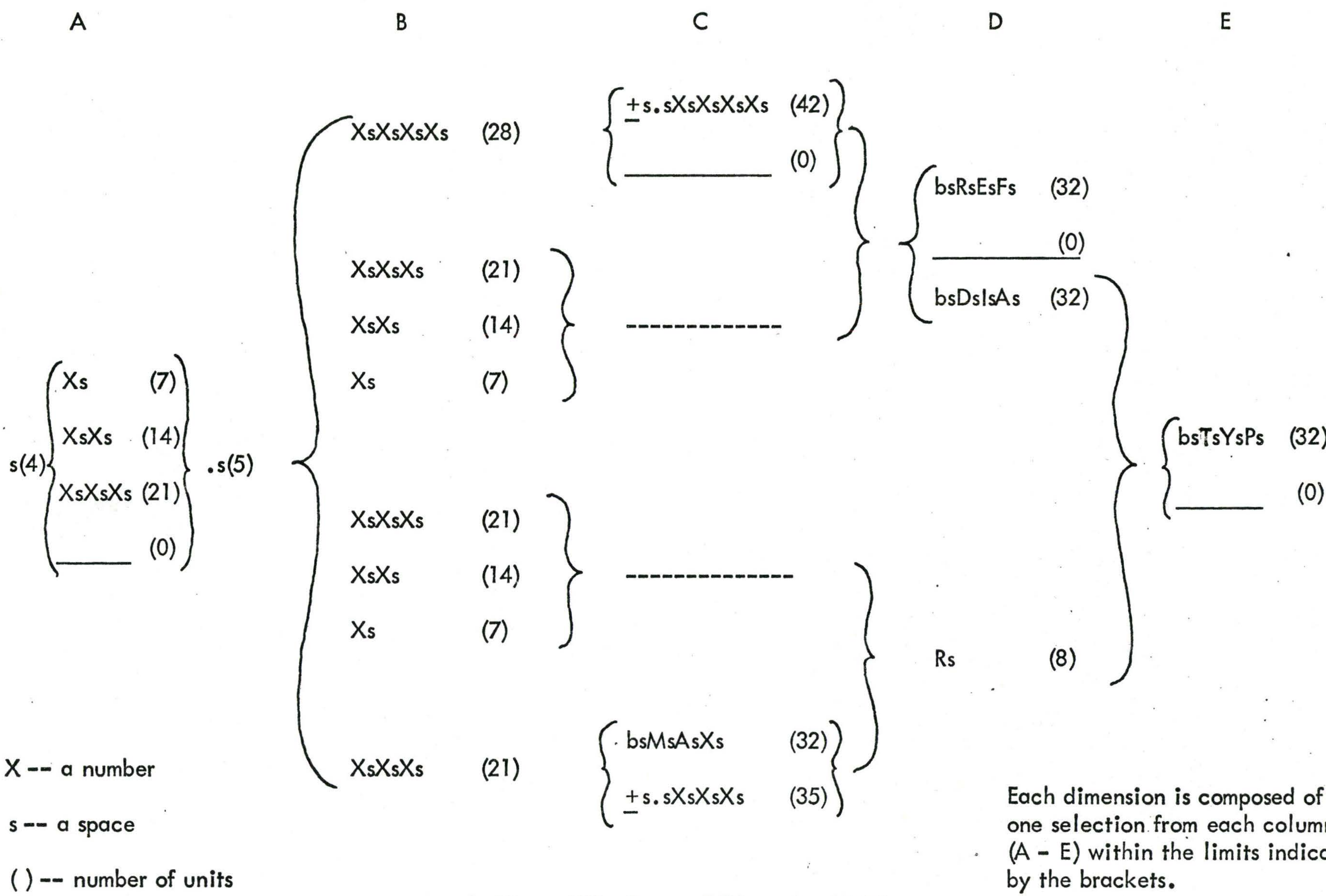


Figure XII Forms of Dimension Writing

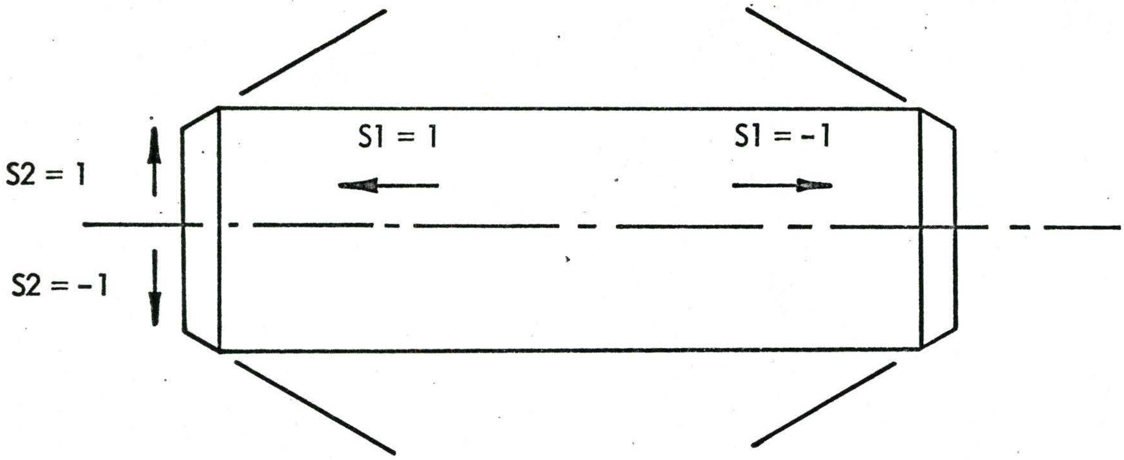
Each dimension is composed of one selection from each column (A - E) within the limits indicated by the brackets.

I. 3 Subroutine ANGPLT (TANG, X, Y)

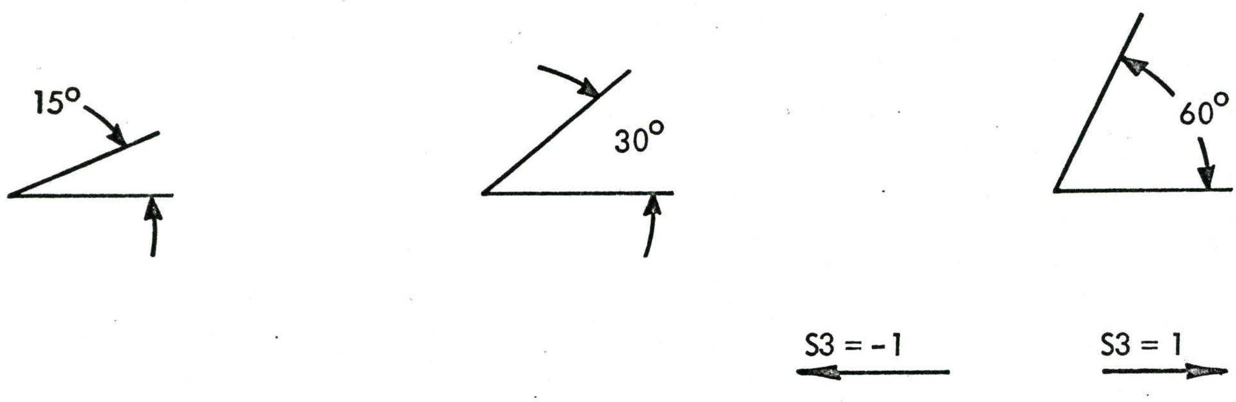
This subroutine dimensions the angle TANG at X, Y where X and Y are in data units. There are four different orientations for the dimension, depending on whether the angle is positive (diameter increasing to the right) or negative; and whether the dimension is to be above or below the shaft (determined from the sign of Y). In all cases the angle is dimensioned from the horizontal. The angle is expressed in radians and must be less than one-half pi.

In addition to four variations in orientation, there are three variations in format, depending on the size of the angle to be dimensioned (see Fig. XIII). If the angle is less than .35 radians (20° approximately) the writing and arrows are outside the witness lines; if the angle is less than .79 radians (45° approximately) but greater than .35 radians, the writing is between the witness lines but the arrows are outside; and for angles greater than .79 radians, both the arrows and writing are inside the witness lines. Considering both orientation and format, there are eight total possible positions for the arrows. Any one of the eight cases can be identified by a unique combination of S1, S2 and S3 (see Fig. XIII).

The horizontal witness line is never plotted because in some cases it could do more harm than good; in other cases it is unneeded or can be added by a draftsman.



Four Possible Orientations



Three Formats

Figure XIII Angular Dimension Formats

I. 4 Subroutine CHPLT (X, Y, ANG, WIDTH)

This subroutine is called both by subroutine ANGPLT and directly by the main program. When called by subroutine ANGPLT, WIDTH equals zero and subroutine CHPLT writes the angle, ANG, rounded to the nearest degree. X gives the location of the left end of the writing and Y gives the mid-height of the lettering (X and Y in plotter coordinates).

When subroutine CHPLT is called by the main program, it plots the writing for a noted chamfer. In this case, the argument WIDTH is non-zero and is written before the angle, according to International Harvester standards. WIDTH is always written as a decimal point followed by two digits.

I. 5 The Arrow Subroutines: ARROW (X, Y, I), ARROWA (X, Y, A), ARRA (X1, Y1, X2, Y2, XP, YP)

Either ARROW or ARROWA plots an arrowhead .15" long and .05" wide with its tip at X, Y (in data units). The proportions were determined by International Harvester standards. For ARROWA, the third argument (A) indicates the direction, in radians, in which the arrow points. A is measured counter-clockwise from the left so that, for example, A = pi/2 for an arrow pointing down. For ARROW, the third argument and integer code(I) determines the direction in which the arrow points. Multiplying I by pi/4 gives the angle (equivalent to A) measured counter-clockwise from the left. Thus ARROW is a special version of ARROWA, restricted to eight directions along the horizontal, vertical and 45° lines. It was written because for these common cases it is possible to avoid the use of the time-consuming SIN and COS functions. ARROWA is called only by subroutine ANGPLT for the dimensioning of angles.

The actual plotting of the arrowhead, identical for both ARROW and ARROWA, is accomplished by their calling ARRA (X1, Y1, X2, Y2, XP, YP). The plotting proceeds from the tip point, XP, YP, to one point, X1 , Y1, with the pen down; the pen is lifted and returned to the tip point, and finally the pen goes down as plotting moves from the tip point to the second point, X2, Y2. This apparently inconvenient order of plotting was used ot obtain the best-looking arrowhead in spite of plotter backlash. It was decided on the basis of earlier experience that simple plotting between three points in succession might yield an unsatisfactory arrowhead.

I. 6 Subroutine FORMAT

This plotting routine is called once before each shaft is plotted. It writes vertically on the page the identifying data which belongs with the shaft--the shaft name and the scale of the drawing--as shown in Fig. XIV.

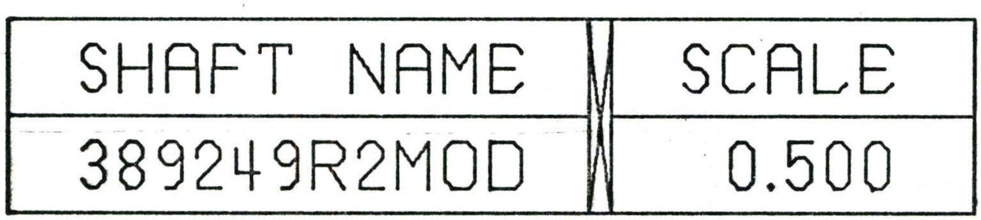


Figure XIV Sample Drawing Heading

This subroutine was not included in the main program because it was easier to de-bug when independent, and once de-bugged was most easily used as an object deck.

I. 7 Subroutine PLTARC (X1, Y1, X2, Y2, XC, YC, DEV)

This subroutine is the Benson-Lehner subroutine to plot circular arcs, modified to accept center points off the page. The nature of the modification is such that the subroutine will work only when called by the shaft plotting program. The formula for the coordinate of the center point contains a translation of 10", the distance from the plotter's y-axis to the data units y-axis.

APPENDIX II

LISTING OF PROGRAM AND SUBPROGRAMS

```
$IBFTC SHFTDW
C                                                                  DECK NO.  105
      DIMENSION A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),NUM(50)
      DIMENSION IROOT(50),POSIT(50),LEVEL(50),IREF(50),IREFQ(50),A8(50)
      DIMENSION DNOM(50),DTOL(50),RNOM(50),RTOL(50),FRNOM(50),FRTOL(50)
      DIMENSION TANG(50),II(100),JJ(50),KK(50),PL(50),PR(50),KCF(2,51)
      COMMON  SCALE,YAXIS
      COMMON  SN(2)
      COMMON WL(2,17,50)
      COMMON /BLAST/ DEV
      DIMENSION ABC(2,51), DL(51,3),YACL(50),DN(50,4),PABK(2,20)
      DIMENSION           WLH(30,3),KAB1(2),KAB2(2),M1(2),M2(2),M3(2)
      DIMENSION IH(50,2), LOH(50,2), LCODE(50,2), WLY(50), NLEV(2)
      LOGICAL DTYP(50),FRTYP(50),LAB(2,51),PLTO,LDUM,PLTSW
      DATA PLTSW/.FALSE./
    1 READ (5,5) SN,SCALE,PWIDTH,PLTO
    5 FORMAT (2A6,F11.3,F8.1,48X,L1)
      WRITE (6,10) SN,SCALE,PWIDTH
   10 FORMAT(48H1  EXECUTION OF PROGRAM SHFTDW FOR SHAFT NUMBER ,2A6// 9
     1X,8H SCALE =,F6.3/9X,23H PAPER WIDTH FOR PLOT =,F5.1)
      IF(.NOT.PLTO) GO TO 20
      WRITE(6,15)
   15 FORMAT(32H      THE INPUT CALLS FOR A PLOT.)
      GO TO 30
   20 WRITE(6,25)
   25 FORMAT(39H       THE INPUT DOESN'T CALL FOR A PLOT)
C   ZERO ARRAYS  (IMPORTANT FOR RUNS WITH MORE THAN ONE SHAFT)  30 TO 40
   30 N = 1
   32 IROOT(N) = N
      POSIT(N) = 0.0
      DO 33 L=1,2
      DO 33 M=1,17
   33 WL(L,M,N) = 0.0
   34 DO 36 L=1,2
      KCF(L,N) = 0
      LAB(L,N) = .FALSE.
   36 DL(N,L) = 0.0
      N = N+1
      IF(N-51)32,34,40
   40 WRITE (6,42)
   42 FORMAT(///40X,20H*** DATA AS READ ***///4H NUM,6X,2HA1,8X,2HA2,3X,
     1 4HIREF,6X,2HA3,9X,2HA4,6X,2HA5,7X,2HA6,7H  IREFQ,3X,2HA7,3X,
     2 4HLAST/)
C
C   INPUT LOOP 00045 TO 00390
C
      DO 390 N=1,49
      NM1 = N-1
      READ (5,49) NUM(N),A1(N),A2(N),IREF(N),A3(N),A4(N),A5(N),A6(N),
     1 IREFQ(N),A7(N),LAST
   49 FORMAT (I2,2X,2(F8.4,2X),I2,2X,2(F9.4,2X),2(F7.4,2X),I2,2X,F5.1,
     1 2X,I1)
CB-+2BB02BBB-+22202BBB-+2B-+222202BBB-+2BB202BBB-+2B02BBB-+2202BBB-+2B-+2BB
C   ABOVE IS THE DRUM CARD FOR FORMAT 49 (PUT A 2 IN COL. 1)
      WRITE(6,59) NUM(N),A1(N),A2(N),IREF(N),A3(N),A4(N),A5(N),A6(N),
     1 IREFQ(N),A7(N),LAST
   59 FORMAT(2H0 ,I2,2X,2(F8.4,2X),I2,2X,2(F9.4,2X),2(F7.4,2X),I2,2X,
     1 F5.1,2X,I1)
C   CHANGE ANGLES FROM DEGREES TO RADIANS
```

```
      A7(N) = A7(N)*.01745885
C
C     COMPLETE DATA WHERE DITTO HAS BEEN USED    100 TO 199
C
      IF(N.EQ.1) GO TO 200
      IF(A1(N).GT.0.0.OR.A2(N).EQ.0.0) GO TO 150
C     THE DIAMETER HAS BEEN DITTOED
      IA2N = IFIX(A2(N)+.5)
C     IA2N IS THE NUMBER OF THE PLANE TO BE COPIED
      DO 130 I=1,NM1
      IF(NUM(I).EQ.IA2N) GO TO 140
  130 CONTINUE
      WRITE (6,135) NUM(N)
  135 FORMAT (38H1  THE DITTO IMPLIED FOR PLANE NUMBER ,I2,32H REFERS TO
     1 AN UNDEFINED DIAMETER)
      GO TO 99990
  140 LAB(1,I) = .TRUE.
      A1(N) = A1(I)
      A2(N) = -99.
  150 IF(A5(N).GT.0.0.OR.A6(N).EQ.0.0) GO TO 200
C     THE FILLET RADIUS HAS BEEN DITTOED
      IA6N = IFIX(A6(N)+.5)
C     IA6N IS THE NUMBER OF THE PLANE TO BE COPIED
      DO 180  I=1,NM1
      IF(NUM(I).EQ.IA6N) GO TO 190
  180 CONTINUE
      WRITE(6,185) NUM(N)
  185 FORMAT (38H1  THE DITTO IMPLIED FOR PLANE NUMBER ,I2,37H REFERS TO
     1 AN UNDEFINED FILLET RADIUS)
      GO TO 99990
  190 LAB(2,I) = .TRUE.
      A5(N) = A5(I)
      A6(N) = -99.
C
C     CHECK FOR CIRCULAR REFERENCES AND UPDATE POSITION    200 TO 299
C
  200 IF(IREF(N)) 99990,201,240
C     NTH PLANE REFERS TO THE LEFT-MOST PLANE  (PLANE ZERO)    201
  201 POSIT(N) = -A3(N)
      LEVEL(N) = 1
      IROOT(N) = 0
      IF(NM1) 300,300,205
C     MOVE ALL PLANES ROOTED IN PLANE N AND ROOT THEM IN ZERO.  205 TO 230
  205 DO 230 I=1,NM1
      IF(IROOT(I)-NUM(N)) 230,220,230
  220 LEVEL(I) = LEVEL(I)+1
      IROOT(I) = 0
      POSIT(I) = POSIT(I)+POSIT(N)
  230 CONTINUE
      GO TO 300
  240 IR = IREF(N)
      IF(N.EQ.1) GO TO 260
      DO 250 I=1,NM1
      IF(IR-NUM(I)) 250,275,250
  250 CONTINUE
      IF(IR-NUM(N)) 260,255,260
  255 WRITE(6,258) NUM(N)
  258 FORMAT(16H1  PLANE NUMBER ,I2, 27H HAS BEEN REFERED TO ITSELF)
```

```
      GO TO 99990
C     THE NTH PLANE IS REFERRED TO A PLANE (IR) NOT YET READ.    260 TO 270
  260 POSIT(N) = -A3(N)
      IROOT(N) = IR
      LEVEL(N) = 1
      IF(N.EQ.1) GO TO 300
C     MOVE ALL PLANES ROOTED IN N UP ONE LEVEL AND ROOT THEM IN IR.
      DO 270 I=1,NM1
      IF(IROOT(I)-NUM(N)) 270,265,270
  265 LEVEL(I) = LEVEL(I)+1
      POSIT(I) = POSIT(I)+POSIT(N)
      IROOT(I) = IR
  270 CONTINUE
      GO TO 300
C     THE NTH PLANE IS REFERRED TO A PLANE ALREADY READ.    275  TO  290
  275 IF(IROOT(I).NE.NUM(N)) GO TO 280
      WRITE(6,278) NUM(N)
  278 FORMAT (28H1  THE REFERENCE FOR PLANE  ,I2,85H CLOSES THE LOOP OF
     1A CIRCULAR REFERENCE. CHECK THIS AND ALL REFERENCES LEADING TO IT)
      GO TO 99990
C     THE NTH PLANE IS POSITIONED RELATIVE TO ITS REFERENCE.    280
  280 POSIT(N) = POSIT(I)-A3(N)
      LEVEL(N) = LEVEL(I)+1
      IROOT(N) = IROOT(I)
      DO 290 I=1,NM1
      IF(IROOT(I)-NUM(N)) 290,285,290
C     PLANES ROOTED IN N ARE MOVED WITH IT AND SHARE ITS ROOT.    285
  285 POSIT(I) = POSIT(I)+POSIT(N)
      LEVEL(I) = LEVEL(I)+LEVEL(N)
      IROOT(I) = IROOT(N)
  290 CONTINUE
C     CHECK FOR THE LAST PLANE OF A SHAFT.    300
  300 IF(LAST)400,390,400
  390 CONTINUE
      WRITE(6,395)
  395 FORMAT(45H1  THE 49TH HAS NOT BEEN SIGNALED AS THE LAST)
      GO TO 99990
  400 NPLANE = N+1
      NPLM1 = NPLANE - 1
      NPLP1 = NPLANE +1
C     CHECK FOR TWO TYPES OF INPUT ERROR.    405  TO  499
      DO 499 I=1,N
      IF(POSIT(I))450,450,430
C     ALL POSITIONS HAVE BEEN ASSIGNED TEMPORARILY NEGATIVE TO AID SORTING
  430 WRITE(6,440) NUM(I)
  440 FORMAT(16H1  PLANE NUMBER ,I2, 44H HAS BEEN LOCATED TO THE LEFT OF
     1 PLANE ZERO.)
      GO TO 99990
  450 IF(IROOT(I).EQ.0) GO TO 499
      WRITE (6,460) IROOT(I)
  460 FORMAT(14H1 PLANE NUMBER,I3,40H HAS BEEN REFERED TO BUT NEVER DEFI
     1NED. )
      GO TO 99990
  499 CONTINUE
C
C     THE PLANES ARE RENUMBERED ON THE BASIS OF THEIR POSITION  500 TO 599
C
```

```
  500 DO 549 N=2,NPLANE
      L = NPLANE-N+2
C    L IS THE NEW (WORKING) NUMBER OF THE PLANE.
      M = 1
C    M IS THE OLD (INPUT) NUMBER OF THE PLANE.
      DO 530 I=2,NPLM1
      IF(POSIT(I)-POSIT(M)) 525,520,530
  520 IF(A7(I).LT.A7(M)) GO TO 530
  525 M = I
  530 CONTINUE
      DNOM(L) = A1(M)
      DTOL(L) = A2(M)
      RNOM(L) = A3(M)
      RTOL(L) = A4(M)
      FRNOM(L) = A5(M)
      FRTOL(L) = A6(M)
      TANG(L) = A7(M)
      A8(L) = -POSIT(M)
      POSIT(M) = A8(L)
C    POSIT  IS SWITCHED POSITIVE SO IT WILL BE OUT OF THE SORTING
      DTYP(L) = LAB(1,M)
      FRTYP(L) = LAB(2,M)
C    LAB HAS TO BE CLEARED FOR ITS LATER USE.
      LAB(1,M) = .FALSE.
      LAB(2,M) = .FALSE.
      K = NUM(M)
C    K IS THE DESIGNER'S NUMBER FOR THE PLANE.
      II(K) = L
      JJ(L) = IREF(M)
      KK(L) = IREFQ(M)
  549 CONTINUE
      DO 599 N=2,NPLANE
      IF(JJ(N))570,560,570
  560 IREF(N) = 1
      GO TO 575
  570 M = JJ(N)
      IREF(N) = II(M)
  575 POSIT(N) = A8(N)
      IF(KK(N)) 580,585,590
C    ANY NEGATIVE NUMBER IS INTERPRETED AS A REF TO THE LEFTMOST PLANE.
C    A ZERO INTERP. AS NONE-A BLANK.   THIS IS SET = TO ITSELF FOR
C     DETECTION IN S.U.L.D. (STATEMENT  1500)
  580 IREFQ(N) = 1
      GO TO 599
  585 IREFQ(N) = N
      GO TO 599
  590 M = KK(N)
      IREFQ(N) = II(M)
  599 CONTINUE
      POSIT(1) = 0.0
C
C    CHAMFER DATA COMPLETION SECTION      600 TO 699
C
      DO 699 I=2,NPLANE
      IF(FRNOM(I).NE.0.0.OR.TANG(I).EQ.0.0) GO TO 699
      IF(I.LT.NPLANE) GO TO 605
      WRITE (6,603)
```

```
  603 FORMAT(74H1  THE RIGHTMOST PLANE CANNOT BE A CHAMFER PLANE.  CHECK
     1 THE SIGN OF RNOM.)
      GO TO 99990
  605 QTAN = TAN(TANG(I))
      IT = I -IFIX(TANG(I)/ABS(TANG(I)))
C   IT  IS THE NUMBER OF THE SHOULDER PLANE.
      IF(RNOM(I)) 630,610,630
  610 D  = DNOM(I+1) - DNOM(I)
      IF(D .GT.0.0) GO TO 620
      WRITE(6,615)  POSIT(I)
  615 FORMAT(53H1  THE REFERENCE DIAMETER ON CHAMFER PLANE LOCATED AT,
     1  F6.2, 42H IS NOT SMALLER THAN THE ADJACENT DIAMETER)
      GO TO 99990
  620 POSIT(I) = POSIT(IT) +  .5*D/QTAN
      RTOL(I) = -99.
      GO TO 650
  630 DNOM(I) = DNOM(I+1)-QTAN*(POSIT(I)-POSIT(IT))*2.0
      DTOL(I) = -99.
  650 IF(TANG(I)) 670,699,699
C   THE SWITCH IS MADE SO THAT D(I) WILL ALWAYS BE THE DIAMETER LEFT
C    OF PLANE I.   670
  670 T1 = DNOM(I)
      T2 = DTOL(I)
      LDUM = DTYP(I)
      DNOM(I) = DNOM(I+1)
      DTOL(I) = DTOL(I+1)
      DTYP(I) = DTYP(I+1)
      DNOM(I+1) = T1
      DTOL(I+1) = T2
      DTYP(I+1) = LDUM
  699 CONTINUE
C
C   FIND FILLET TANGENCY POSITIONS AND MAKE INPUT CHECKS 700 TO 799
C
      PL(2) = 0.0
      DO 799 N=2,NPLANE
      IF(FRNOM(N).NE.0.0) GO TO 710
C   CHAMFER PLANE
      PR(N) = POSIT(N)
      PL(N+1) = POSIT(N)
      GO TO 780
  710 IF(TANG(N).NE.0.0) GO TO 740
C   NO TAPER ANGLE ASSOCIATED WITH PLANE
      IF(ABS(DNOM(N+1)-DNOM(N)) .GE. 2.*FRNOM(N)) GO TO 730
C   FILLET IS LESS THAN 90DEG. ARC
      P = SQRT(FRNOM(N)**2 - (FRNOM(N)-ABS((DNOM(N+1)-DNOM(N))*.5))**2)
      GO TO 770
C   FILLET IS 90DEG. ARC  730
  730 P = FRNOM(N)
      GO TO 770
C   TAPER IS ASSOC. W/ PLANE    740
C   A IS THE DIST. TO INTERSECTION OF TAPER AND HORIZONTAL
  740 A = ((DNOM(N+1)-DNOM(N))*.5) / TAN(TANG(N))
      IF(A.GE.0.0) GO TO 760
      WRITE (6,750) N
  750 FORMAT (71H1  SIGN OF DIAMETER DIFFERENCE AND TANG(N) ARE NOT IN A
     1GREEMENT FOR N =,I3)
```

```
      GO TO 99990
  760 P = A+ABS(TAN(TANG(N)*.5)) * FRNOM(N)
  770 IF(DNOM(N+1).GT.DNOM(N)) GO TO 775
C     TAPER AND/OR FILLET ARE ON RIGHT OF PLANE
      PR(N) = POSIT(N)
      PL(N+1) = P + POSIT(N)
      GO TO 780
  775 PR(N) =POSIT(N) - P
      PL(N+1) = POSIT(N)
  780 IF(PR(N).GT.PL(N)) GO TO 799
      WRITE (6,790) N
  790 FORMAT(43H1  INVESTIGATE FILLETS TO THE LEFT OF PLANE,I3,44H.  AN
     1IMPOSSIBLE GEOMETRY HAS BEEN DESCRIBED)
  799 CONTINUE
C
C     ANGULAR DIMENSION SET-UP  (ADSU)     1000 TO 1499
C
C     INITIALIZE SHAFT CONTOUR ARRAY ABC AND ZERO PABK 1000 TO 1100
      DO 1010 L=1,2
      ABC(L,1) = 0.0
 1010 ABC(L,NPLP1) = 0.0
      DO 1050 N=1,20
      DO 1030 L=1,2
 1030 PABK(L,N) = 0.0
 1050 CONTINUE
      DO 1100 N=2,NPLANE
      ABC(1,N) = DNOM(N)*.5
      IF(TANG(N).GT.0.0) ABC(1,N) = DNOM(N+1) * .5
 1100 ABC(2,N) = ABC(1,N)
      S1A = .8/SCALE
C     S1A IS THE WIDTH ALLOTTED TO AN ANGULAR DIMENSION.
      S2A = .8/SCALE
C     S2A IS THE HEIGHT ALLOTTED TO AN ANGULAR DIMENSION.
      S3A = .79/SCALE
C     S3A IS PROVIDED SO THAT AN ANGLE WILL NOT BLOCK ITS OWN PLANE.
      DO 1400 I=2,NPLANE
      DN(I,4) = 0.0
      IF(TANG(I)) 1110,1120,1130
 1110 XL = POSIT(I) - S1A
      XR = POSIT(I)
      GO TO 1150
 1120 YACL(I) = 0.0
      GO TO 1400
 1130 XL = POSIT(I)
      XR = POSIT(I) + S1A
C     TEST FOR CHAMFER TO BE INDICATED BY NOTE  1150
 1150 IF(ABS(ABS(TANG(I)) - .7854) .GT. 1E-3  .OR.  ABS(RNOM(I)).GT..5)
     1 GO TO 1160
      IF(FRNOM(I).GT.0.0) GO TO 1160
      IF(RNOM(I).EQ.0.0) GO TO 1160
      IF(IABS(IREF(I)-I) .GT. 1) GO TO 1160
C     NOTED CHAMFER
      DN(I,4) = 1.0
      RTOL(I) = -99.
      GO TO 1120
C     ANGLE NOT TO BE INDICATED BY NOTE   1160
 1160 YACL(I) = ABC(1,I)
```

```
      DN(I,4) = 0.0
C    CHECK FOR OUTLINE INTERFERENCE
      DO 1200 N=1,NPLANE
      IF(POSIT(N).LE.XL) GO TO 1200
      IF(YACL(I).LT.ABC(1,N)) GO TO 1210
      IF(POSIT(N).GE.XR) GO TO 1250
 1200 CONTINUE
      IF(ABC(1,NPLP1).LE.YACL(I)) GO TO 1250
      N = NPLP1
 1210 WRITE(6,1220) I,N
 1220 FORMAT(40H0  THE ANGLE LABEL ASSOCIATED WITH PLANE,I3,49H INTERFER
     1ES WITH THE OUTLINE TO THE LEFT OF PLANE,I3)
      GO TO 1120
C    SCAN FOR 1ST OPENING IN PABK, ABOVE.      1250
 1250 DO 1260 N=1,20
      IF(PABK(1,N)) 1260,1290,1260
 1260 CONTINUE
 1270 WRITE(6,1280)
 1280 FORMAT(62H1  PABK DIMENSION IS INADEQUATE FOR THE NUMBER OF TAPER
     1ANGLES)
      GO TO 99990
 1290 IF(N.EQ.1) GO TO 1295
      IF(PABK(1,N-1).GT.XL) GO TO 1300
C    ANGLE LABEL PLACED ABOVE
 1295 JAB = 1
      GO TO 1350
C    SCAN FOR THE 1ST OPENING IN PABK, BELOW.  1300
 1300 DO 1310 N=1,20
      IF(PABK(2,N)) 1310,1320,1310
 1310 CONTINUE
      GO TO 1270
 1320 IF(N.EQ.1) GO TO 1340
      IF(PABK(2,N-1).LE.XL) GO TO 1340
      WRITE(6,1330) I
 1330 FORMAT(40H0  THE ANGLE LABEL ASSOCIATED WITH PLANE,I3,64H CANNOT B
     1E PLACED WITHOUT INTERFERING WITH ONE TO THE LEFT OF IT)
      GO TO 1120
C    ANGLE LABEL PLACED BELOW.  1340
 1340 JAB = 2
      YACL(I) = -YACL(I)
 1350 PABK(JAB,N) = XR
      Y = ABS(YACL(I)) + S2A
C    RECORD NEW LEVELS OF INNER PROFILE TEMPORARILY IN DL.
C      (SO THEY WON'T MESS UP THE OUTLINE INTERFERENCE CHECK)
      DO 1390 J=1,NPLANE
      IF(POSIT(J).LE.XL) GO TO 1390
      DL(J,JAB) = Y
      IF(POSIT(J).GT.XR) GO TO 1400
 1390 CONTINUE
      DL(NPLP1,JAB) = Y
 1400 CONTINUE
C    UPDATE ALL ABC VALUES       1400 TO 1499
      DO 1499 J=1,NPLP1
      DO 1450 L=1,2
 1450 ABC(L,J) = AMAX1(ABC(L,J),DL(J,L))
 1499 CONTINUE
C
```

```
C     SET UP LENGTH DIMENSIONS (S.U.L.D.)        1500  TO  2499
C
C     SORT BY NUMBER OF PLANES DISTANT      1500 TO 1595
C     II(1) IS THE PLANE WITH THE LARGEST REFERENCE
C     QREFS ARE SHOWN BY NEGATIVE PLANE NUMBERS.
C     JJ(N) HOLDS THE LENGTH OF THE REF ASSOC. WITH PLANE N.
C     KK(N) HOLDS THE LENGTH OF THE QREF ASSOCIATED WITH PLANE N
      JJ(2) = IABS(2-IREF(2))
      KK(2) = IABS(2-IREFQ(2))
      IF(KK(2)) 1505,1505,1510
 1505 NQ = 0
 1507 II(1) = 2
      GO TO 1520
 1510 NQ = 1
      IF(KK(2).GT.JJ(2)) GO TO 1515
      II(2) = -2
      GO TO 1507
 1515 II(1) = -2
      II(2) = 2
 1520 DO 1595 N=3,NPLANE
      JJ(N) = IABS(N-IREF(N))
      KK(N) = IABS(N-IREFQ(N))
      IF(KK(N)-1) 1560,1550,1525
 1525 M = N+NQ-2
      DO 1535 I=1,M
      IR = II(I)
      IF(IR.GT.0) GO TO 1530
      MIR = -IR
      IF(KK(N) - KK(MIR)) 1535,1535,1540
 1530 IF(KK(N).GT.JJ(IR)) GO TO 1540
 1535 CONTINUE
      II(M+1) = -N
      GO TO 1555
C     THE CURRENT QREF GOES IN AT I.   ALL PREVIOUS ARE MOVED BACK.  1540
 1540 DO 1545 K=I,M
      L = M+I-K
 1545 II(L+1) = II(L)
      II(I) = -N
      GO TO 1555
C     QREF OF LENGTH 1 IS PUT IN LAST PLACE.      1550
 1550 M = N+NQ-1
      II(M) = -N
C     QREF COUNTER IS INCREASED BY ONE.       1555
 1555 NQ = NQ + 1
C     SHORTCUT TEST. IF JJ(N) = 1 IT IS PLACED AT THE BACK END.     1560
 1560 IF(JJ(N)-1) 1565,1590,1565
 1565 M = N+NQ-2
      DO 1575 I=1,M
      IR = II(I)
      IF(IR.GT.0) GO TO 1570
      MIR = -IR
      IF(JJ(N) - KK(MIR)) 1575,1575,1580
 1570 IF(JJ(N).GT.JJ(IR)) GO TO 1580
 1575 CONTINUE
      GO TO 1590
 1580 DO 1585 K=I,M
      L = M+I-K
```

```
 1585 II(L+1) = II(L)
      II(I) = N
      GO TO 1595
 1590 M = N+NQ-1
      II(M) = N
 1595 CONTINUE
C   DEAL WITH DIMENSIONS ONE AT A TIME, LONGEST FIRST.   1600  TO 2500
      S1L = .20/SCALE
C   S1L IS THE LENGTH OF LEADER LINE FOR TYPE 3 DIMENSION FORMAT.
      S2L = .30/SCALE
C   S2L IS THE LENGTH OF AN EXTERNAL ARROW WITH EXTERNAL WRITING.
      S3L = .45/SCALE
C   S3L IS THE LENGTH OF AN EXTERNAL ARROW WITH INTERNAL WRITING.
      S4L = .50/SCALE
C   S4L,THE MINIMUM LENGTH OF THE INTERNAL ARROW WITH INTERNAL WRITING.
      S5L = .50/SCALE
C   S5L IS THE LENGTH OF THE INTERNAL ARROW WITH EXTERNAL WRITING.
      S6L = .20/SCALE
C   S6L IS THE LENGTH ALLOTTED AN ARROWHEAD IF PROVIDING A W. L. BREAK.
      M = NPLM1 + NQ
      DO 2450 N=1,M
      IF(II(N).LT.0) GO TO 1610
      NP = II(N)
      IR = IREF(NP)
      IF(RTOL(NP).LT.(-10.)) GO TO 2450
      A = ABS(RNOM(NP))
      ATOL = RTOL(NP)
      JREF = 1
      GO TO 1620
C   THE PRESENT LENGTH DIMENSION IS A REFERENCE DIMENSION.  1610
 1610 NP = -II(N)
      IR = IREFQ(NP)
      A = ABS(POSIT(NP) - POSIT(IR))
      ATOL = -4.
      JREF = 2
C   TEST FOR PLANES BLOCKED BY ANGLES   1620 TO 1660
 1620 N1 = MIN0(NP,IR)
      N2 = MAX0(NP,IR)
      MA = 0
      DO 1630 I=1,20
      IF(PABK(1,I).LE.POSIT(N1)) GO TO 1630
      IF(PABK(1,I)-S3A.LT.POSIT(N1))MA = 1
      DO 1625 L=1,20
      IF(PABK(1,L).LE.POSIT(N2)) GO TO 1625
      IF(PABK(1,L)-S3A.LT.POSIT(N2))MA = 1
      GO TO 1635
 1625 CONTINUE
      GO TO 1635
 1630 CONTINUE
 1635 MB = 0
      DO 1645 I=1,20
      IF(PABK(2,I).LE.POSIT(N1)) GO TO 1645
      IF(PABK(2,I)-S1A.LT.POSIT(N1))MB = 1
      DO 1640 L=1,20
      IF(PABK(2,L).LE.POSIT(N2)) GO TO 1640
      IF(PABK(2,L)-S1A.LT.POSIT(N2))MB = 1
      GO TO 1650
 1640 CONTINUE
```

```
      GO TO 1650
 1645 CONTINUE
 1650 IF(MA+MB.LT.2) GO TO 1660
      WRITE(6,1655) NP,IR
 1655 FORMAT(39H0  UNABLE TO PLACE DIMENSION FROM PLANE,I3, 9H TO PLANE,
     1I3,37H BECAUSE OF INTERFERENCE WITH ANGLES.)
      RTOL(NP) = -50.
      GO TO 2450
 1660 JAB = MB
      IF(MA.EQ.1) JAB = 2
C   WC = THE LENGTH OF THE WRITING (IN DATA UNITS).
      WC = CLEN(A,ATOL,0,JREF-1,.FALSE.)/SCALE
      NL = N1 + 1
      NR = N2 - 1
C   COUNT THE NUMBER OF APPLICABLE WITNESS LINES DRAWN ALREADY.
      LA = 0
      LB = 0
      IF(LAB(1,NP)) LA = LA + 1
      IF(LAB(1,IR)) LA = LA + 1
      IF(LAB(2,IR)) LB = LB + 1
      IF(LAB(2,NP)) LB = LB + 1
C   TEST TO SEE WHICH ARROW FORMAT WILL BE USED.
C   T1 AND T2 ARE THE LONGER AND SHORTER OVERHANG DISTANCES OF THE
C    DIMENSIONING BEYOND THE PLANES CONCERNED.
      IF(A.LT.WC+S4L) GO TO 1670
C   INTERNAL ARROWS AND LETTERING.
      LCODE(NP,JREF) = 1
      IF(NL.GT.NR) GO TO 1700
      T1 = 0.0
      T2 = 0.0
      GO TO 1800
 1670 IF(A.LT.WC) GO TO 1680
C   INTERNAL WRITING AND EXTERNAL ARROWS.
      LCODE(NP,JREF) = 2
      T1 = S3L
      T2 = S3L
      GO TO 1800
 1680 IF(A.LT.S5L) GO TO 1690
C   INTERNAL ARROWS AND EXTERNAL WRITING
      LCODE(NP,JREF) = 3
      T1 = S1L + WC
      T2 = 0.0
      GO TO 1800
C   EXTERNAL ARROWS AND WRITING.   1690
 1690 T1 = S2L + WC
      T2 = S2L
      LCODE(NP,JREF) = 4
      GO TO 1800
C   THE SIMPLE CASE OF NEIGHBORING PLANES WITH NO OVERHANG. 1700 TO 1800
 1700 IF(JAB.GT.0) GO TO 1740
      IF(LA-LB) 1710,1720,1730
 1710 JAB = 2
      GO TO 1740
 1720 IF(KCF(1,NL).GT.KCF(2,NL)) GO TO 1710
 1730 JAB = 1
 1740 K = KCF(JAB,NL) + 1
      KCF(JAB,NL) = K
      ASSIGN 1780 TO M
```

```
C    PROVIDE FOR STARTING NEW WITNESS LINES(WL) AND SIGNALLING
C     THEIR PRESENCE(LAB). 1750
 1750 IF(LAB(JAB,NP)) GO TO 1755
      WL(JAB,1,NP) =(.375*FLOAT(K) - .12)/SCALE
      LAB(JAB,NP) = .TRUE.
      GO TO 1760
 1755 WL(JAB,1,NP) = AMIN1(WL(JAB,1,NP),(.375*FLOAT(K)-.12)/SCALE)
 1760 IF(LAB(JAB,IR)) GO TO 1765
      WL(JAB,1,IR) =(.375*FLOAT(K) - .12)/SCALE
      LAB(JAB,IR) = .TRUE.
      GO TO 1770
 1765 WL(JAB,1,IR) = AMIN1(WL(JAB,1,IR),(.375*FLOAT(K)-.12)/SCALE)
 1770 GO TO M, (1780,2100)
 1780 IH(NP,JREF) = KCF(JAB,NL) * (1-2*(JAB-1))
      GO TO 2450
C    THE NON-SIMPLE CASE WITH POSSIBLE INTERFERENCE.    1800 TO 2450
C    ESTABLISHE END PLANES FOR POSSIBLE LEFT AND RIGHT LOCATIONS,
C     LEFT END FIRST.    1800 TO 1900
 1800 END1 = POSIT(N1) - T1
      END2 = POSIT(N1) - T2
      IF(END2.GE.0.0) GO TO 1805
      NL2 = 1
      NL1 = 1
      GO TO 1840
 1805 NL2 = NL
 1810 IF(POSIT(NL2-1).LE.END2) GO TO 1820
      NL2 = NL2 - 1
      GO TO 1810
 1820 IF(END1.GE.0.0) GO TO 1825
      NL1 = 1
      GO TO 1840
 1825 NL1 = NL2
 1830 IF(POSIT(NL1-1).LE.END1) GO TO 1840
      NL1 = NL1 - 1
      GO TO 1830
C    ESTABLISH RIGHT END PLANES.    1840
 1840 END2 = POSIT(N2) + T1
      END1 = POSIT(N2) + T2
      IF(END1.LE.POSIT(NPLANE)) GO TO 1845
      NR2 = NPLANE
      NR1 = NPLANE
      GO TO 1900
 1845 NR1 = NR
 1850 IF(POSIT(NR1+1).GE.END1) GO TO 1860
      NR1 = NR1 + 1
      GO TO 1850
 1860 IF(END2.LE.POSIT(NPLANE)) GO TO 1865
      NR2 = NPLANE
      GO TO 1900
 1865 NR2 = NR1
 1870 IF(POSIT(NR2+1).GE.END2) GO TO 1900
      NR2 = NR2 + 1
      GO TO 1870
C    SURVEY NUMBER OF INTERFERENCES(M1 AND M2) AND SPACE(KAB1 AND KAB2)
C     1900  TO 2000
 1900 DO 2000 L = 1,2
      M1(L) = 0
      M2(L) = 0
```

```
      KAB1(L) = KCF(L,NR1+1)
      KAB2(L) = KCF(L,NR2+1)
      DO 1940 J=NL1,NR1
      KAB1(L) = MAX0(KAB1(L),KCF(L,J))
      IF(J.EQ.NP.OR.J.EQ.IR) GO TO 1940
      IF(LAB(L,J))  M1(L) = M1(L) + 1
 1940 CONTINUE
      DO 1980 J=NL2,NR2
      KAB2(L) = MAX0(KAB2(L),KCF(L,J))
      IF(J.EQ.NP.OR.J.EQ.IR) GO TO 1980
      IF(LAB(L,J))  M2(L) = M2(L) + 1
 1980 CONTINUE
      M3(L) = MIN0(M1(L),M2(L))
 2000 CONTINUE
C    DETERMINE WHETHER ABOVE(JAB=1) OR BELOW(JAB=2) IS PREFERABLE.
      IF(JAB.GT.0) GO TO 2050
      IF(M3(2)-M3(1)) 2010,2020,2030
 2010 JAB = 2
      GO TO 2050
 2020 IF(LA-LB) 2010,2040,2030
 2030 JAB = 1
      GO TO 2050
 2040 IF(MIN0(KAB1(1),KAB2(1)) - MIN0(KAB1(2),KAB2(2))) 2030,2030,2010
C    CHOOSE THE RIGHT OR LEFT OPTION      2050  TO  2090
 2050 IF(M1(JAB)-M2(JAB)) 2060,2070,2080
C    LEFT OPTION.    2060
 2060 LO = -1
      K = KAB1(JAB) + 1
      N1 = NL1
      N2 = NR1 + 1
      GO TO 2090
 2070 IF(KAB1(JAB).LE.KAB2(JAB)) GO TO 2060
C    RIGHT OPTION.   2080
 2080 LO = 1
      K = KAB2(JAB) + 1
      N1 = NL2
      N2 = NR2 + 1
 2090 ASSIGN 2100 TO M
      GO TO 1750
C    RETURN FROM UPDATE LOGICAL ARRAY, LAB.      2100
 2100 IH(NP,JREF) = K*(1-2*(JAB-1))
      LOH(NP,JREF) = LO
C    UPDATE KCF.
      DO 2130  I=N1,N2
 2130 KCF(JAB,I) = K
C    TEST FOR INTERFERENCE FOR PROVIDING WITNESS LINE BREAKS.
      IF(M3(JAB).EQ.0) GO TO 2450
      NM1 = N2 - 1
      DO 2300 I=N1,NM1
      IF(.NOT.LAB(JAB,I)) GO TO 2300
      IF(I-NL+1) 2150,2300,2140
 2140 IF(I-NR-1) 2200,2300,2160
C    PLANE I IS LEFT OF THE LEFT WITNESS LINE.  2150
 2150 B = POSIT(NL-1) - POSIT(I)
      GO TO 2170
C    PLANE I IS RIGHT OF THE RIGHT WITNESS LINE.  2160
 2160 B = POSIT(I) - POSIT(NR+1)
 2170 IF(B.GT.T2)  CALL SKIP(JAB,I,K,.10)
```

```
      IF(B.LE.S6L) CALL SKIP(JAB,I,K,.05)
      GO TO 2300
C   PLANE I IS BETWEEN THE WITNESS LINES.     2200
 2200 IF(T1.NE.T2) GO TO 2220
C   T1 = T2 MEANS THE WRITING IS BETWEEN THE WITNESS LINES
C   B IS THE MIDPOINT. (RECALL THAT A IS THE DISTANCE BETWEEN W.L.).
      B = POSIT(NL-1) + A*.5
      IF(ABS(B-POSIT(I)).GE.WC*.5) GO TO 2220
      CALL SKIP(JAB,I,K,.10)
      GO TO 2300
C   TEST FOF INTERNAL ARROWS.     2220
 2220 IF(A.LT.S5L) GO TO 2300
      B = AMIN1(POSIT(I)-POSIT(NL-1),POSIT(NR+1)-POSIT(I))
      IF(B.LE.S6L) CALL SKIP(JAB,I,K,.05)
 2300 CONTINUE
 2450 CONTINUE
C
C   LOCATE DIAMETRAL DIMENSION LINES     2500   TO   3499
C
C   ZERO ARRAY WLH
 2500 DO 2550 J=1,30
 2550 WLH(J,1) = 0.0
C   LOCATE DIAMETERS, CENTRAL ONES FIRST  2600   TO   3499
      S2D = .30/SCALE
C   S2D IS THE LENGTH OF HORIZONTAL WITNESS LINES
      DO 3499 I=1,NPLM1
      N = (NPLANE+2)/2 + (1-(MOD(I,2)*2))*I/2
      IF(DTOL(N).GT.(-10.)) GO TO 2610
C   TOLERANCE = -99.,INDICATING A TYPICAL DIMENSION NOT TO APPEAR
      GO TO 3490
 2610 IF(FRNOM(N).NE.0.0  .OR.  TANG(N).LE. 0.0) GO TO 2620
C   A LEFT CHAMFER
      IF(RNOM(N).EQ.0.0) GO TO 2680
C   THE CHAMFER HAS NO SHOULDER DIAMETER.
      GO TO 3490
 2620 IF(FRNOM(N-1).NE.0.0 .OR. TANG(N-1).GE.0.0) GO TO 2630
C   A RIGHT CHAMFER
      IF(RNOM(N-1).EQ.0.0) GO TO 2680
C   THE CHAMFER HAS NO SHOULDER DIAMETER.
      GO TO 3490
 2630 A = POSIT(N)-POSIT(N-1)
      IF(A.GT..20/SCALE) GO TO 2640
C   A SECTION NARROW ENOUGH THAT WITNESS LINES MUST BE USED
      GO TO 2690
 2640 B = PR(N) - PL(N)
C   COMPLICATED METHOD OF DEDUCING TWO POSSIBLE X LOCATIONS FOR THE
C    DIAMETER INDICATING ARROW. COMPLICATION ARISES FROM EFFORT TO BE
C    CONSISTENT WITH LOWER LIMIT ON A OF .20/SCALE.
      XL2 = AMAX1(PL(N)+.1*B,POSIT(N-1)+.10/SCALE)
      XR1 = AMIN1(PR(N)-.1*B,POSIT(N)-.10/SCALE)
      IF(XL2.LT.XR1) GO TO 2950
      XL2 = PL(N) +.5*B
      XR1 = XL2
      S1D = 0.0
      GO TO 2950
C   CHAMFER WITH SHOULDER DIAMETER TO BE INDICATED.     2680
 2680 XD1 = POSIT(N) - S2D
      XD2 = POSIT(N) + S2D
```

```
      ASSIGN 2800 TO K
      GO TO 2700
C   GROOVE OR RIDGE REQUIRING WITNESS LINES. 2690
 2690 XD1 = POSIT(N-1) - S2D
      XD2 = POSIT(N) +S2D
      ASSIGN 2830 TO K
C   SEGMENT TO CHECK FOR INTERFERENCE OF DIAMETER DIMENSION LINES WITH
C    THE ANGLES RECORDED BY A.D.S.U. IN PABK.  K1 AND K2 COUNT
C    INTERFERENCES FOR THE LEFT AND RIGHT ARROW POSITIONS RESPECTABLY.
 2700 JAB1 = 0
      JAB2 = 0
      K1 = 0
      K2 = 0
C   EXAMINE THE LEFT HAND POSSIBILITY, ABOVE AND BELOW
      DO 2740 L=1,2
      DO 2730 J=1,20
      IF(PABK(L,J).EQ.0.0) GO TO 2740
      IF(PABK(L,J).LE.XD1) GO TO 2730
      IF(XD1.LE.PABK(L,J)-S1A) GO TO 2740
      K1= K1 + 1
      JAB1 = 3-L
      GO TO 2740
 2730 CONTINUE
 2740 CONTINUE
C   EXAMINE THE RIGHT HAND POSSIBILITY, ABOVE AND BELOW
      DO 2780 L=1,2
      DO 2770 J=1,20
      IF(PABK(L,J).EQ.0.0) GO TO 2780
      IF(PABK(L,J).LE.XD2) GO TO 2770
      IF(XD2.LE.PABK(L,J)-S1A) GO TO 2780
      K2 = K2 +1
      JAB2 = 3-L
      GO TO 2780
 2770 CONTINUE
 2780 CONTINUE
      GO TO K,(2800,2830,3050)
C   CHOOSE THE SIDE FOR WITNESS LINE PLACEMENT.   2800  TO  2899
C   SHOULDER DIAMETER, ONE SIDE ELIMINATED ONLY IF INTERFERENCE ABOVE
C    AND BELOW.     2800
 2800 IF(K1-2) 2820,2810,2820
 2810 IF(K2.LT.2) GO TO 2890
      WRITE(6,2815) N
 2815 FORMAT(59H0  CANNOT DIMENSION SHOULDER DIAMETER ASSOCIATED WITH PL
     1ANE,I3,39H BECAUSE OF INTERFERENCE WITH AN ANGLE.)
      GO TO 3490
 2820 IF(K2.EQ.2) GO TO 2880
      IF(TANG(N)) 2890,2890,2880
C   GROOVE OR RIDGE, EITHER SIDE ELIMINATED BECAUSE OF MORE
C    INTERFERENCES. 2830
 2830 IF(K1-K2) 2880,2840,2890
 2840 IF(K1.LT.2) GO TO 2850
      WRITE(6,2845)N
 2845 FORMAT(49H0  CANNOT DIMENSION DIAMETER TO THE LEFT OF PLANE,I3,
     1 39H BECAUSE OF INTERFERENCE WITH AN ANGLE.)
      GO TO 3490
 2850 IF(POSIT(N+1)-POSIT(N) .GT. POSIT(N-1)-POSIT(N-2)) GO TO 2890
C2880CHOOSE=LEFT.OHAND WITNESS LINES, FAVORING WRITING ABOVE.   2880
      IS = N-1
```

```
      JAB = MAX0(JAB1,1)
      GO TO 2900
C    CHOOSE RIGHT HAND WITNESS LINES, FAVORING WRITING ABOVE.   2890
 2890 S1D = 1.0
      IS = N
      JAB = MAX0(JAB2,1)
C    SCAN FOR 1ST OPEN SPOT IN WLH  2900
 2900 DO 2910 J=1,30
      IF(WLH(J,1).EQ.0.0 ) GO TO 2920
 2910 CONTINUE
      WRITE(6,2915)
 2915 FORMAT(25H1  WLH DIMENSION EXCEEDED)
      GO TO 99990
 2920 X1 = POSIT(IS) + S1D*S2D
      WLH(J,1) = DNOM(N)*.5
      WLH(J,2) = X1 + S1D*S2D*.2
      WLH(J,3) = X1 - S1D*S2D*.65
      XL2 = X1
      XR1 = X1
      JSTORE = J
C    J IS STORED FOR UPDATING ABC (3330)
 2950 WC = CLEN(DNOM(N),DTOL(N),2,0,DTYP(N))/SCALE
      XR2 = XL2+WC+S2D
      XL1 = XR1-WC-S2D
C    LEFT HAND ALTERNATE (XR1,XL1 ARE ITS ENDPOINTS) CORRESPONDS TO
C     RIGHT HAND ARROW LOCATION.
      ID = 1+(N-2)/(NPLANE/2)
C    ID SHOWS LEFT(ID=1) OR RIGHT(ID=2) CHOICE
      IF(I.EQ.1) ID = 0
      S3D = 1.0
      IF(XL2.GE.XL1) GO TO 2960
      S3D = -1.0
C    S3D RECORDS THE SWITCH MADE TO ASSURE US THAT 1 IS THE LEFT HAND SET
C     OF END POINTS
C    AFTER SWITCHING, LEFT HAND PLACEMENT (ALTERNATE 1) CORRESPONDS TO
C     LEFT HAND ARROW LOCATION.
      XL = XL1
      XR = XR1
      XL1 = XL2
      XR1 = XR2
      XR2 = XR
      XL2 = XL
C    IF A SWITCH IS MADE, THIS MEANS SECTION IS LONG ENOUGH THAT
C     DIAMETER DIMENSION INTERFERENCE CANNOT BE A PROBLEM.
      ID = 0
C    TEST FOR INTERFERENCE WITH ANGLES FOR THOSE NOT ALREADY TESTED  2960
 2960 IF(S1D.EQ.0.0 ) GO TO 2970
C    DIAMETER WITH WITNESS LINES, ALREADY TESTED
      GO TO 3105
 2970 IF(XR1.NE.XL2 .OR. S3D.EQ.(-1.0))  GO TO 3000
C    THERE IS ONLY ONE POSSIBLE X POSITION FOR THE DIAMETER ARROW.
C    EXAMINE ABOVE AND BELOW
      K = 0
      JAB = 0
      DO 2990 L=1,2
      DO 2980 J=1,20
      IF(PABK(L,J).EQ.0.0) GO TO 2990
      IF(PABK(L,J).LE.XR1) GO TO 2980
```

```
      IF(XR1.LE.PABK(L,J)-S1A) GO TO 2990
      K = K+1
      JAB = 3-L
      GO.TO 2990
 2980 CONTINUE
 2990 CONTINUE
      IF(K.LT.2) GO TO 3105
      WRITE(6,2995) N
 2995 FORMAT(32H0  DIAMETER TO THE LEFT OF PLANE,I3,73H IS NOT BEING LAB
     1ELED BECAUSE INTERFERENCE WITH AN ANGLE WAS UNAVOIDABLE.)
      GO TO 3490
 3000 ASSIGN 3050 TO K
      IF(S3D)3010,3010,3020
 3010 XD1 = XL1
      XD2 = XR2
      GO TO 2700
 3020 XD1 = XL2
      XD2 = XR1
      GO TO 2700
C   RETURN FROM INTERFERENCE COUNTING.  3050
 3050 IF(K1.LT.2 .OR. K2.LT.2) GO TO 3060
      WRITE(6,2995) N
      GO TO 3490
C   ONE SIDE IS BLOCKED COMPLETELY. THE OTHER HAS AT LEAST ONE OPENING.
C       3060 AND 3070
 3060 IF(K2.LT.2) GO TO 3070
      ID = 1
C   ID SHOWS A LEFT OR RIGHT PREFERENCE AS JAB SHOWS ONE ABOVE OR BELOW
      GO TO 3100
 3070 IF(K1.LT.2) GO TO 3080
      ID = 2
      GO TO 3100
 3080 IF(ID.NE.0) GO TO 3100
      IF(JAB1.NE.JAB2) GO TO 3090
C   BOTH SIDES IDENTICAL W/ AT LEAST ONE OPENING
      JAB = JAB1
      GO TO 3100
C   IDA AND IDB ARE INDICATORS OF (RIGHT OR LEFT SIDE) CHOICE ABOVE AND
C    BELOW RESPECTIVELY.  THE TWO SIDES ARE DIFFERENT.    3090
 3090 IDA = 0
      IDB = 0
      JAB = 0
      IF(JAB1.EQ.1) IDB = 2
      IF(JAB2.EQ.1) IDB = 1
      IF(JAB1.EQ.2) IDA = 2
      IF(JAB2.EQ.2) IDA = 1
      GO TO 3105
 3100 IF(ID.EQ.1)  JAB = JAB1
      IF(ID.EQ.2)  JAB = JAB2
      IDA = ID
      IDB = ID
C   CHECK WITNESS LINE INTERFERENCES AND SPACE FOR FOUR ALTERNATIVES.
 3105 DO 3110 L=1,2
      M1(L) = 0
      M2(L) = 0
      KAB1(L) = 0
 3110 KAB2(L) = 0
C   THIS ROUTINE DEPENDS ON ALTERNATE 1 BEING TO THE LEFT OF ALTERNATE 2
```

```
      DO 3190 K=1,NPLANE
      IF(POSIT(K).LT.XL1) GO TO 3130
      IF(POSIT(K).GT.XR1) GO TO 3150
      DO 3120 L=1,2
      IF(LAB(L,K)) M1(L)=M1(L)+1
C   M1 TALLYS THE NUMBER OF WITNESS LINES CUT BY ALTERNATE 1.
 3120 CONTINUE
      GO TO 3135
 3130 IF(POSIT(K+1).LT.XL1) GO TO 3190
 3135 DO 3140 L=1,2
 3140 KAB1(L) = MAX0(KAB1(L),KCF(L,K+1))
 3150 IF(POSIT(K).LT.XL2) GO TO 3170
      IF(POSIT(K).GT.XR2) GO TO 3200
      DO 3160 L=1,2
      IF(LAB(L,K)) M2(L) = M2(L) + 1
 3160 CONTINUE
      GO TO 3175
 3170 IF(POSIT(K+1).LT.XL2) GO TO 3190
 3175 DO 3180 L=1,2
 3180 KAB2(L) = MAX0(KAB2(L),KCF(L,K+1))
 3190 CONTINUE
C   DETERMINE WHETHER THE DIMENSION WILL BE PLACED ABOVE (JAB=1) OR
C    BELOW (JAB=2).  3200
 3200 IF(JAB.GT.0) GO TO 3260
      IF(IDA-1) 3205,3210,3215
 3205 MA = MIN0(M1(1),M2(1))
      KA = MIN0(KAB1(1),KAB2(1))
      GO TO 3220
 3210 MA = M1(1)
      KA = KAB1(1)
      GO TO 3220
 3215 MA = M2(1)
      KA = KAB2(1)
 3220 IF(IDB-1) 3225,3230,3235
 3225 MB = MIN0(M1(2),M2(2))
      KB  = MIN0(KAB1(2),KAB2(2))
      GO TO 3240
 3230 MB = M1(2)
      KB = KAB1(2)
      GO TO 3240
 3235 MB = M2(2)
      KB = KAB2(2)
C   FIRST CRITERION IS MINIMIZING INTERFERENCE.   3240
 3240 IF(MA-MB) 3255,3250,3245
 3245 JAB = 2
      IF(ID.EQ.0)  ID=IDB
      GO TO 3260
 3250 IF(KA-KB) 3255,3255,3245
 3255 JAB = 1
      IF(ID.EQ.C)  ID=IDA
C   DETERMINE WHETHER DIMENSION WILL BE GIVEN LEFT OF RIGHT PLACEMENT.
 3260 IF(ID-1) 3265,3290,3280
 3265 IF(M1(JAB)-M2(JAB)) 3290,3270,3280
 3270 IF(KAB1(JAB)-KAB2(JAB)) 3290,3290,3280
C   RECORD HAND OF TURN(AS OPPOSED TO HAND OF PLACEMENT) IN DL(N,3)
C   RIGHT HAND PLACEMENT (ALTERNATE 2).   3280
 3280 DL(N,3) = S3D
      K = KAB2(JAB) + 1
```

```
      IF(S3D.GT.0.0) GO TO 3285
C     LEFT HAND TURN
      DL(N,1) = XR2
      GO TO 3300
C     RIGHT HAND TURN 3285
 3285 DL(N,1) = XL2
      GO TO 3300
C     LEFT HAND PLACEMENT(ALTERNATE 1).    3290
 3290 DL(N,3) = -S3D
      K = KAB1(JAB) + 1
      IF(S3D.GT.0.0) GO TO 3295
C     RIGHT HAND TURN
      DL(N,1) = XL1
      GO TO 3300
C     LEFT HAND TURN  3295
 3295 DL(N,1) = XR1
C     RECORD PLACEMENT LEVEL OF WRITING IN DL(N,2).  3300
 3300 DL(N,2) = FLOAT(K*(1-2*(JAB-1)))
C     X1 AND X2 ARE THE TWO ENDS OF THE WRITING.
      X1 = DL(N,1) + DL(N,3)*S2D
      X2 = DL(N,1) + DL(N,3)*WC
C     XLO AND XRO ARE THE ENDS OF WRITING PLUS LEADER.
      XLO = AMIN1(X2,DL(N,1))
      XRO = AMAX1(X2,DL(N,1))
C     XLI AND XRI ARE THE LEFT AND RIGHT ENDS OF THE WRITING.
      XLI = AMIN1(X1,X2)
      XRI = AMAX1(X1,X2)
C     UPDATE LEVELS FILLED ARRAY, KCF  AND PROVIDE FOR W.L. BREAKS
      DO 3325 J=1,NPLANE
      IF(POSIT(J).LE.XLO) GO TO 3325
      KCF(JAB,J) = K
      IF(POSIT(J).GE.XRO) GO TO 3330
      IF(XLI.GE.POSIT(J) .OR. XRI.LE.POSIT(J)) GO TO 3325
      IF(.NOT.LAB(JAB,J)) GO TO 3325
      CALL SKIP(JAB,J,K,.10)
 3325 CONTINUE
      KCF(JAB,NPLP1) = K
 3330 IF(S1D.EQ.0.0) GO TO 3499
C     FOR CASES WITH WITNESS LINES, ABC IS UPDATED
      J = JSTORE
      XL = AMIN1(WLH(J,2),WLH(J,3))
      XR = AMAX1(WLH(J,2),WLH(J,3))
      DO 3350 L=1,NPLANE
      IF(POSIT(L).LE.XL) GO TO 3350
      ABC(1,L) = AMAX1(WLH(J,1),ABC(1,L))
      ABC(2,L) = AMAX1(WLH(J,1),ABC(2,L))
      IF(POSIT(L).GE.XR) GO TO 3499
 3350 CONTINUE
      ABC(1,NPLP1) = AMAX1(WLH(J,1),ABC(1,NPLP1))
      ABC(2,NPLP1) = AMAX1(WLH(J,1),ABC(2,NPLP1))
      GO TO 3499
 3490 DL(N,2) = 0.0
 3499 CONTINUE
C
C     SET-UP NOTES WITH LEADERS       3500    TO   3999
C
      S1N = .30/SCALE
C     S1N IS THE LENGTH OF THE HORIZONTAL TAIL OF THE LEADER ARROWS.
```

```
C·  LOCATE LEADER ARROW POINT        3500   TO  3700
      DO 3700 I=2,NPLANE
      IF(DN(I,4).NE.1.0) GO TO 3600
C   THIS IS A NOTED CHAMFER.
      J = IFIX(SIGN(1.,TANG(I)))
      IMJ = I-J
      DN(I,1) = POSIT(I)-.5*(POSIT(I)-POSIT(IMJ))
C   DN(I,1) IS POSITIVE FOR RIGHT POINTING ARROWS
      DN(I,1) = DN(I,1) * FLOAT(J)
      DN(I,2) = (DNOM(I)+DNOM(I+1)) * .25
      GO TO 3700
C   TEST FOR CHAMFER(NOT A NOTED ONE).  3600
 3600 IF(FRNOM(I).EQ.0.0) GO TO 3690
C   TEST FOR DIMENSION TO BE OMITTED.
      IF(FRTOL(I).LT.(-10.)) GO TO 3690
      IF(DNOM(I)-DNOM(I+1)) 3610,3690,3620
 3610 RB = DNOM(I+1)*.5
      RS = DNOM(I)*.5
      J = 1
      GO TO 3630
 3620 RB = DNOM(I) * .5
      RS = DNOM(I+1) * .5
      J = -1
C   TEST FOR A TAPER.    3630
 3630 IF(TANG(I).EQ.0.0) GO TO 3640
      IF(ABS(TANG(I)).LE..7854) GO TO 3650
C   LOCATE AT 45DEG. PT. WITH TAPER ENTERING PROBLEM.
      DN(I,1) = (POSIT(I)-(RB-RS)/TAN(TANG(I))-FRNOM(I)*TAN(TANG(I)*.5)
     1   + FRNOM(I)*SIN(TANG(I)*.5)) * FLOAT(J)
      DN(I,2) = RS + FRNOM(I)*(1.-COS(TANG(I)*.5))
      GO TO 3660
C   NO TAPER INVOLVED.   3640
 3640 IF(RB-RS.LT..293*FRNOM(I)) GO TO 3650
      DN(I,1) =(POSIT(I) - .293*FRNOM(I)*FLOAT(J)) * FLOAT(J)
      DN(I,2) = RS + .293 * FRNOM(I)
      GO TO 3660
C   FOR LACK OF BETTER TO DO THE ARROW IS DIRECTED TO THE OUTLINE.  3650
 3650 DN(I,1) = POSIT(I) * FLOAT(J)
      DN(I,2) = RB
 3660 DN(I,4) = 2.0
      GO TO 3700
C   SET DN4 = 0 FOR NO NOTE.     3690
 3690 DN(I,4) = 0.0
 3700 CONTINUE
C   LOCATE THE WRITING LEVEL FOR NOTES.  3700 TO 3999
      DO 3999 N=2,NPLANE
      N2I = NPLANE + 2 - N
      IF(DN(N,4).EQ.0.0 .OR.  DN(N,1).LT.0.0) GO TO 3970
C   A RIGHT POINTING NOTE TO BE LOCATED.
      ASSIGN 3970 TO K
      I = N
      IF(DN(N,4).GT.1.5) GO TO 3710
      WC = 1.02
      GO TO 3720
 3710 WC = CLEN(FRNOM(N),FRTOL(N),1,0,FRTYP(N))
C   LOCATING ROUTINE.    3720 TO 3950
 3720 S = SIGN(1.,DN(I,1))
C   S IS POSITIVE FOR A RIGHT-POINTING NOTE.
```

```
      IS = IFIX(S+.5)
      T = .375/SCALE
      D = FLOAT(IFIX((DN(I,2) + .40/SCALE)/T)+1) * T
C     D IS THE 1ST POSSIBLE LEVEL
      P = ABS(DN(I,1)) - S*(D-DN(I,2))
C     P IS THE PROPOSED CROOK POINT X LOCATION.
      Q = P - S*(WC/SCALE + S1N)
C     Q IS THE CORRESPONDING X LOC OF THE END OF THE WRITING.
      XL = AMIN1(P,Q)
      XR = AMAX1(P,Q)
 3750 NL = 1
      DO 3770 J=1,NPLANE
      IF(POSIT(J).GT.XL) GO TO 3760
      NL = NL + 1
      GO TO 3770
 3760 NR = J
      IF(POSIT(J).GT.XR) GO TO 3780
 3770 CONTINUE
      NR = NPLP1
 3780 DO 3850 L=1,2
      DO 3800 J=NL,NR
      IF(D-ABC(L,J).LT..35/SCALE) GO TO 3850
 3800 CONTINUE
C     SUCCEED AT THIS LEVEL
      GO TO 3900
 3850 CONTINUE
C     FAIL AT THIS LEVEL.  TRY THE NEXT ONE OUT
      P = P-T*S
      XL = XL - T*S
      XR = XR - T*S
      D = D + T
      GO TO 3750
 3900 DO 3910 J=NL,NR
 3910 ABC(L,J) = D
      DN(I,3) = D*FLOAT(1-2*(L-1))
      GO TO K (3970,3999)
 3970 IF(DN(N2I,4).EQ.0.0 .OR. DN(N2I,1).GT.0.0) GO TO 3999
      IF(DN(N2I,4).GT.1.5) GO TO 3980
      WC = 1.02
      GO TO 3990
 3980 WC = CLEN(FRNOM(N2I),FRTOL(N2I),1,0,FRTYP(N2I))
 3990 I = N2I
      ASSIGN 3999 TO K
      GO TO 3720
 3999 CONTINUE
C
C     MERGE THE INNER AND OUTER LEVELS.    4000 TO 4099
C
      S1M = .375
C     S1M IS THE DISTANCE BETWEEN LEVELS OF DIMENSIONING (IN INCHES).
      S2M = .40/SCALE
C     S2M IS THE MIN. ALLOWABLE APPROACH OF DIMENSION LINES TO
C     PART OUTLINE IN DATA UNITS.
C COUNT THE NUMBER OF LEVELS OF DIMENSIONING REQUIRED.
      NLEV(1) = 0
      NLEV(2) = 0
      DO 4020 I = 1,NPLP1
      DO 4010 L=1,2
```

```
 4010 NLEV(L) = MAX0(NLEV(L),KCF(L,I)+IFIX((ABC(L,I)+S2M)*SCALE/S1M))
 4020 CONTINUE
      NPW = IFIX(PWIDTH/S1M)
      IF(NLEV(1) + NLEV(2) + 2 .LE. NPW) GO TO 4050
      P = S1M * FLOAT(NLEV(1)+NLEV(2)+2-NPW)
      WRITE(6,4030) P
 4030 FORMAT(30H0 PAPER WIDTH IS TOO NARROW BY,F6.1,8H INCHES.)
      GO TO 9000
 4050 IF(NLEV(1).LT.(NPW-1)/2) GO TO 4060
      YAXIS = S1M * FLOAT(NPW-NLEV(1)-1)
      GO TO 4099
 4060 IF(NLEV(2).LT.(NPW-1)/2) GO TO 4070
      YAXIS = S1M * FLOAT(NLEV(2)+1)
      GO TO 4099
 4070 YAXIS = PWIDTH * .5
 4099 CONTINUE
C
C    DRAW THE OUTLINE OF THE SHAFT        5001  TO  5999
C
 5000 IF(PLTO) GO TO 5002
      WRITE (6,5001)
 5001 FORMAT(49H0      A PLOT IS NOT BEING PRODUCED    --  NO PLOT.///)
      GO TO 9000
C    PLTSW IS TURNED TRUE ONLY IF THERE IS SOME PLOTTING DONE.
 5002 IF(.NOT.PLTSW)  CALL IDPLOT
      PLTSW = .TRUE.
      CALL PLTIN(1./SCALE, 1./SCALE, -10.       /SCALE, -YAXIS/SCALE,
     1-10.      /SCALE, POSIT(NPLANE)+10.       /SCALE, -YAXIS/SCALE,
     2(PWIDTH-YAXIS)/SCALE)
      CALL FORMAT
      DEV = .01
      WLY(1) = DNOM(2)*.5
      CALL PLTLN(0.0,WLY(1),0.0,-WLY(1))
      DO 5999 I=2,NPLANE
C
C    SET UP DRAWING SECTION  5100  TO  5499
C
      X4 = PL(I)
      X5 = PR(I)
      IF(FRNOM(I).NE.0.0) GO TO 5140
      IF(TANG(I)) 5145,5140,5120
C    CHAMFER SECTION TO THE LEFT OF PLANE I   5120
 5120 IF(I.LT.NPLANE) GO TO 5125
      WRITE (6,5122)
 5122 FORMAT(53H1  THE LAST PLANE CANNOT HAVE A POSITIVE TAPER ANGLE.)
      GO TO 99990
 5125 Y4 = DNOM(I)*.5
      Y8 = DNOM(I+1)*.5
      Y5 = DNOM(I+1)*.5
      GO TO 5135
 5130 Y4 = DNOM(I-1)*.5
      Y5 = DNOM(I)*.5
      Y8 = DNOM(I)*.5
 5135 Y1 = 0.0
      Y3 = 0.0
      Y6 = 0.0
      X7 = POSIT(I)
      X8 = POSIT(I)
```

```
      MFER = 0
C     MFER IS USED TO DETECT A RIGHT CHAMFER FROM ITS SHOULDER PLANE.
      GO TO 5500
 5140 IF(MFER) 5130,5150,5130
 5145 MFER = 1
 5150 IF(TANG(I-1).LT.0.0) GO TO 5250
C     NO LEFT TAPER
      Y1 = 0.0
      Y4 = DNOM(I)*.5
      IF(DNOM(I).GT.DNOM(I-1)) GO TO 5190
C     LEFT HAND FILLET
      X2 = POSIT(I-1)
      Y3 = DNOM(I)*.5 + FRNOM(I-1)
      IF(DNOM(I-1)-DNOM(I).GT.FRNOM(I-1)*2.) GO TO 5200
C     LEFT HAND FILLET WILL BE LESS THAN 90 DEG. ARC
      Y2 = DNOM(I-1)*.5
      GO TO 5290
C     NO LEFT FILLET  5190
 5190 Y3 = 0.0
      GO TO 5300
C     FULL 90 DEG. LEFT FILLET   5200
 5200 X3 = X4
      Y2 = Y3
      GO TO 5300
C     LEFT HAND TAPER   5250
 5250 Y1 = DNOM(I-1)*.5
      X1 = POSIT(I-1)
      Y2 = DNOM(I)*.5 + FRNOM(I-1)*(1.-COS(TANG(I-1)))
      X2 = X1 - (Y1-Y2)/TAN(TANG(I-1))
      Y3 = DNOM(I)*.5 + FRNOM(I-1)
      Y4 = DNOM(I)*.5
 5290 X3 = X4
 5300 X8 = POSIT(I)
      Y5 = DNOM(I)*.5
      IF(TANG(I).GT.0.0) GO TO 5400
C     NO RIGHT HAND TAPER
      X7 = POSIT(I)
      IF(I.EQ.NPLANE) GO TO 5340
      IF(DNOM(I).LT.DNOM(I+1)) GO TO 5350
C     NO RIGHT HAND FILLET
 5340 Y6 = 0.0
      Y8 = DNOM(I)*.5
      X5 = POSIT(I)
      GO TO 5500
C     RIGHT HAND FILLET  5350
 5350 Y6 = DNOM(I)*.5 + FRNOM(I)
      Y8 = DNOM(I+1)*.5
      IF(DNOM(I+1)-DNOM(I).GE.FRNOM(I)*2.) GO TO 5370
C     RIGHT HAND FILLET WILL BE LESS THAN 90 DEG. ARC
      Y7 = DNOM(I+1)*.5
      GO TO 5450
C     RIGHT FILLET ARC IS FULL 90 DEG.
 5370 X6 = X5
      Y7 = Y6
      GO TO 5500
C     RIGHT HAND TAPER  5400
 5400 Y8 = DNOM(I+1)*.5
      Y7 = DNOM(I)*.5 + FRNOM(I)*(1.-COS(TANG(I)))
```

```
      Y6 = DNOM(I)*.5 + FRNOM(I)
      X7 = X8 - (Y8-Y7)/TAN(TANG(I))
 5450 X6 = X5
 5500 CONTINUE
 5600 IF(Y1.GT.0.0)CALL PLTLN(X1,Y1,X2,Y2)
      IF(Y3.GT.0.0) CALL PLTARC(X2,Y2,X4,Y4,X3,Y3,DEV)
      IF(X5.GT.X4) CALL PLTLN(X4,Y4,X5,Y5)
      IF(Y6.GT.0.0) CALL PLTARC(X5,Y5,X7,Y7,X6,Y6,DEV)
      IF(X8.GT.X7)  CALL PLTLN(X7,Y7,X8,Y8)
      CALL PLTLN(X8,Y8,X8,-Y8)
      IF(X8.GT.X7)  CALL PLTLN(X8,-Y8,X7,-Y7)
      IF(Y6.GT.0.0) CALL PLTARC(X7,-Y7,X5,-Y5,X6,-Y6,DEV)
      IF(X5.GT.X4)  CALL PLTLN(X5,-Y5,X4,-Y4)
      IF(Y3.GT.0.0) CALL PLTARC(X4,-Y4,X2,-Y2,X3,-Y3,DEV)
      IF(Y1.GT.0.0) CALL PLTLN(X2,-Y2,X1,-Y1)
C   Y8 IS RECORDED IN WLY TO USE IN PLOTTING VERTICAL WITNESS LINES.
      WLY(I) = Y8
 5999 CONTINUE
C
C   PLOT THE ANGULAR DIMENSIONS          6000 TO 6099
C
      S1P = S1M/SCALE
C   S1P IS THE DISTANCE BETWEEN LEVELS OF DIMENSIONING IN DATA UNITS.
      DO 6099 N=2,NPLANE
      IF(YACL(N).EQ.0.0) GO TO 6099
C   ANGLE TO BE PLOTTED FOR PLANE N
      CALL ANGPLT(TANG(N),POSIT(N),YACL(N))
 6099 CONTINUE
C
C   PLOT VERTICAL WITNESS LINES    6100 TO 6199
C
      DO 6199 L=1,2
      Y = FLOAT(NLEV(L)+1)*S1P
C   Y IS THE DATA UNITS DISTANCE TO THE BASE LINE (L=1 FOR ABOVE.)
      S = FLOAT(1-2*(L-1))
C   S IS POSITIVE WHEN ABOVE, NEGATIVE WHEN BELOW.
      DO 6180 I=1,NPLANE
      IF(.NOT.LAB(L,I)) GO TO 6180
      DO 6150 K=2,16,2
      IF(WL(L,K,I).EQ.0.0) GO TO 6170
      CALL PLTLN(POSIT(I),(Y-WL(L,K-1,I))*S,POSIT(I),(Y-WL(L,K,I))*S)
 6150 CONTINUE
      K = 18
C   PLOT THE SEGMENT OF THE WITNESS LINE CLOSEST TO THE SHAFT.
 6170 CALL PLTLN(POSIT(I),(Y-WL(L,K-1,I))*S,POSIT(I),(WLY(I)+S2L)*S)
 6180 CONTINUE
 6199 CONTINUE
C
C   PLOT HORIZONTAL WITNESS LINES  6200 TO 6299
C
      DO 6290 I=1,30
      IF(WLH(I,1).EQ.0.0) GO TO 6299
      CALL PLTLN(WLH(I,2),-WLH(I,1),WLH(I,3),-WLH(I,1))
 6290 CALL PLTLN(WLH(I,2),WLH(I,1),WLH(I,3),WLH(I,1))
 6299 CONTINUE
C
C   PLOT THE LENGTH DIMENSIONS   6300 TO 6499
C
```

```
      DO 6499 L=1,2
C     PLOT THE LOCATING DIMENSIONS (L=1) BEFORE THE REFERENCE DIMENSIONS.
      J = L-1
      DO 6490 I=2,NPLANE
      IF(L.EQ.2) GO TO 6310
      IF(RTOL(I).LT.(-10.)) GO TO 6490
      IR = IREF(I)
      A = ABS(RNOM(I))
      ATOL = RTOL(I)
      GO TO 6320
C     TEST FOR LACK OF REFERENCE DIMENSION.
 6310 IF(IREFQ(I).EQ.I) GO TO 6490
      IR = IREFQ(I)
      A = ABS(POSIT(I) - POSIT(IR))
      ATOL = -4.0
 6320 NL = MINO(IR,I)
      NR = MAXO(IR,I)
      IF(IH(I,L)) 6330,6330,6340
C     IH IS NEGATIVE FOR DIMENSIONS PLOTTED BELOW THE SHAFT.
 6330 Y = FLOAT(NLEV(2)+1+IH(I,L)) * (-S1P)
      GO TO 6350
 6340 Y = FLOAT(NLEV(1)+1-IH(I,L)) * S1P
 6350 IF(LCODE(I,L).GT.1) GO TO 6380
C     INTERNAL ARROWS AND WRITING.
      CALL ARROW(POSIT(NL),Y,0)
      XS = POSIT(NL) + (POSIT(NR)-POSIT(NL)-CLEN(A,ATOL,0,J,.FALSE.)
     1/SCALE)*.5
      CALL PLTPT(POSIT(NL),Y)
      CALL PLTLN(POSIT(NL),Y,XS,Y)
      CALL ALLPLT(XS,Y,A,ATOL,0,J,.FALSE.,XF)
      CALL PLTLN(XF,Y,POSIT(NR),Y)
      CALL ARROW(POSIT(NR),Y,4)
      GO TO 6490
 6380 IF(LCODE(I,L)-3) 6400,6420,6460
C     INTERNAL WRITING AND EXTERNAL ARROWS.      6400
 6400 CALL PLTLN(POSIT(NL)-S3L,Y,POSIT(NL),Y)
      CALL ARROW(POSIT(NL),Y,4)
      CALL ALLPLT(POSIT(NL)+(POSIT(NR)-POSIT(NL)-CLEN(A,ATOL,0,J,.FALSE.
     1)/SCALE)*.5,Y,A,ATOL,0,J,.FALSE.,XF)
      CALL ARROW(POSIT(NR),Y,0)
      CALL PLTLN(POSIT(NR),Y,POSIT(NR)+S3L,Y)
      GO TO 6490
C     INTERNAL ARROW, AND EXTERNAL WRITING.    6420
 6420 CALL ARROW(POSIT(NL),Y,0)
      CALL ARROW(POSIT(NR),Y,4)
      IF(LOH(I,L).LT.0) GO TO 6440
C     WRITING IS ON THE RIGHT HAND
      CALL PLTLN(POSIT(NL),Y,POSIT(NR)+S1L,Y)
      CALL ALLPLT(POSIT(NR)+S1L,Y,A,ATOL,0,J,.FALSE.,XF)
      GO TO 6490
C     WRITING IS ON THE LEFT HAND.   6440
 6440 CALL ALLPLT(POSIT(NL)-S1L-CLEN(A,ATOL,0,J,.FALSE.)/SCALE,Y,A,ATOL,
     10,J,.FALSE.,XF)
      CALL PLTLN(XF,Y,POSIT(NR),Y)
      GO TO 6490
C     EXTERNAL ARROWS AND WRITING.   6460
 6460 CALL PLTLN(POSIT(NL)-S2L,Y,POSIT(NL),Y)
      CALL ARROW(POSIT(NL),Y,4)
```

```
      CALL ARROW(POSIT(NR),Y,0)
      CALL PLTLN(POSIT(NR)+S2L,Y,POSIT(NR),Y)
      IF(LOH(I,L).LT.0) GO TO 6480
C   RIGHT HAND WRITING
      CALL ALLPLT(POSIT(NR)+S2L,Y,A,ATOL,0,J,.FALSE.,XF)
      GO TO 6490
C   LEFT HAND WRITING.   6480
 6480 CALL ALLPLT(POSIT(NL)-S2L-CLEN(A,ATOL,0,J,.FALSE.)/SCALE,Y,A,ATOL,
     10,J,.FALSE.,XF)
 6490 CONTINUE
 6499 CONTINUE
C
C   PLOT DIAMETRAL DIMENSIONS.     6500 TO 6599
C
      DO 6599 I=2,NPLANE
      IF(DL(I,2))6510,6599,6520
C   DIMENSION WRITTEN BELOW.       6510
 6510 Y = (FLOAT(NLEV(2)+1)+DL(I,2))*(-S1P)
      S = .5
      GO TO 6530
C   DIMENSION WRITTEN ABOVE.  6520
 6520 Y = (FLOAT(NLEV(1)+1)-DL(I,2)) * S1P
      S = -.5
 6530 CALL ARROW(DL(I,1),DNOM(I)*.5,6)
      CALL ARROW(DL(I,1),-DNOM(I)*.5,2)
      CALL PLTLN(DL(I,1),DNOM(I)*S,DL(I,1),Y)
      IF(DL(I,3).GT.0.0) GO TO 6570
C   LEFT HAND WRITING.
      XS = DL(I,1) - S2D - CLEN(DNOM(I),DTOL(I),2,0,DTYP(I))/SCALE
      CALL ALLPLT(XS,Y,DNOM(I),DTOL(I),2,0,DTYP(I),XF)
      CALL PLTPT(XF,Y)
      CALL PLTLN(XF,Y,DL(I,1),Y)
      GO TO 6599
C   RIGHT HAND WRITING.  6570
 6570 XS = DL(I,1) + S2D
      CALL PLTLN(DL(I,1),Y,XS,Y)
      CALL ALLPLT(XS,Y,DNOM(I),DTOL(I),2,0,DTYP(I),XF)
 6599 CONTINUE
C
C   PLOT NOTES WITH LEADERS   6600  TO  6699
C
      DO 6699 I=2,NPLANE
      IF(DN(I,4).EQ.0.0) GO TO 6699
      S1 = SIGN(1.,DN(I,1))
C   S1 IS POSITIVE FOR A RIGHT POINTING LEADER.
      S2 = SIGN(1.,DN(I,3))
C   S2 IS POSITIVE FOR DIMENSION PLACED ABOVE.
      J = IFIX(4.5 - S2*(2.-S1))
C   J IS THE CODE FOR ARROW DIRECTION (1 = SW, 3 = SE, 5 = NE, 7 = NW).
      XS = DN(I,1) * S1
      YS = DN(I,2) * S2
      CALL ARROW(XS,YS,J)
      XF = XS - S1*(DN(I,3)*S2 - DN(I,2))
      CALL PLTLN(XS,YS,XF,DN(I,3))
      XD = XF - S1N*S1
      CALL PLTLN(XF,DN(I,3),XD,DN(I,3))
      IF(DN(I,4).LT.1.5) GO TO 6660
C   THE PRESENT NOTE IS A FILLET RADIUS.
```

```
      IF(S1.GT.0.0) XD = XD-CLEN(FRNOM(I),FRTOL(I),1,0,FRTYP(I))/SCALE
      CALL ALLPLT(XD,DN(I,3),FRNOM(I),FRTOL(I),1,0,FRTYP(I),XF)
      GO TO 6699
C   THE PRESENT NOTE IS A CHAMFER.        6660
 6660 CALL UNITTO(XD,DN(I,3),X,Y)
      IF(S1.GT.0.0) GO TO 6680
C   THIS IS A RIGHT CHAMFER(A LEFT POINTING LEADER).
      WIDTH = POSIT(I+1) - POSIT(I)
      GO TO 6690
C   THIS IS A LEFT CHAMFER(A RIGHT POINTING LEADER).   6680
 6680 X = X - 1.02
      WIDTH = POSIT(I) - POSIT(I-1)
 6690 CALL CHPLT(X,Y,.7854,WIDTH)
 6699 CONTINUE
C   END PLOT, MOVING PEN 20 INCHES PAST RIGHT END OF SHAFT.
 8000 CALL PLOT(POSIT(NPLANE)*SCALE + 20.        ,0.0,-3)
      WRITE (6,8005)
 8005 FORMAT(32H0        A PLOT IS BEING PRODUCED./////)
 9000 CONTINUE
      GO TO 99995
99990 WRITE (6,99991)
99991 FORMAT(61H0  A FATAL ERROR HAS BEEN COMMITTED AND THE PROGRAM ABHO
     1RTED.)
99995 WRITE (6,99996)
99996 FORMAT(109H0  I     POSIT      DNOM       DTOL      IREF    RNOM       RT
     1OL       FRNOM      FRTOL      TANG     IREFQ    DTYP   FRTYP /)
      DO 99997 I=2,NPLANE
99997 WRITE(6,99998)I,POSIT(I),DNOM(I),DTOL(I),IREF(I),RNOM(I),RTOL(I),
     1  FRNOM(I),FRTOL(I),TANG(I),IREFQ(I),DTYP(I),FRTYP(I)
99998 FORMAT (X,I3,3F10.4,3X,I2,4F10.4,F7.2,4X,I2,2(5X,L3)/)
99999 CONTINUE
      IF(LAST.EQ.1) GO TO 1
C   WRITE 999 TO END ALL PLOTTING (IF THERE HAS BEEN ANY PLOTTING).
      IF(PLTSW)  CALL PLOT(0.0,0.0,999)
      STOP
      END
```

CD TOT   1482

```
$IBFTC CLEN
      FUNCTION CLEN(ANOM,ATOL,I,J,TYP)
C     CALCULATES THE LENGTH(IN INCHES) OF THE WRITING(SIZE 2 = .12IN.HIGH)
C     THIS FUNCTION IS THE COMPANION TO SUBROUTINE ALLPLT
C     I IS THE TYPE CODE -- 0 FOR LENGTH, 1 FOR RADIUS, 2 FOR DIAMETER.
C     J IS ONE FOR REFERENCE DIMENSIONS, ZERO OTHERWISE.
C     ATOL MUST NOT BE -99.
C     TYP IS TRUE IF THIS IS A TYPICAL DIMENSION.
      LOGICAL TYP
      DO 10 L=1,3
      LM1 = L-1
      IF(ANOM.LT.10.**LM1) GO TO 20
   10 CONTINUE
      LM1 = 3
C     ANOM MUST BE LESS THAN 1000
   20 K = 9 + 7*LM1
      IF(I.EQ.1) GO TO 60
      IF(ATOL.LT.0.0) GO TO 30
      K = K + 69
      GO TO 40
C     CODE FOR NUMBER OF DECIMAL PLACES    30
   30 K = K + 7*IFIX(.5-ATOL)
   40 IF(J.EQ.1) GO TO 90
      IF(I.NE.2) GO TO 100
      K = K + 32
      GO TO 80
   60 IF(ATOL.GT.0.0) GO TO 70
      K = K + 7*IFIX(.5-ATOL) + 8
      GO TO 80
   70 K = K + 63
      IF(ANOM-ATOL.LT.1E-3) K=K+3
   80 IF(.NOT.TYP) GO TO 100
   90 K = K + 32
  100 CLEN = .02*FLOAT(K)
      RETURN
      END

                                    CD TOT    0036
```

```
$IBFTC ALLPLT
      SUBROUTINE ALLPLT(X,Y,ABOM,ATOL,I,J,TYP,XF)
C     PLOTS THE DIMENSION WRITING
C     THIS IS THE COMPANION TO FUNCTION CLEN AND CODES ARE THE SAME.
C     I IS THE TYPE CODE -- 0 FOR LENGTH, 1 FOR RADIUS, 2 FOR DIAMETER.
C     J IS ONE FOR REFERENCE DIMENSIONS, ZERO OTHERWISE.
C     ATOL MUST NOT BE -99.
C     TYP IS TRUE IF THIS IS A TYPICAL DIMENSION.
C     X AND Y GIVE THE LEFT HAND POINT AT MID-LETTER HEIGHT, IN DATA UNITS
C      WHERE WRITING BEGINS.
C     XF(OUTPUT) GIVES THE RIGHT HAND END OF THE WRITING.
      DIMENSION A(2)
      DATA  PM/077606060606O/
      LOGICAL TYP
      CALL UNITTO(X,Y,XS,YS)
      YS = YS - .06
      XS = XS + .08
      ANOM = ABOM
      IF(ANOM-ATOL.LT.1E-3) ANOM = 2.*ANOM
      IF(ANOM.LT.1.0) GO TO 100
C     WRITE THE NUMBERS TO THE LEFT OF THE DECIMAL IN ANOM(LT 1000)
      DO 25 L=1,3
      IF(ANOM.LT.10.**L) GO TO 50
   25 CONTINUE
C     IF ANOM .GE.1000 IT IS SET EQUAL TO 0.0 AND WRITTEN AS SUCH
      ANOM = 0.0
   50 CALL OCONVF(ANOM,A,12,5-L)
      CALL LETTER(L,2,0,XS,YS,A(2))
      XS = XS + .14*FLOAT(L)
C     SEPARATE THE RADIUSES FOR WHICH ONLY 3(MAXIMUM) DECIMAL PLACES ARE
C      WRITTEN   100
  100 IF(I.NE.1) GO TO 200
      IF(ATOL.LT.0.0) GO TO 110
      L = 3
      GO TO 120
C     ATOL GIVES CODE FOR NUMBER OF DECIMAL PLACES.    110
  110 L = IFIX(.5-ATOL)
  120 TEMP = ANOM + .5*10.**(-L)
      CALL OCONVF(TEMP,A,12,5)
      CALL LETTER(L+1,2,0,XS,YS,A(2))
      XS = XS + .10 + .14*FLOAT(L)
      IF(ATOL.LT.0.0) GO TO 190
      IF(ABOM-ATOL.GT.1E-3) GO TO 130
      CALL LETTER(4,2,0,XS,YS,4H MAX)
      XS = XS + .64
      GO TO 190
  130 CALL LETTER(1,2,0,XS,YS,PM)
      TEMP = ATOL + .0005
      CALL OCONVF(TEMP,A,12,5)
      CALL LETTER(4,2,0,XS+.16,YS,A(2))
      XS = XS + .70
  190 CALL LETTER(1,2,0,XS,YS,1HR)
      XS = XS + .16
      GO TO 350
C     DIAMETERS AND LENGTHS WITH UP TO 4 DECIMAL PLACES.  200
  200 IF(ATOL.LT.0.0) GO TO 210
      L = 4
      GO TO 220
  210 L = IFIX(.5-ATOL)
```

```
220 TEMP = ANOM + .5*10.**(-L)
    CALL OCONVF(TEMP,A,12,5)
    CALL LETTER(L+1,2,0,XS,YS,A(2))
    XS = XS + .10 + .14*FLOAT(L)
    IF(ATOL.LT.0.0) GO TO 300
    CALL LETTER(1,2,0,XS,YS,PM)
    TEMP = ATOL + .00005
    CALL OCONVF(TEMP,A,12,5)
    CALL LETTER(5,2,0,XS+.16,YS,A(2))
    XS = XS + .84
300 IF(I.NE.2) GO TO 375
    CALL LETTER(4,2,0,XS,YS,4H DIA)
    XS = XS + .64
350 IF(.NOT.TYP) GO TO 400
    CALL LETTER(4,2,0,XS,YS,4H TYP)
    XS = XS + .64
    GO TO 400
375 IF(J.EQ.0) GO TO 400
    CALL LETTER(4,2,0,XS,YS,4H REF)
    XS = XS + .64
400 CALL INCHTO(XS,YS,XF,DUMMY)
    RETURN
    END

                        CD TOT    0082
```

```
$IBFTC SKIP
      SUBROUTINE SKIP(JAB,I;K,GAP)
C     PROVIDES SKIPS IN WITNESS LINES RECORDED IN WL
      COMMON SCALE,DUMMY(3),WL(2,17,50)
      DO 10 L=2,16,2
      IF(WL(JAB,L ,I).EQ.0.0) GO TO 50
   10 CONTINUE
      WRITE(6,20)  JAB,I
   20 FORMAT(43H1  WL DIMENSION IS NOT SUFFICIENT FOR JAB =,I3,5H, I =,
     1 I3)
      RETURN
   50 WL(JAB,L,I) = (.375*FLOAT(K) - GAP)/SCALE
      WL(JAB,L+1,I)=(.375*FLOAT(K) + GAP)/SCALE
      RETURN
      END
```

CD TOT 0015

```
$IBFTC ANGPLT
      SUBROUTINE ANGPLT(TANG,X,Y)
C    PLOTS DIMENSION LINES, WITNESS LINES, AND NUMBERS FOR ANGLES
      COMMON /BLAST/ DEV
      S1 = TANG/ABS(TANG)
      S2 = Y/ABS(Y)
      CALL UNITTO (X,Y,XC,YC)
      AX = COS(TANG)*S1
      AY = S1*SIN(TANG)*S2
      CALL PLOT(XC+.1*AX,YC+.1*AY,3)
      CALL PLOT(XC+.87*AX,YC+.87*AY,2)
      IF(S1*TANG.LT..79) GO TO 10
C    INTERNAL ARROWS
      S3 = 1.0
      BX = COS(TANG*.5)*S1*.75+XC
      BY = S1*SIN(TANG*.5)*S2*.75 + YC
      CY = BY - .12*S2
      DY = BY - .05*S2
      CX = SQRT(.5625 - (CY-YC)**2)*S1 + XC
      DX = SQRT(.5625 - (DY-YC)**2)*S1 + XC
      IF(S1*S2) 50,50,60
C    EXTERNAL ARROWS    10
   10 S3 = -1.0
      BX = COS(TANG + S1*.3)*S1*.75 + XC
      BY = S1*SIN(TANG + S1*.3)*S2*.75 + YC
      CX = .92106*S1*.75 + XC
      CY = -.38942*S2*.75 + YC
      IF(S1*TANG.LT..35) GO TO 20
C    INTERNAL NUMBERS
      DX = COS(TANG*.5)*S1*.75 + XC
      DY = S1*SIN(TANG*.5)*S2*.75 + YC
      GO TO 30
C    EXTERNAL NUMBERS     20
   20 DX = BX
      DY = BY + S2*.07
   30 IF(S1*S2) 60,60,50
C    ARGUMENTS FOR ARC ARE ORDERED TO MAINTAIN COUNTER-CLOCKWISE DRAWING
   50 CALL ARC(XC+.75*AX,YC+.75*AY,BX,BY,XC,YC,DEV)
      CALL ARC(CX,CY,XC+.75*S1,YC,XC,YC,DEV)
      GO TO 70
   60 CALL ARC(BX,BY,XC+.75*AX,YC+.75*AY,XC,YC,DEV)
      CALL ARC(XC+.75*S1,YC,CX,CY,XC,YC,DEV)
   70 CALL ARROWA(XC+.75*S1,YC,3.1416-S2*S3*(1.5708-S1*.12))
      CALL ARROWA(XC+.75*AX,YC+.75*AY,3.1416+S2*(TANG+S3*(1.5708-S1*
     1 .12)))
      CALL CHPLT(DX-S1*.06-.24, DY-S2*.01,ABS(TANG),0.0)
      RETURN
      END

                            CD TOT    0048
```

```
$IBFTC CHPLT
      SUBROUTINE CHPLT(X,Y,ANG,WIDTH)
C    PLOTS THE LETTERS FOR AN ANGLE IF WIDTH .EQ. 0  AND THOSE FOR A
C     CHAMFER NOTE OTHERWISE.  ARGUMENTS IN PLOTTER UNITS INDICATE LEFT
C      MID-POINT OF 1ST WORD. ANGLE GIVEN IN RADIANS.
      DIMENSION C(2)
      XS = X + .08
      YS = Y - .06
      IF(WIDTH.EQ.0.0) GO TO 10
      TEMP = WIDTH
      CALL OCONVF(TEMP,C,9,2)
      CALL LETTER(3,2,0,XS,YS,C(2))
      CALL LETTER(1,2,0,XS+.38,YS,1HX)
      XS = XS + .54
   10 A = ANG*57.29578 + .5
      CALL OCONVF(A,C,12,3)
      CALL LETTER(2,2,0,XS,YS,C(2))
      CALL LETTER(1,1,0,XS+.28,YS+.06,1HO)
      RETURN
      END

                                    CD TOT    0020
```

```
$IBFTC ARROWA
      SUBROUTINE ARROWA(X,Y,A)
C     PLOTS AN ARROW HEAD POINTING IN DIRECTION A, WITH ITS POINT AT X,Y
C     X,Y - IN PLOTTER UNITS      A - RADIANS IN USUAL SENSE
      X1 = .15*COS(A+.1652244) + X
      X2 = .15*COS(A-.1652244) + X
      Y2 = .15*SIN(A-.1652244) + Y
      Y1 = .15*SIN(A+.1652244) + Y
      CALL ARRA(X1,Y1,X2,Y2,X,Y)
      RETURN
      END

                              CD TOT   0011
```

```
$IBFTC ARROW
      SUBROUTINE ARROW(X,Y,I)
C    DRAWS ARROW HEAD IN ONE OF EIGHT DIRECTIONS, ACCORDING TO I, WITH
C    ITS TIP AT POINT X,Y.
C    X,Y, IN DATA UNITS.  I=0 GIVES A LEFT-POINTING ARROW, 1 A SOUTHWEST,
C    2 A DOWNPOINTING,3 A SOUTHEAST-POINTING, ETC.
      CALL UNITTO(X,Y,XP,YP)
      IF(MOD(I,2)) 10,10,30
   10 IF(MOD(I,4).EQ.2) GO TO 20
C    I=ZERO OR FOUR
      X1 = XP + .07398*FLOAT(2-I)
      Y1 = YP + .02465
      Y2 = YP - .02465
      CALL ARRA(X1,Y1,X1,Y2,XP,YP)
      RETURN
C    I=TWO OR SIX  20
   20 Y1 = YP + .07398*FLOAT(4-I)
      X1 = XP + .02465
      X2 = XP - .02465
      CALL ARRA(X1,Y1,X2,Y1,XP,YP)
      RETURN
C    I IS ODD    30
   30 C = .15
      IF(I.GT.4) C=-.15
C    C IS POSITIVE FOR I = 1 OR 3
      Y1 = YP + C*.81378
      Y2 = YP + C*.58118
      IF(MOD(I+1,4).EQ.0) C=-C
C    C IS POSITIVE FOR I = 1 OR 7
      X1 = XP + C*.58118
      X2 = XP + C*.81378
      CALL ARRA(X1,Y1,X2,Y2,XP,YP)
      RETURN
      END
```

CD TOT 0034

```
$IBFTC ARRA
      SUBROUTINE ARRA(X1,Y1,X2,Y2,XP,YP)
      CALL PLOT(XP,YP,3)
      CALL PLOT(X1,Y1,2)
      CALL PLOT(XP,YP,3)
      CALL PLOT(X2,Y2,2)
      CALL PLOT(X1,Y1,2)
      RETURN
      END
```

CD TOT 0009

```
$IBFTC FORMAT
      SUBROUTINE FORMAT
      COMMON  SCALE,PWIDTH,SN(2)
      CALL PLOT(1.625,2.625,3)
      CALL PLOT(2.625,2.625,2)
      CALL PLOT(2.625,7.625,2)
      CALL PLOT(1.625,7.625,2)
      CALL PLOT(1.625,2.625,2)
      CALL PLOT(2.125,2.625,3)
      CALL PLOT(2.125,5.625,2)
      CALL PLOT(1.625,5.625,3)
      CALL PLOT(2.625,5.750,2)
      CALL PLOT(1.625,5.750,2)
      CALL PLOT(2.625,5.625,2)
      CALL PLOT(1.625,5.625,2)
      CALL PLOT(2.125,5.750,3)
      CALL PLOT(2.125,7.625,2)
      CALL OCONVF(SCALE,SC,6,3)
      CALL LETTER(10,4,90,2.0,3.0,10HSHAFT NAME)
      CALL LETTER( 5,4,90,2.0,6.125,5HSCALE)
      CALL LETTER(12,4,90,2.5,3.0,SN)
      CALL LETTER( 6,4,90,2.5,6.125,SC)
      RETURN
      END

                        CD TOT   0024
```

```
$IBFTC PARCJN
      SUBROUTINE PLTARC (X1,Y1,X2,Y2,XC,YC,DEV)
C     MODIFICATION OF PLTARC TO ALLOW FOR A CENTER PT OFF THE PAGE
      COMMON  SCALE,YAXIS
      CALL UNITTO (X1,Y1,XX1,YY1)
      CALL UNITTO (X2,Y2,XX2,YY2)
      XXC = XC*SCALE + 10.
      YYC = YC*SCALE  +  YAXIS
      CALL ARC (XX1,YY1,XX2,YY2,XXC,YYC,DEV)
      RETURN
      END

                              CD TOT   0011
```

APPENDIX III

SAMPLE SHAFTS -- INPUT AND OUTPUT

EXECUTIØN ØF PRØGRAM SHFTDW FØR SHAFT NUMBER 396143R1

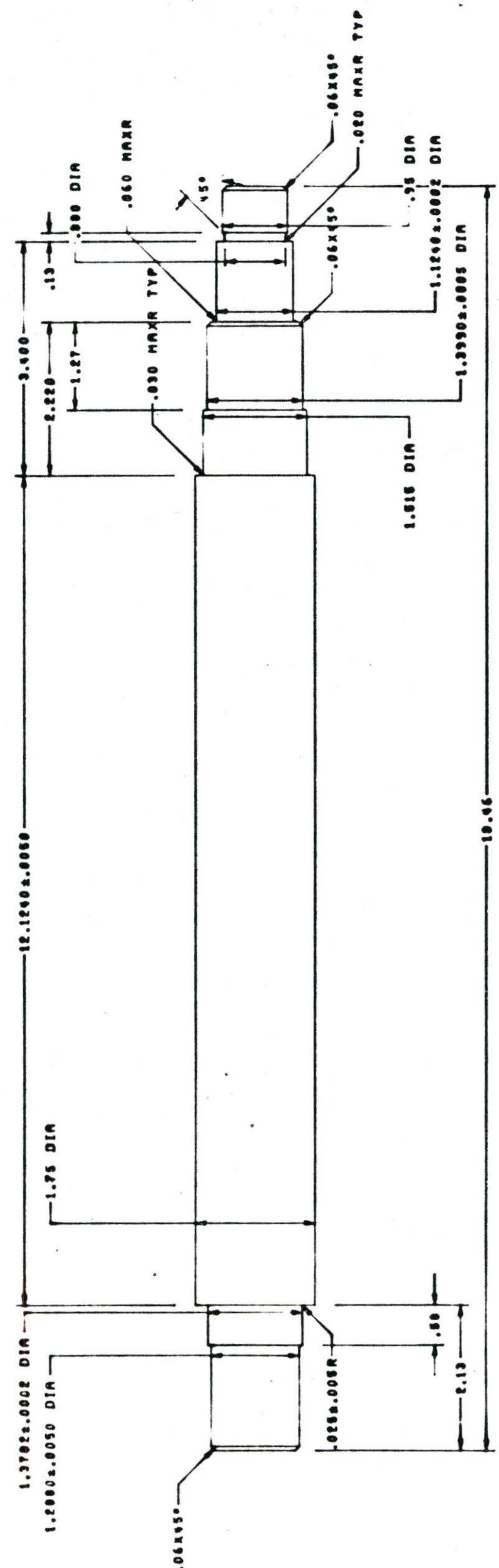
SCALE = 1.000
PAPER WIDTH FØR PLØT = 10.0
THE INPUT CALLS FØR A PLØT.

*** DATA AS READ ***

NUM	A1	A2	IREF	A3	A4	A5	A6	IREFQ	A7	LAST
1	-0.0000	-0.0000	0	0.0600	-2.0000	-0.0000	-0.0000	-0	45.0	*
2	1.2880	0.0050	3	-0.5800	-2.0000	-0.0000	-0.0000	-0	-0.0	*
3	1.3782	0.0002	0	2.1300	-2.0000	0.0250	0.0050	-0	-0.0	*
4	1.7500	-2.0000	3	12.1240	0.0050	0.0150	0.0150	-0	-0.0	*
5	1.5150	-3.0000	7	-1.2700	-2.0000	-0.0000	4.0000	-0	-0.0	*
6	-0.0000	-0.0000	7	-0.0600	-2.0000	-0.0000	-0.0000	-0	-45.0	*
7	1.3990	0.0005	4	2.2200	-3.0000	0.0300	0.0300	-0	-0.0	*
8	1.1240	0.0002	4	3.4000	-3.0000	0.0100	0.0100	-0	-0.0	*
9	0.8800	-3.0000	8	0.1300	-2.0000	-0.0000	8.0000	-0	45.0	*
10	-0.0000	-0.0000	11	-0.0600	-2.0000	-0.0000	-0.0000	-0	-45.0	*
11	0.9500	-2.0000	0	18.4600	-2.0000	-0.0000	-0.0000	-0	-0.0	2

A PLØT IS BEING PRØDUCED.

Cost = $2.56



396143R1

EXECUTIØN ØF PRØGRAM SHFTDW FØR SHAFT NUMBER 389529R1

SCALE = 1.000
PAPER WIDTH FØR PLØT = 15.0
THE INPUT CALLS FØR A PLØT.

*** DATA AS READ ***

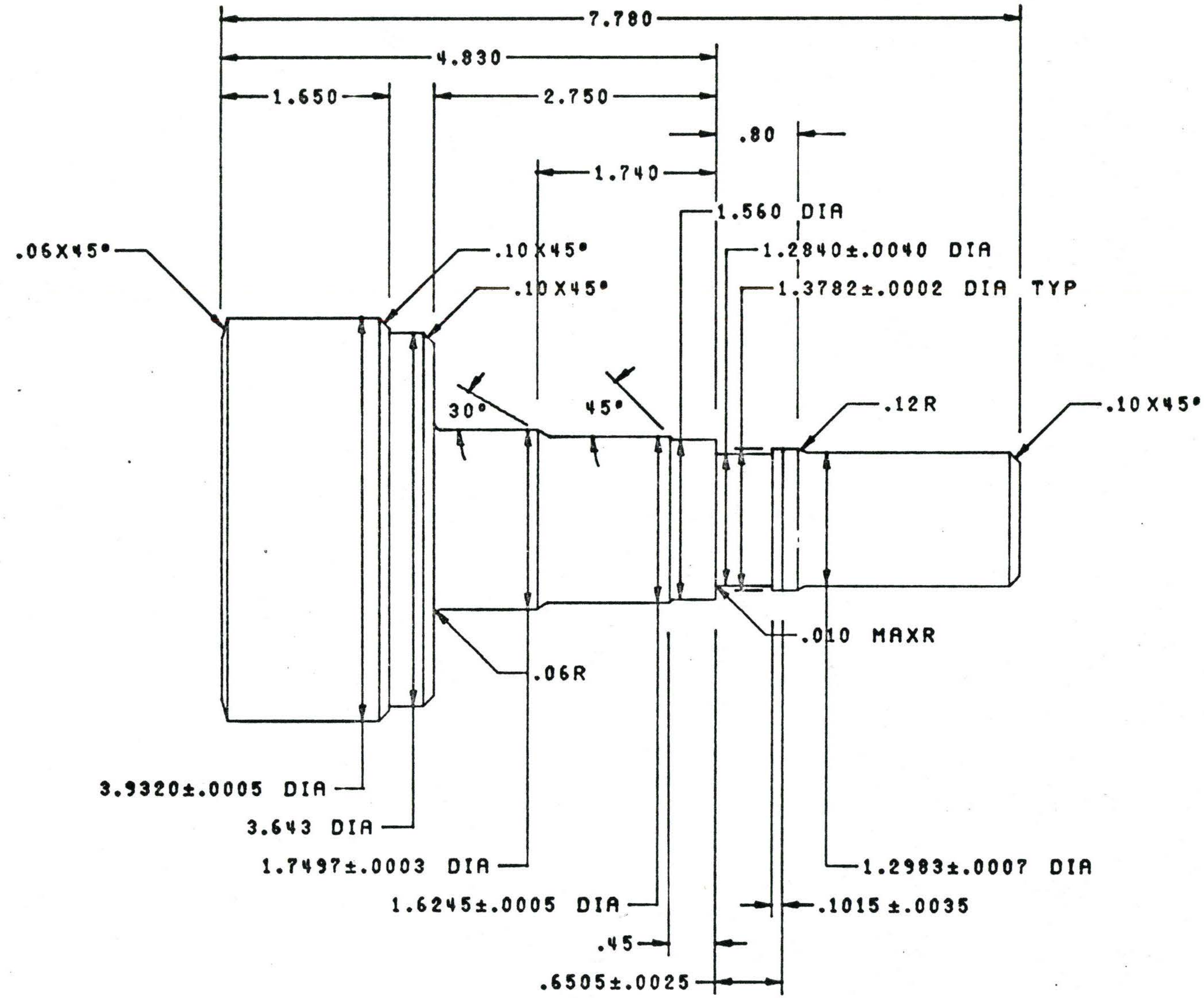
NUM	A1	A2	IREF	A3	A4	A5	A6	IREFQ	A7	LAST
1	1.5600	-3.0000	0	4.8300	-3.0000	0.0050	0.0050	-0	-0.0	*
2	1.2983	0.0007	0	7.7800	-3.0000	-0.0000	-0.0000	-0	-0.0	*
21	-0.0000	-0.0000	2	-0.1000	-2.0000	-0.0000	-0.0000	-0	-45.0	*
3	-0.0000	-0.0000	0	0.0600	-3.0000	-0.0000	-0.0000	-0	45.0	*
4	3.9320	0.0005	0	1.6500	-3.0000	-0.0000	-0.0000	-0	-0.0	*
5	3.6430	-99.0000	4	-0.1000	-2.0000	-0.0000	-0.0000	-0	-45.0	*
11	3.6430	-3.0000	1	-2.7500	-3.0000	0.0600	-2.0000	-0	-0.0	*
12	-0.0000	-0.0000	11	-0.1000	-2.0000	-0.0000	-0.0000	-0	-45.0	*
13	1.7497	0.0003	1	-1.7400	-3.0000	0.0010	*******	-0	-30.0	*
14	1.6245	0.0005	1	-0.4500	-2.0000	-0.0000	*******	-0	-45.0	*
15	1.3782	0.0002	1	0.6505	0.0025	0.0050	0.0050	-0	-0.0	*
16	1.2840	0.0040	15	-0.1015	0.0035	-0.0000	*******	-0	-0.0	*
17	-0.0000	15.0000	1	0.8000	-2.0000	0.1200	-2.0000	-0	-0.0	2

ERRØR CØNDITIØN AT 062374 FØRTRAN ERRØR 09 IGNØRED, RETURN TØ EXECUTIØN

ERRØR CØNDITIØN AT 062374 FØRTRAN ERRØR 09 IGNØRED, RETURN TØ EXECUTIØN

A PLØT IS BEING PRØDUCED.

Cost = $2.78


7.780
4.830
1.650
2.750
.80
1.740
1.560 DIA
.06X45°
.10X45°
1.2840±.0040 DIA
.10X45°
1.3782±.0002 DIA TYP
30°
45°
.12R
.10X45°
.010 MAXR
.06R
3.9320±.0005 DIA
3.643 DIA
1.7497±.0003 DIA
1.2983±.0007 DIA
1.6245±.0005 DIA
.1015±.0035
.45
.6505±.0025

389529R1

EXECUTIØN ØF PRØGRAM SHFTDW FØR SHAFT NUMBER 396733R1

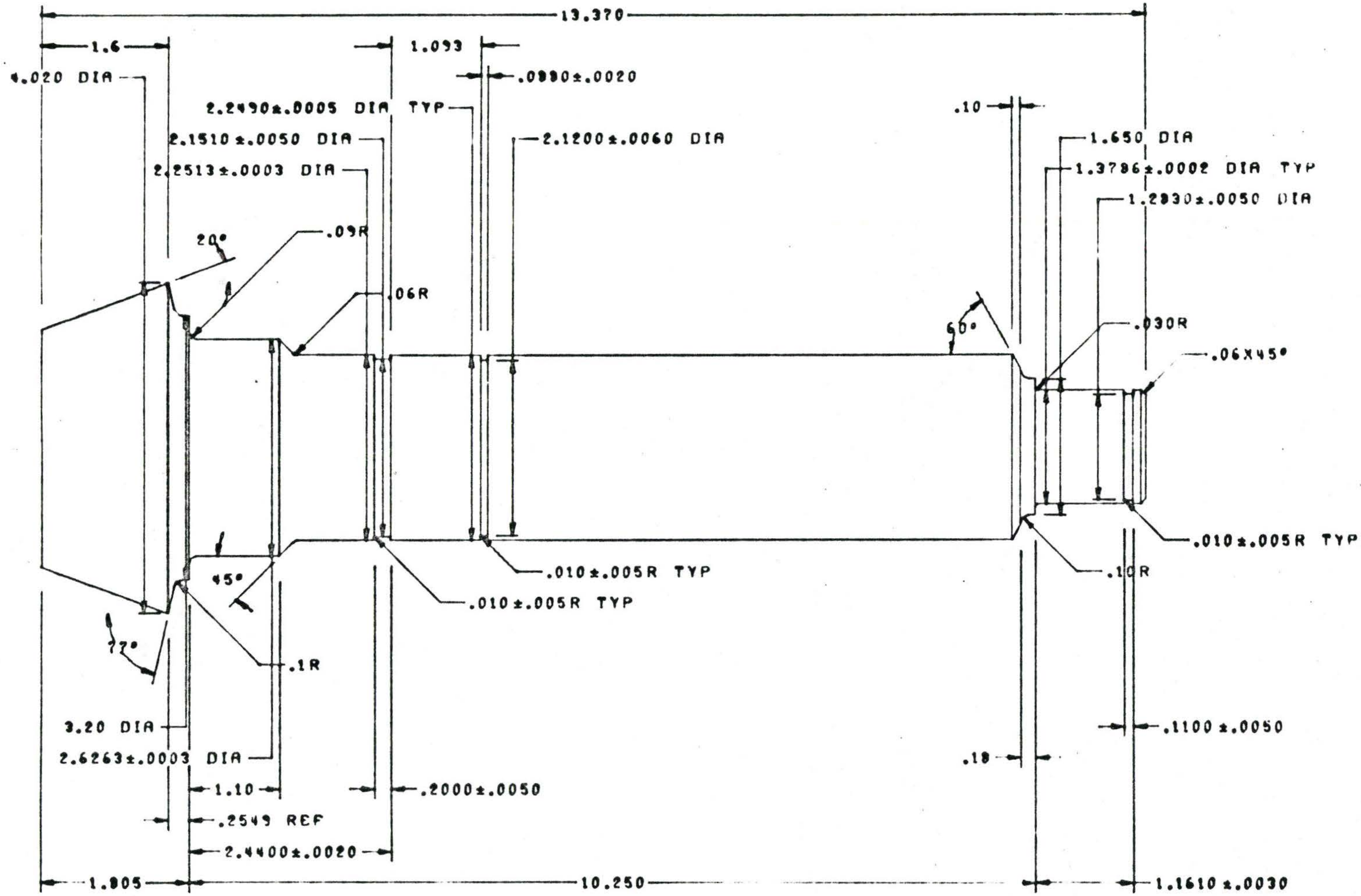
SCALE = 1.000
PAPER WIDTH FØR PLØT = 20.0
THE INPUT CALLS FØR A PLØT.

*** DATA AS READ ***

NUM	A1	A2	IREF	A3	A4	A5	A6	IREFQ	A7	LAST
1	-0.0000	-0.0000	0	1.5500	-1.0000	-0.0000	-0.0000	-0	20.0	*
2	4.0200	-3.0000	0	1.5501	-80.0000	0.1000	-1.0000	3	-77.0	*
3	3.2000	-2.0000	0	1.8050	-3.0000	0.0900	-2.0000	-0	-0.0	*
4	2.6263	0.0003	3	1.1000	-2.0000	0.0600	-2.0000	-0	-45.0	*
5	2.2513	0.0003	6	-0.2000	0.0050	0.0100	0.0050	-0	-0.0	*
6	2.1510	0.0050	3	2.4400	0.0020	-0.0000	5.0000	-0	-0.0	*
7	2.2490	0.0005	6	1.0930	-3.0000	0.0100	0.0050	-0	-0.0	*
8	2.1200	0.0060	7	0.0880	0.0020	-0.0000	7.0000	-0	-0.0	*
9	-0.0000	-0.0000	10	-0.1000	-2.0000	-0.0000	-0.0000	-0	-60.0	*
10	-0.0000	7.0000	11	-0.1800	-2.0000	0.1000	-2.0000	-0	-0.0	*
11	1.6500	-3.0000	3	10.2500	-3.0000	0.0300	-3.0000	-0	-0.0	*
12	1.3786	0.0002	13	-0.1100	0.0050	0.0100	0.0050	-0	-0.0	*
13	1.2830	0.0050	11	1.1610	0.0030	-0.0000	*******	-0	-0.0	*
14	-0.0000	-0.0000	15	-0.0600	-2.0000	-0.0000	-0.0000	-0	-45.0	*
15	-0.0000	12.0000	0	13.3700	-3.0000	-0.0000	-0.0000	-0	-0.0	2

A PLØT IS BEING PRØDUCED.

Cost = $3.60


13.370
1.6
4.020 DIA
1.093
.0880±.0020
2.2490±.0005 DIA TYP
2.1510±.0050 DIA
2.1200±.0060 DIA
2.2513±.0003 DIA
.10
1.650 DIA
1.3786±.0002 DIA TYP
1.2830±.0050 DIA
20°
.09R
.06R
60°
.030R
.06X45°
.010±.005R TYP
.10R
.010±.005R TYP
.010±.005R TYP
45°
77°
.1R
3.20 DIA
2.6263±.0003 DIA
.1100±.0050
.18
1.10
.2000±.0050
.2549 REF
2.4400±.0020
1.805
10.250
1.1610±.0030

396733R1

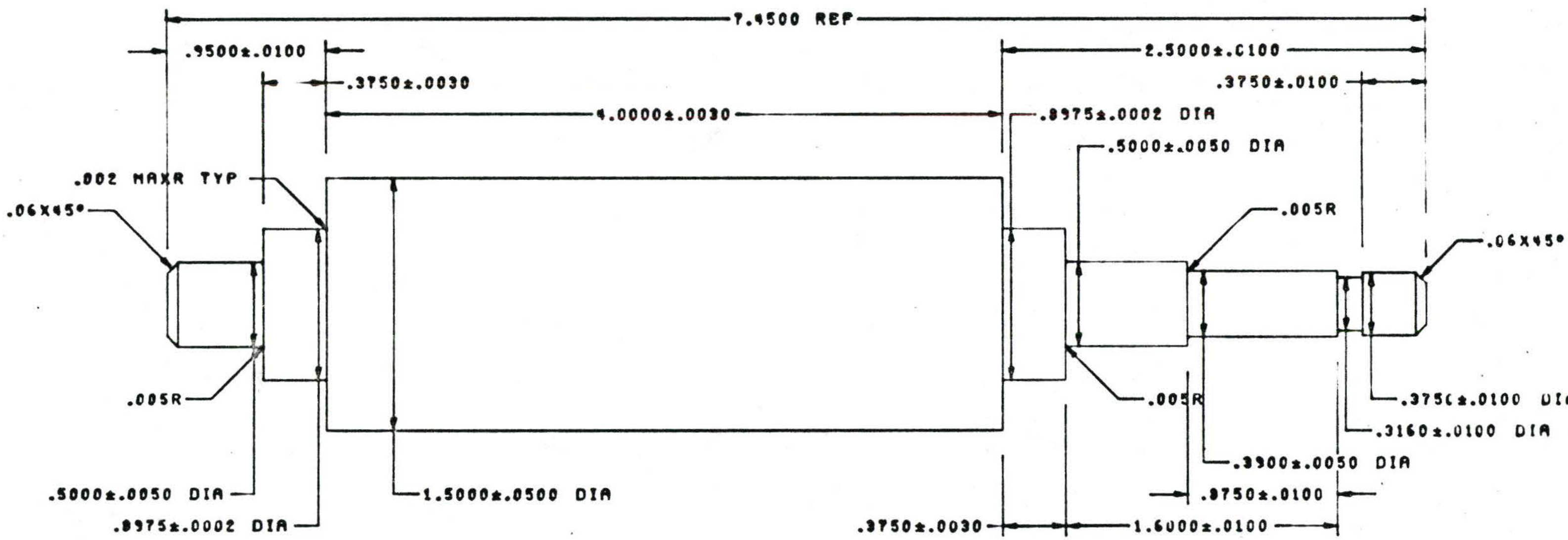
EXECUTIØN ØF PRØGRAM SHFTDW FØR SHAFT NUMBER KARDØS

SCALE = 2.000
PAPER WIDTH FØR PLØT = 10.0
THE INPUT CALLS FØR A PLØT.

*** DATA AS READ ***

NUM	A1	A2	IREF	A3	A4	A5	A6	IREFQ	A7	LAST
1	0.8975	0.0002	0	0.9500	0.0100	0.0010	0.0010	-0	-0.0	*
2	1.5000	0.0500	1	4.0000	0.0030	-0.0000	1.0000	-0	-0.0	*
3	0.3750	0.0100	2	2.5000	0.0100	-0.0000	-0.0000	-1	-0.0	*
4	0.5000	0.0050	1	-0.3750	0.0030	0.0050	-3.0000	-0	-0.0	*
5	0.8975	0.0002	2	0.3750	0.0030	0.0050	-3.0000	-0	-0.0	*
6	0.3900	0.0050	5	1.6000	0.0100	-0.0000	1.0000	-0	-0.0	*
7	0.5000	0.0050	6	-0.8750	0.0100	0.0050	-3.0000	-0	-0.0	*
8	0.3160	0.0100	3	-0.3750	0.0100	-0.0000	1.0000	-0	-0.0	*
9	-0.0000	-0.0000	3	-0.0625	0.0100	-0.0000	-0.0000	-0	-45.0	*
10	-0.0000	-0.0000	0	0.0625	0.0100	-0.0000	-0.0000	-0	45.0	2

A PLØT IS BEING PRØDUCED.

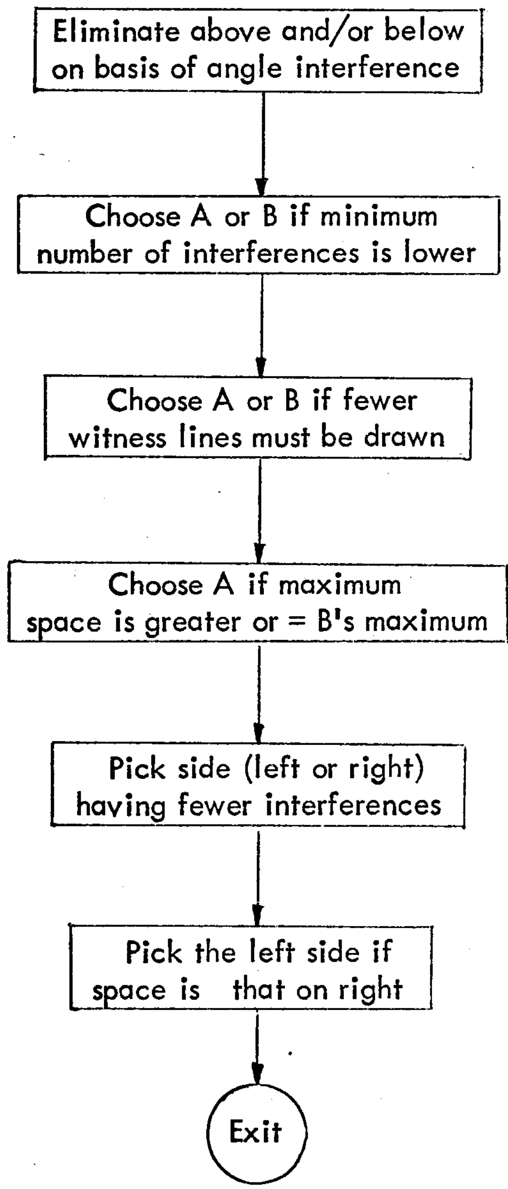

7.4500 REF
.9500±.0100
.3750±.0030
4.0000±.0030
2.5000±.C100
.3750±.0100
.8975±.0002 DIA
.5000±.0050 DIA
.002 MAXR TYP
.06X45°
.005R
.06X45°
.005R
.005R
.3750±.0100 DIA
.3160±.0100 DIA
.3900±.0050 DIA
.8750±.0100
.5000±.0050 DIA
1.5000±.0500 DIA
.8975±.0002 DIA
.3750±.0030
1.6000±.0100

KARDOS

APPENDIX IV

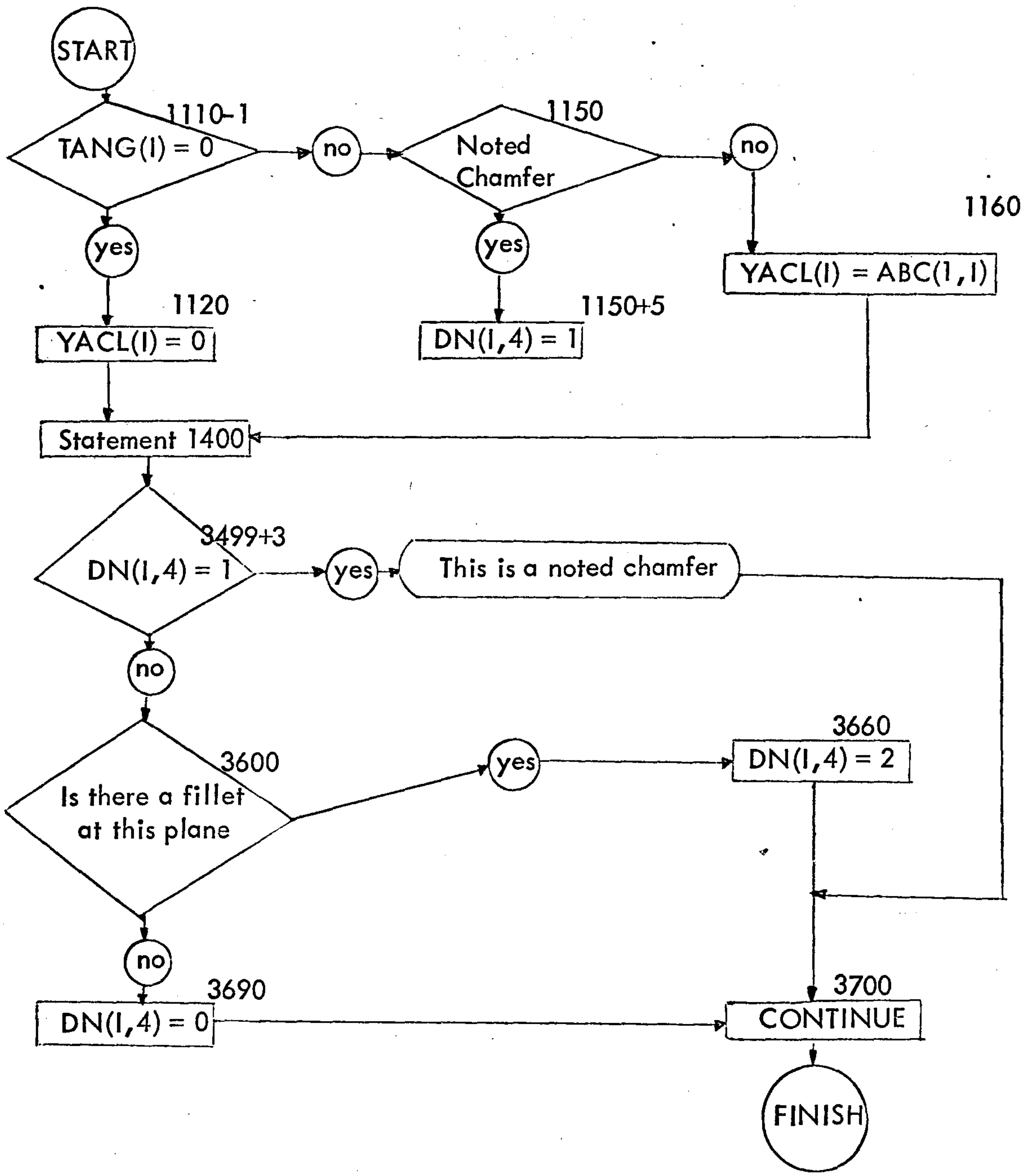
MACRO FLOWCHARTS

Length Dimension Location Logic (Paraphrased)



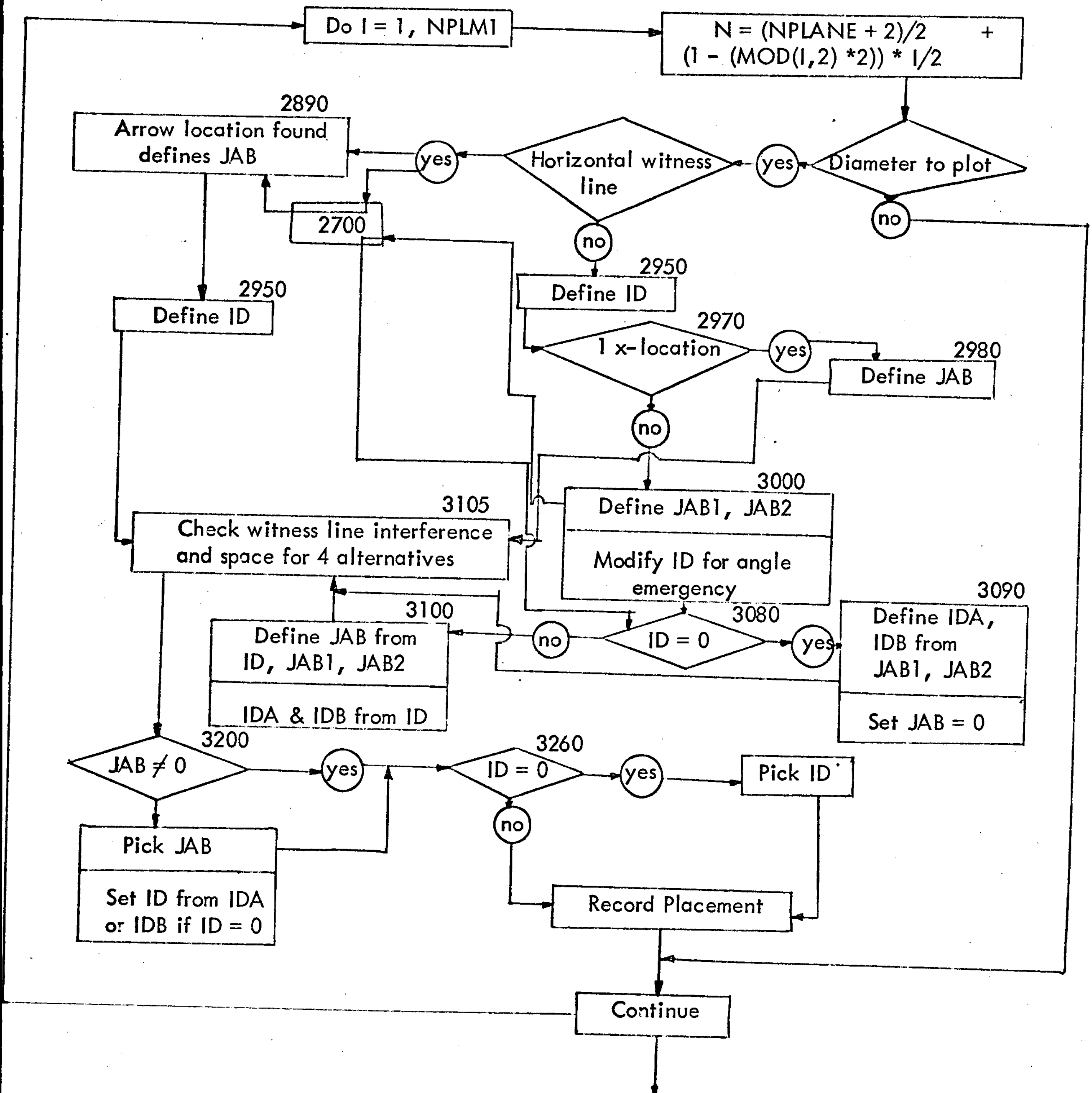
Overall Flowchart for Notes

(Position of each statement, relative to a numbered one, is given)



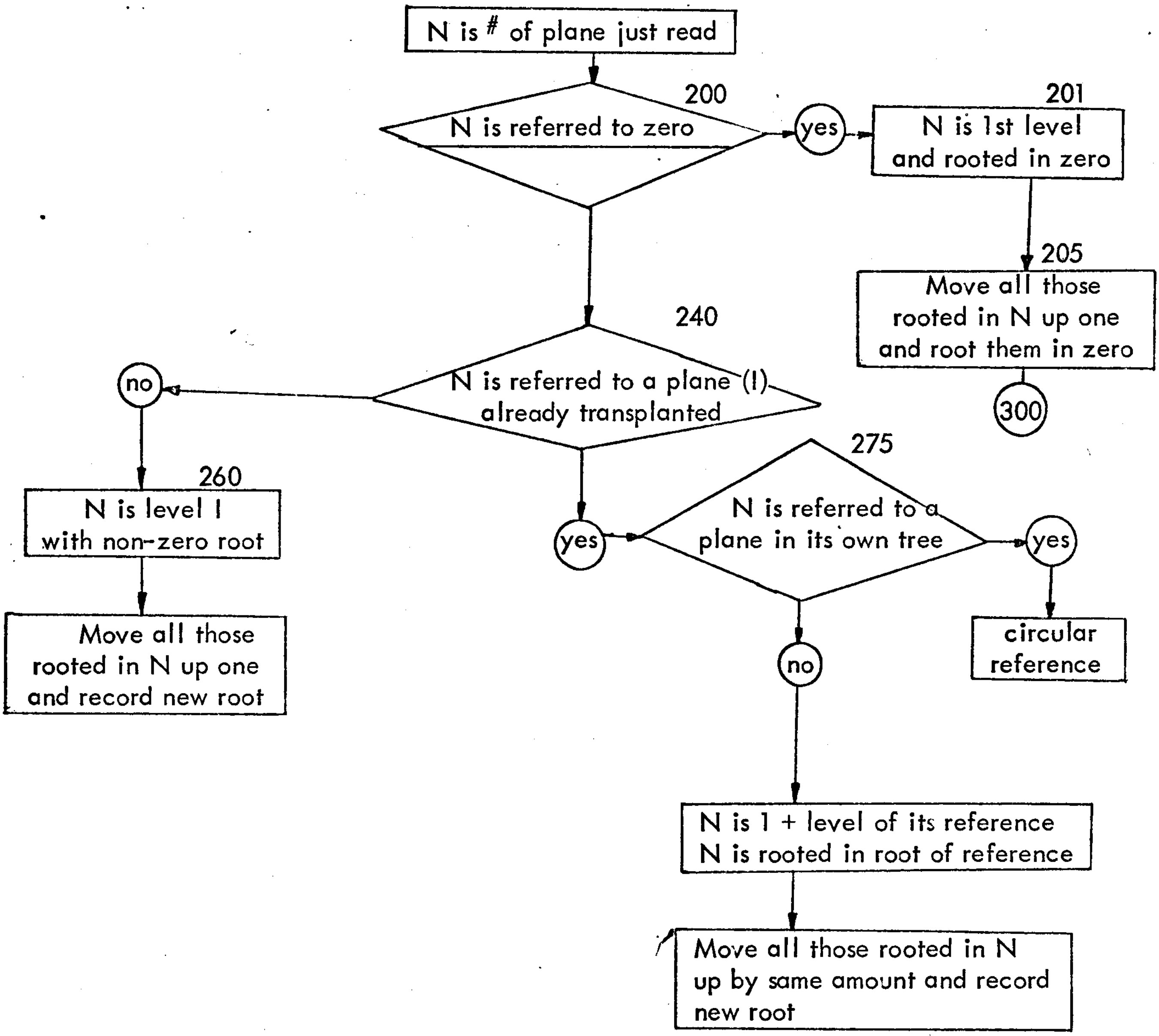
Revised LDDC (Paraphrased)

(Statement numbers are given)



"CHECK FOR CIRCULAR REFERENCES" (Paraphrased)

(Statement numbers are given)



Positions are all set negative to help in sorting for "RENUMBER"

APPENDIX V

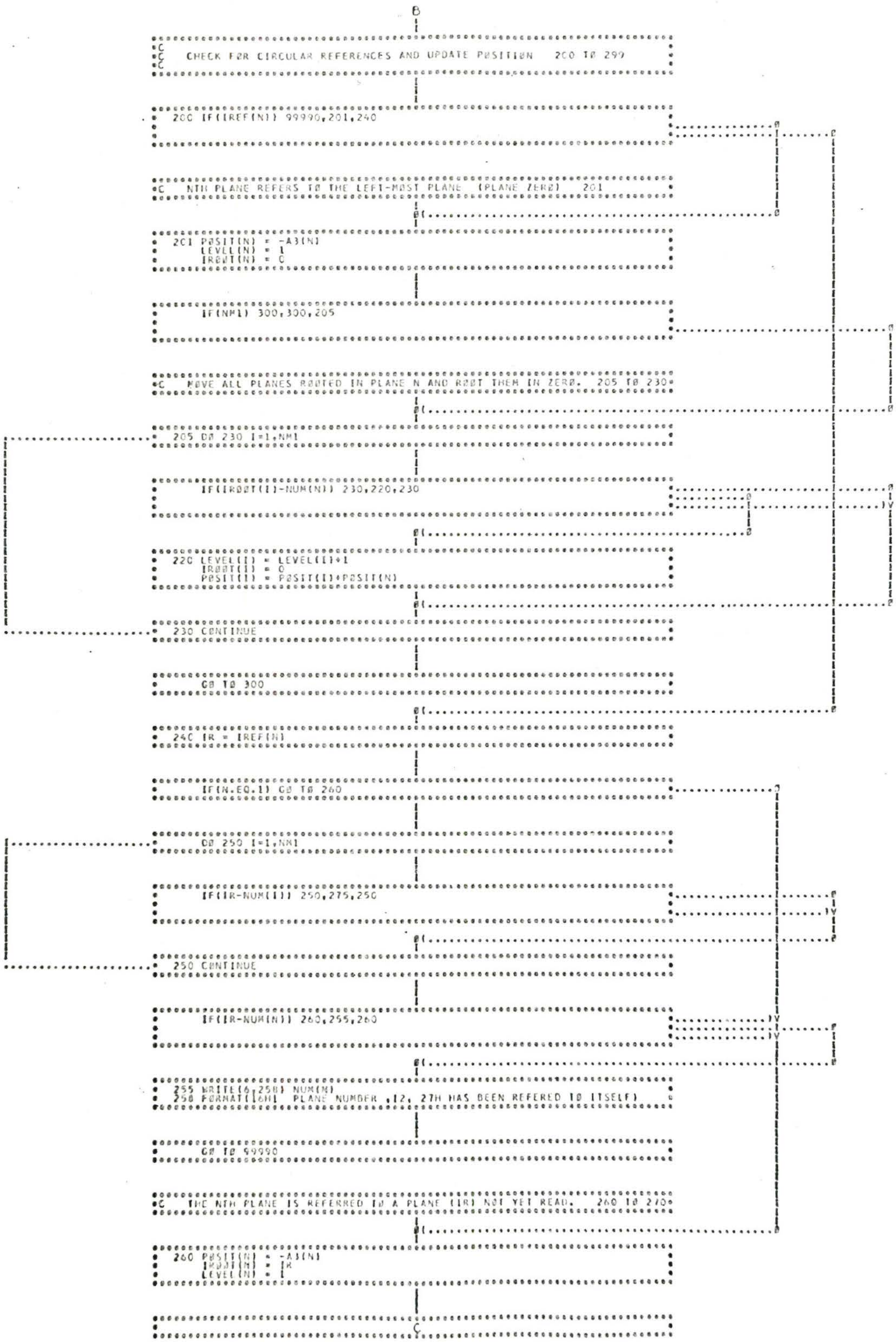
DETAIL FLOWCHARTS

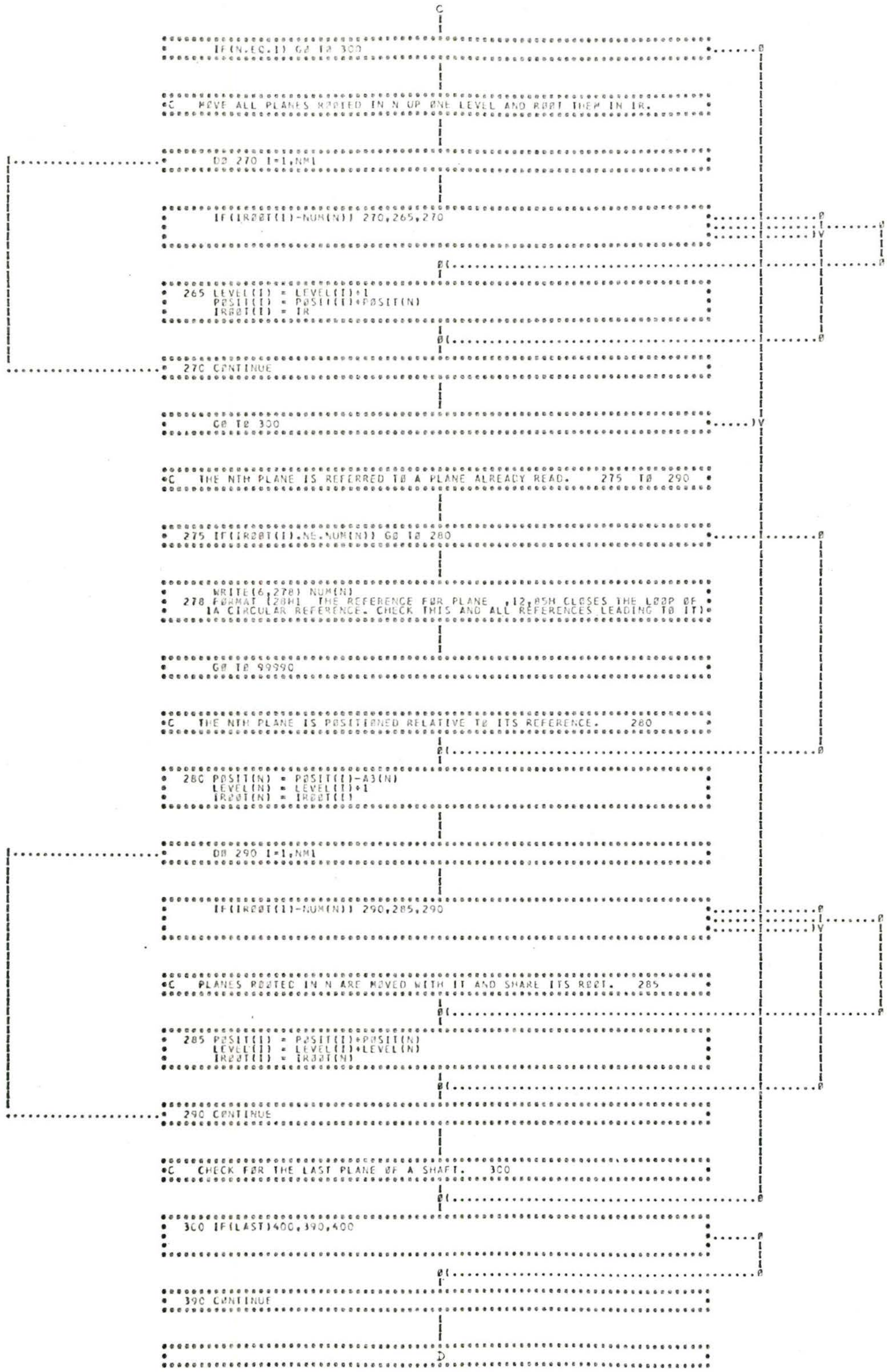
Index to Detail Flowcharts

(ENTRANCE)

```
C                                                                     DECK NØ.   105
      DIMENSIØN A1(50),A2(50),A3(50),A4(50),A5(50),A6(50),A7(50),NUM(50)
      DIMENSIØN IRØØT(50),PØSIT(50),LEVEL(50),IREF(50),IREFQ(50),A8(50)
      DIMENSIØN DNØM(50),DTØL(50),RNØM(50),RTØL(50),FRNØM(50),FRTØL(50)
      DIMENSIØN TANG(50),II(100),JJ(50),KK(50),PL(50),PR(50),KCF(2,51)
      CØMMØN  SCALE,YAXIS
      CØMMØN  SN(2)
      CØMMØN WL(2,17,50)
      CØMMØN /BLAST/ DEV
      DIMENSIØN ABC(2,51), DL(51,3),YACL(50),DN(50,4),PADK(2,20)
      DIMENSIØN           WLH(30,3),KAB1(2),KAB2(2),M1(2),M2(2),M3(2)
      DIMENSIØN IH(50,2), LØH(50,2), LCØDE(50,2), WLY(50), NLEV(2)
      LØGICAL DTYP(50),FRTYP(50),LAB(2,51),PLTØ,LDUM,PLTSW
      DATA PLTSW/.FALSE./
    1 READ (5,5) SN,SCALE,PWIDTH,PLTØ
    5 FØRMAT (2A6,F11.3,F8.1,48X,L1)
      WRITE (6,10) SN,SCALE,PWIDTH
   10 FØRMAT(48H1  EXECUTIØN ØF PRØGRAM SHFTDW FØR SHAFT NUMBER ,2A6// 9
     1X,8H SCALE =,F6.3/9X,23H PAPER WIDTH FØR PLØT =,F5.1)

      IF(.NØT.PLTØ) GØ TØ 20

      WRITE(6,15)
   15 FØRMAT(32H      THE INPUT CALLS FØR A PLØT.)

      GØ TØ 30

   20 WRITE(6,25)
   25 FØRMAT(39H        THE INPUT DØESN'T CALL FØR A PLØT)
C    ZERØ ARRAYS  (IMPØRTANT FØR RUNS WITH MØRE THAN ØNE SHAFT)  30 TØ 40

   30 N = 1

   32 IRØØT(N) = N
      PØSIT(N) = 0.0

      DØ 33 L=1,2

      DØ 33 M=1,17

   33 WL(L,M,N) = 0.0

   34 DØ 36 L=1,2

      KCF(L,N) = 0
      LAB(L,N) = .FALSE.

   36 DL(N,L) = 0.0

      N = N + 1

      IF(N-51) 32,34,40

   40 WRITE (6,42)
   42 FØRMAT(///40X,20H*** DATA AS READ ***///4H NUM,6X,2HA1,8X,2HA2,3X,
     1 4HIREF,6X,2HA3,9X,2HA4,6X,2HA5,7X,2HA6,7H  IREFQ,3X,2HA7,3X,
     2 4HLAST/)

                                      A
```

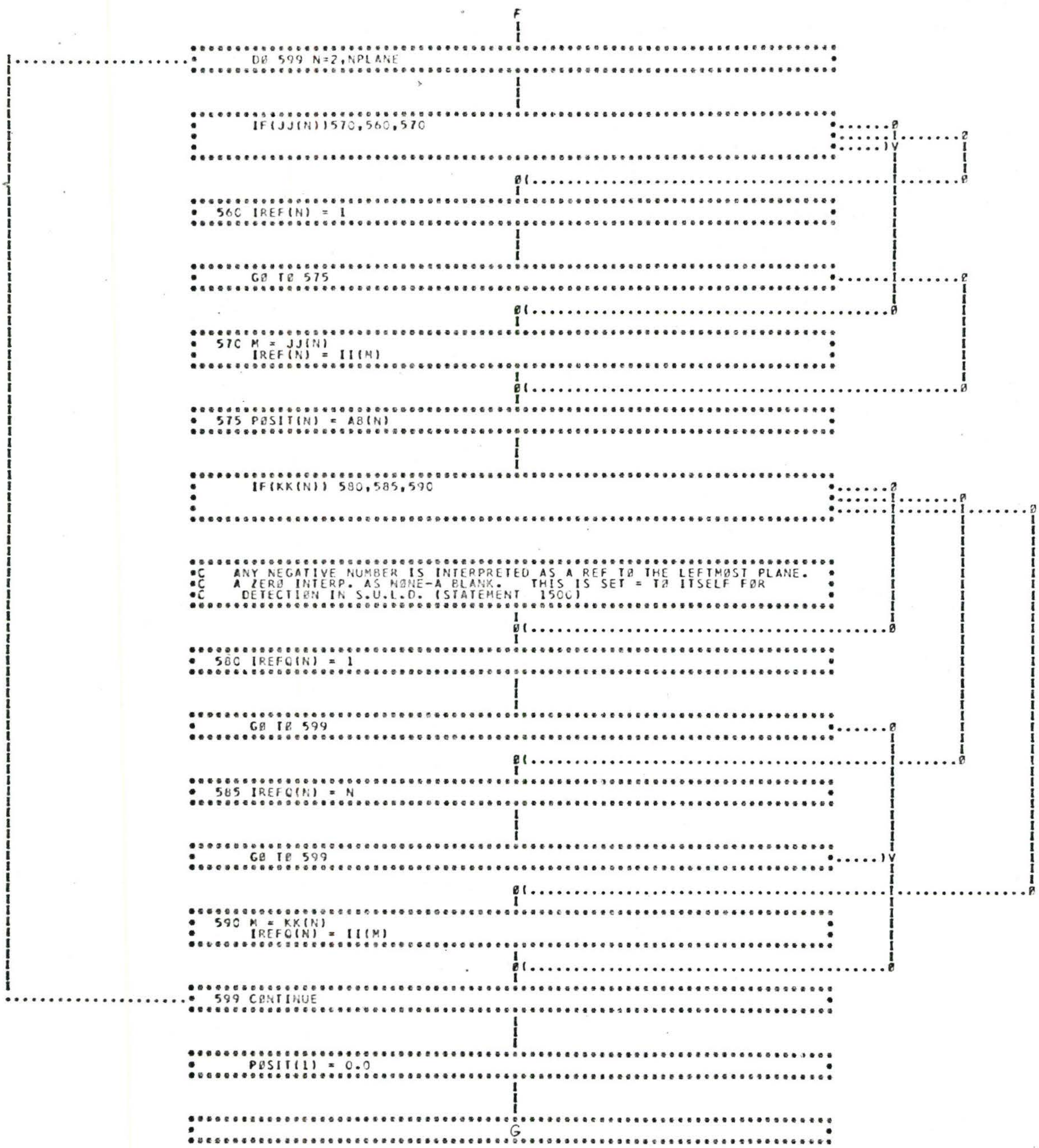
```
A

C
C    INPUT LØØP 00045 TØ 00390
C

      DØ 390 N=1,49

      NM1 = N-1
      READ (5,49) NUM(N),A1(N),A2(N),IREF(N),A3(N),A4(N),A5(N),A6(N),
     1 IREFO(N),A7(N),LAST
   49 FØRMAT (I2,2X,2(F8.4,2X),I2,2X,2(F9.4,2X),2(F7.4,2X),I2,2X,F5.1,
     1 2X,I1)
CB-+2BB02BBB-+22202BBB-+2B-+222202BBB-+2BB202BBB-+2B02BBB-+22202BBB-+2B-+
C    ABØVE IS THE DRUM CARD FØR FØRMAT 49 (PUT A 2 IN CØL. 1)
      WRITE(6,59) NUM(N),A1(N),A2(N),IREF(N),A3(N),A4(N),A5(N),A6(N),
     1 IREFO(N),A7(N),LAST
   59 FØRMAT(2H0 ,I2,2X,2(F8.4,2X),I2,2X,2(F9.4,2X),2(F7.4,2X),I2,2X,
     1 F5.1,2X,I1)
C    CHANGE ANGLES FRØM DEGREES TØ RADIANS
      A7(N) = A7(N)*.01745885
C
C    CØMPLETE DATA WHERE DITTØ HAS BEEN USED    100 TØ 199
C

      IF(N.EQ.1) GØ TØ 200

      IF(A1(N).GT.0.0.ØR.A2(N).EQ.0.0) GØ TØ 150

C    THE DIAMETER HAS BEEN DITTØED
      IA2N = IFIX(A2(N)+.5)
C    IA2N IS THE NUMBER ØF THE PLANE TØ BE CØPIED

      DØ 130 I=1,NM1

      IF(NUM(I).EQ.IA2N) GØ TØ 140

  130 CØNTINUE

      WRITE (6,135) NUM(N)
  135 FØRMAT (38H1  THE DITTØ IMPLIED FØR PLANE NUMBER ,I2,32H REFERS TØ
     1 AN UNDEFINED DIAMETER)

      GØ TØ 99990

  140 LAB(1,I) = .TRUE.
      A1(N) = A1(I)
      A2(N) = -99.

  150 IF(A5(N).GT.0.0.ØR.A6(N).EQ.0.0) GØ TØ 200

C    THE FILLET RADIUS HAS BEEN DITTØED
      IA6N = IFIX(A6(N)+.5)
C    IA6N IS THE NUMBER ØF THE PLANE TØ BE CØPIED

      DØ 180  I=1,NM1

      IF(NUM(I).EQ.IA6N) GØ TØ 190

  180 CØNTINUE

      WRITE(6,185) NUM(N)
  185 FØRMAT (38H1  THE DITTØ IMPLIED FØR PLANE NUMBER ,I2,37H REFERS TØ
     1 AN UNDEFINED FILLET RADIUS)

      GØ TØ 99990

  190 LAB(2,I) = .TRUE.
      A5(N) = A5(I)
      A6(N) = -99.

B
```


B
C CHECK FØR CIRCULAR REFERENCES AND UPDATE PØSITIØN 200 TØ 299
200 IF(IREF(N)) 99990,201,240
C NTH PLANE REFERS TØ THE LEFT-MØST PLANE (PLANE ZERØ) 201
201 PØSIT(N) = -A3(N)
LEVEL(N) = 1
IRØØT(N) = 0
IF(NM1) 300,300,205
C MØVE ALL PLANES RØØTED IN PLANE N AND RØØT THEM IN ZERØ. 205 TØ 230
205 DØ 230 I=1,NM1
IF(IRØØT(I)-NUM(N)) 230,220,230
220 LEVEL(I) = LEVEL(I)+1
IRØØT(I) = 0
PØSIT(I) = PØSIT(I)+PØSIT(N)
230 CØNTINUE
GØ TØ 300
240 IR = IREF(N)
IF(N.EQ.1) GØ TØ 260
DØ 250 I=1,NM1
IF(IR-NUM(I)) 250,275,250
250 CØNTINUE
IF(IR-NUM(N)) 260,255,260
255 WRITE(6,258) NUM(N)
258 FØRMAT(16H1 PLANE NUMBER ,I2, 27H HAS BEEN REFERED TØ ITSELF)
GØ TØ 99990
C THE NTH PLANE IS REFERRED TØ A PLANE (IR) NØT YET READ. 260 TØ 270
260 PØSIT(N) = -A3(N)
IRØØT(N) = IR
LEVEL(N) = 1
C


C
IF(N.EQ.1) GØ TØ 300
C MØVE ALL PLANES RØØTED IN N UP ØNE LEVEL AND RØØT THEM IN IR.
DØ 270 I=1,NM1
IF(IRØØT(I)-NUM(N)) 270,265,270
265 LEVEL(I) = LEVEL(I)+1
PØSIT(I) = PØSIT(I)+PØSIT(N)
IRØØT(I) = IR
270 CØNTINUE
GØ TØ 300
C THE NTH PLANE IS REFERRED TØ A PLANE ALREADY READ. 275 TØ 290
275 IF(IRØØT(I).NE.NUM(N)) GØ TØ 280
WRITE(6,278) NUM(N)
278 FØRMAT (28H1 THE REFERENCE FØR PLANE ,I2,85H CLØSES THE LØØP ØF
1A CIRCULAR REFERENCE. CHECK THIS AND ALL REFERENCES LEADING TØ IT)
GØ TØ 99990
C THE NTH PLANE IS PØSITIØNED RELATIVE TØ ITS REFERENCE. 280
280 PØSIT(N) = PØSIT(I)-A3(N)
LEVEL(N) = LEVEL(I)+1
IRØØT(N) = IRØØT(I)
DØ 290 I=1,NM1
IF(IRØØT(I)-NUM(N)) 290,285,290
C PLANES RØØTED IN N ARE MØVED WITH IT AND SHARE ITS RØØT. 285
285 PØSIT(I) = PØSIT(I)+PØSIT(N)
LEVEL(I) = LEVEL(I)+LEVEL(N)
IRØØT(I) = IRØØT(N)
290 CØNTINUE
C CHECK FØR THE LAST PLANE ØF A SHAFT. 300
300 IF(LAST)400,390,400
390 CØNTINUE
D

```
D

      WRITE(6,395)
  395 FØRMAT(45H1  THE 49TH HAS NØT BEEN SIGNALED AS THE LAST)

      GØ TØ 99990

  400 NPLANE = N+1
      NPLM1 = NPLANE - 1
      NPLP1 = NPLANE +1
C   CHECK FØR TWØ TYPES ØF INPUT ERRØR.   405  TØ  499

      DØ 499 I=1,N

      IF(PØSIT(I))450,450,430

C   ALL PØSITIØNS HAVE BEEN ASSIGNED TEMPØRARILY NEGATIVE TØ AID SØRTING

  430 WRITE(6,440) NUM(I)
  440 FØRMAT(16H1  PLANE NUMBER ,I2, 44H HAS BEEN LØCATED TØ THE LEFT ØF
     1 PLANE ZERØ.)

      GØ TØ 99990

  450 IF(IRØØT(I).EQ.0) GØ TØ 499

      WRITE (6,460) IRØØT(I)
  460 FØRMAT(14H1 PLANE NUMBER,I3,40H HAS BEEN REFERED TØ BUT NEVER DEFI
     1NED. )

      GØ TØ 99990

  499 CONTINUE

E
```

```
E

C
C   THE PLANES ARE RENUMBERED ØN THE BASIS ØF THEIR PØSITIØN  500 TØ 599
C

 5CC DØ 549 N=2,NPLANE

      L = NPLANE-N+2
C   L IS THE NEW (WØRKING) NUMBER ØF THE PLANE.
      M = 1
C   M IS THE ØLD (INPUT) NUMBER ØF THE PLANE.

      DØ 530 I=2,NPLM1

      IF(PØSIT(I)-PØSIT(M)) 525,520,530

 520 IF(A7(I).LT.A7(M)) GØ TØ 530

 525 M = I

 530 CØNTINUE

      DNØM(L) = A1(M)
      DTØL(L) = A2(M)
      RNØM(L) = A3(M)
      RTØL(L) = A4(M)
      FRNØM(L) = A5(M)
      FRTØL(L) = A6(M)
      TANG(L) = A7(M)
      A8(L) = -PØSIT(M)
      PØSIT(M) = A8(L)
C   PØSIT  IS SWITCHED PØSITIVE SØ IT WILL BE ØUT ØF THE SØRTING
      DTYP(L) = LAB(1,M)
      FRTYP(L) = LAB(2,M)
C   LAB HAS TØ BE CLEARED FØR ITS LATER USE.
      LAB(1,M) = .FALSE.
      LAB(2,M) = .FALSE.
      K = NUM(M)
C   K IS THE DESIGNER'S NUMBER FØR THE PLANE.
      II(K) = L
      JJ(L) = IREF(M)
      KK(L) = IREFO(M)

 549 CØNTINUE

F
```


F
DØ 599 N=2,NPLANE
IF(JJ(N))570,560,570
560 IREF(N) = 1
GØ TØ 575
570 M = JJ(N)
IREF(N) = II(M)
575 PØSIT(N) = AB(N)
IF(KK(N)) 580,585,590
C ANY NEGATIVE NUMBER IS INTERPRETED AS A REF TØ THE LEFTMØST PLANE.
C A ZERØ INTERP. AS NØNE-A BLANK. THIS IS SET = TØ ITSELF FØR
C DETECTIØN IN S.U.L.D. (STATEMENT 1500)
580 IREFQ(N) = 1
GØ TØ 599
585 IREFQ(N) = N
GØ TØ 599
590 M = KK(N)
IREFQ(N) = II(M)
599 CØNTINUE
PØSIT(1) = 0.0
G

G

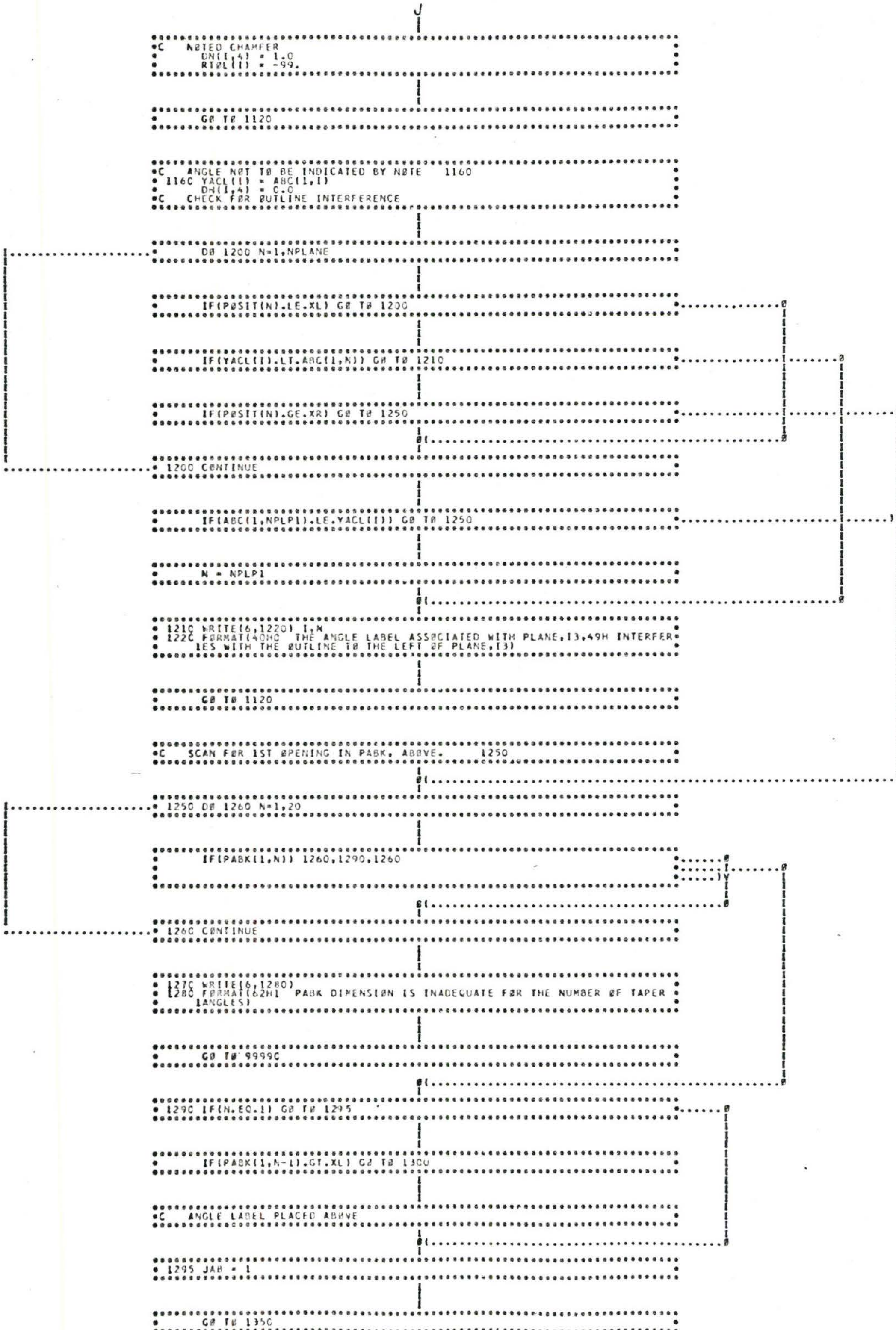
```
C
C    CHAMFER DATA CØMPLETIØN SECTIØN      600 TØ 699
C

      DØ 699 I=2,NPLANE

      IF(FRNØM(I).NE.0.0.ØR.TANG(I).EQ.0.0) GØ TØ 699

      IF(I.LT.NPLANE) GØ TØ 605

      WRITE (6,603)
  603 FØRMAT(74H1  THE RIGHTMØST PLANE CANNØT BE A CHAMFER PLANE.  CHECK
     1 THE SIGN ØF RNØM.)

      GØ TØ 99990

  605 QTAN = TAN(TANG(I))
      IT = I -IFIX(TANG(I)/ABS(TANG(I)))
C    IT  IS THE NUMBER ØF THE SHØULDER PLANE.

      IF(RNØM(I)) 630,610,630

  610 D  = DNØM(I+1) - DNØM(I)

      IF(D .GT.0.0) GØ TØ 620

      WRITE(6,615)  PØSIT(I)
  615 FØRMAT(53H1  THE REFERENCE DIAMETER ØN CHAMFER PLANE LØCATED AT,
     1  F6.2, 42H IS NØT SMALLER THAN THE ADJACENT DIAMETER)

      GØ TØ 99990

  620 PØSIT(I) = PØSIT(IT) +  .5*D/QTAN
      RTØL(I) = -99.

      GØ TØ 650

  630 DNØM(I) = DNØM(I+1)-QTAN*(PØSIT(I)-PØSIT(IT))*2.0
      DTØL(I) = -99.

  650 IF(TANG(I)) 670,699,699

C    THE SWITCH IS MADE SØ THAT D(I) WILL ALWAYS BE THE DIAMETER LEFT
C     ØF PLANE I.    670

  670 T1 = DNØM(I)
      T2 = DTØL(I)
      LDUM = DTYP(I)
      DNØM(I) = DNØM(I+1)
      DTØL(I) = DTØL(I+1)
      DTYP(I) = DTYP(I+1)
      DNØM(I+1) = T1
      DTØL(I+1) = T2
      DTYP(I+1) = LDUM

  699 CØNTINUE
```

H

```
H
C
C   FIND FILLET TANGENCY PØSITIØNS AND MAKE INPUT CHECKS 700 TØ 799
C
      PL(2) = 0.0

      DØ 799 N=2,NPLANE

      IF(FRNØM(N).NE.0.0) GØ TØ 710

C   CHAMFER PLANE
      PR(N) = PØSIT(N)
      PL(N+1) = PØSIT(N)

      GØ TØ 780

  710 IF(TANG(N).NE.0.0) GØ TØ 740

C   NØ TAPER ANGLE ASSØCIATED WITH PLANE

      IF(ABS(DNØM(N+1)-DNØM(N)) .GE. 2.*FRNØM(N)) GØ TØ 730

C   FILLET IS LESS THAN 90DEG. ARC
      P = SQRT(FRNØM(N)**2 - (FRNØM(N)-ABS((DNØM(N+1)-DNØM(N))*.5))**2)

      GØ TØ 770

C   FILLET IS 90DEG. ARC  730

  730 P = FRNØM(N)

      GØ TØ 770
```

```
C    TAPER IS ASSØC. W/ PLANE     740
C     A IS THE DIST. TØ INTERSECTIØN ØF TAPER AND HØRIZØNTAL
  740 A = ((DNØM(N+1)-DNØM(N))*.5) / TAN(TANG(N))

      IF(A.GE.0.0) GØ TØ 760

      WRITE (6,750) N
  750 FØRMAT (71H1  SIGN ØF DIAMETER DIFFERENCE AND TANG(N) ARE NØT IN A
     1GREEMENT FØR N =,I3)

      GØ TØ 99990

  760 P = A*ABS(TAN(TANG(N)*.5)) * FRNØM(N)

  770 IF(DNØM(N+1).GT.DNØM(N)) GØ TØ 775

C    TAPER AND/ØR FILLET ARE ØN RIGHT ØF PLANE
      PR(N) = PØSIT(N)
      PL(N+1) = P + PØSIT(N)

      GØ TØ 780

  775 PR(N) =PØSIT(N) - P
      PL(N+1) = PØSIT(N)

  780 IF(PR(N).GT.PL(N)) GØ TØ 799

      WRITE (6,790) N
  790 FØRMAT(43H1  INVESTIGATE FILLETS TØ THE LEFT ØF PLANE,I3,44H.  AN
     1IMPØSSIBLE GEØMETRY HAS BEEN DESCRIBED)

  799 CØNTINUE
```

```
                                   I
C
C    ANGULAR DIMENSIØN SET-UP  (ADSU)     1000 TØ 1499
C
C    INITIALIZE SHAFT CØNTØUR ARRAY ABC AND ZERØ PABK 1000 TØ 1100

        DØ 1010 L=1,2

        ABC(L,1) = 0.0

  1010 ABC(L,NPLP1) = 0.0

        DØ 1050 N=1,20

        DØ 1030 L=1,2

  1030 PABK(L,N) = 0.0

  1050 CØNTINUE

        DØ 1100 N=2,NPLANE

        ABC(1,N) = DNØM(N)*.5
        IF(TANG(N).GT.0.0) ABC(1,N) = DNØM(N+1) * .5

  1100 ABC(2,N) = ABC(1,N)

        S1A = .8/SCALE
C    S1A IS THE WIDTH ALLØTTED TØ AN ANGULAR DIMENSIØN.
        S2A = .8/SCALE
C    S2A IS THE HEIGHT ALLØTTED TØ AN ANGULAR DIMENSIØN.
        S3A = .79/SCALE
C    S3A IS PRØVIDED SØ THAT AN ANGLE WILL NØT BLØCK ITS ØWN PLANE.

        DØ 1400 I=2,NPLANE

        DN(I,4) = 0.0

        IF(TANG(I)) 1110,1120,1130

  1110 XL = PØSIT(I) - S1A
        XR = PØSIT(I)

        GØ TØ 1150

  1120 YACL(I) = 0.0

        GØ TØ 1400

  1130 XL = PØSIT(I)
        XR = PØSIT(I) + S1A
C    TEST FØR CHAMFER TØ BE INDICATED BY NØTE  1150

  1150 IF(ABS(ABS(TANG(I)) - .7854) .GT. 1E-3  .ØR.  ABS(RNØM(I)).GT..5)
      1 GØ TØ 1160

        IF(FRNØM(I).GT.0.0) GØ TØ 1160                                   )A

        IF(RNØM(I).EQ.0.0) GØ TØ 1160                                    )A

        IF(IABS(IREF(I)-I) .GT. 1) GØ TØ 1160

                                   U
```


J
C NØTED CHAMFER
DN(I,4) = 1.0
RTØL(I) = -99.
GØ TØ 1120
C ANGLE NØT TØ BE INDICATED BY NØTE 1160
1160 YACL(I) = ABC(1,I)
DH(I,4) = C.0
C CHECK FØR ØUTLINE INTERFERENCE
DØ 1200 N=1,NPLANE
IF(PØSIT(N).LE.XL) GØ TØ 1200
IF(YACL(I).LT.ABC(1,N)) GØ TØ 1210
IF(PØSIT(N).GE.XR) GØ TØ 1250
1200 CØNTINUE
IF(ABC(1,NPLP1).LE.YACL(I)) GØ TØ 1250
N = NPLP1
1210 WRITE(6,1220) I,N
1220 FØRMAT(40H0 THE ANGLE LABEL ASSØCIATED WITH PLANE,I3,49H INTERFER
1ES WITH THE ØUTLINE TØ THE LEFT ØF PLANE,I3)
GØ TØ 1120
C SCAN FØR 1ST ØPENING IN PABK, ABØVE. 1250
1250 DØ 1260 N=1,20
IF(PABK(1,N)) 1260,1290,1260
1260 CØNTINUE
1270 WRITE(6,1280)
1280 FØRMAT(62H1 PABK DIMENSIØN IS INADEQUATE FØR THE NUMBER ØF TAPER
1ANGLES)
GØ TØ 9999C
1290 IF(N.EQ.1) GØ TØ 1295
IF(PABK(1,N-1).GT.XL) GØ TØ 1300
C ANGLE LABEL PLACED ABØVE
1295 JAB = 1
GØ TØ 1350

```
C   SCAN FØR THE 1ST ØPENING IN PABK, BELØW.  1300

1300 DØ 1310 N=1,20

     IF(PABK(2,N)) 1310,1320,1310

1310 CØNTINUE

     GØ TØ 1270

1320 IF(N.EQ.1) GØ TØ 1340

     IF(PABK(2,N-1).LE.XL) GØ TØ 1340

     WRITE(6,1330) I
1330 FØRMAT(40H0  THE ANGLE LABEL ASSØCIATED WITH PLANE,I3,64H CANNØT B
    1E PLACED WITHØUT INTERFERING WITH ØNE TØ THE LEFT ØF IT)

     GØ TØ 1120

C   ANGLE LABEL PLACED BELØW.  1340

1340 JAB = 2
     YACL(I) = -YACL(I)
1350 PABK(JAB,N) = XR
     Y = ABS(YACL(I)) + S2A
C   RECØRD NEW LEVELS ØF INNER PRØFILE TEMPØRARILY IN DL.
C      (SØ THEY WØN'T MESS UP THE ØUTLINE INTERFERENCE CHECK)

     DØ 1390 J=1,NPLANE

     IF(PØSIT(J).LE.XL) GØ TØ 1390

     DL(J,JAB) = Y

     IF(PØSIT(J).GT.XR) GØ TØ 1400

1390 CØNTINUE

     DL(NPLP1,JAB) = Y

1400 CØNTINUE
C   UPDATE ALL ABC VALUES     1400 TØ 1499

     DØ 1499 J=1,NPLP1

     DØ 1450 L=1,2

1450 ABC(L,J) = AMAX1(ABC(L,J),DL(J,L))

1499 CØNTINUE

                              K
```

K

```
C
C    SET UP LENGTH DIMENSIØNS (S.U.L.D.)        1500  TØ  2499
C
C    SØRT BY NUMBER ØF PLANES DISTANT     1500 TØ 1595
C    II(1) IS THE PLANE WITH THE LARGEST REFERENCE
C    QREFS ARE SHØWN BY NEGATIVE PLANE NUMBERS.
C    JJ(N) HØLDS THE LENGTH ØF THE REF ASSØC. WITH PLANE N.
C    KK(N) HØLDS THE LENGTH ØF THE QREF ASSØCIATED WITH PLANE N
       JJ(2) = IABS(2-IREF(2))
       KK(2) = IABS(2-IREFQ(2))

       IF(KK(2)) 1505,1505,1510

 1505 NQ = 0

 1507 II(1) = 2

       GØ TØ 1520

 1510 NQ = 1

       IF(KK(2).GT.JJ(2)) GØ TØ 1515

       II(2) = -2

       GØ TØ 1507

 1515 II(1) = -2
       II(2) = 2

 1520 DØ 1595 N=3,NPLANE

       JJ(N) = IABS(N-IREF(N))
       KK(N) = IABS(N-IREFQ(N))

       IF(KK(N)-1) 1560,1550,1525

 1525 M = N+NQ-2

       DØ 1535 I=1,M

       IR = II(I)

       IF(IR.GT.0) GØ TØ 1530

       MIR = -IR

       IF(KK(N) - KK(MIR)) 1535,1535,1540

 1530 IF(KK(N).GT.JJ(IR)) GØ TØ 1540

 1535 CØNTINUE

       II(M+1) = -N

       GØ TØ 1555
```

```
C     THE CURRENT CREF GØES IN AT I.   ALL PREVIØUS ARE MØVED BACK.   1540
 1540 DØ 1545 K=I,M
       L = M+I-K
 1545 II(L+1) = II(L)
       II(I) = -N
       GØ TØ 1555
C     CREF ØF LENGTH 1 IS PUT IN LAST PLACE.      1550
 1550 M = N+NQ-1
       II(N) = -N
C     CREF CØUNTER IS INCREASED BY ØNE.      1555
 1555 NQ = NQ + 1
C     SHØRTCUT TEST. IF JJ(N) = 1 IT IS PLACED AT THE BACK END.    1560
 1560 IF(JJ(N)-1) 1565,1590,1565
 1565 M = N+NQ-2
       DØ 1575 I=1,M
       IR = II(I)
       IF(IR.GT.0) GØ TØ 1570
       MIR = -IR
       IF(JJ(N) - KK(MIR)) 1575,1575,1580
 1570 IF(JJ(N).GT.JJ(IR)) GØ TØ 1580
 1575 CØNTINUE
       GØ TØ 1590
 1580 DØ 1585 K=I,M
       L = M+I-K
 1585 II(L+1) = II(L)
       II(I) = N
       GØ TØ 1595
 1590 M = N+NQ-1
       II(M) = N
 1595 CØNTINUE
```

```
L

C     CEAL WITH DIMENSIØNS ØNE AT A TIME, LØNGEST FIRST.    1600  TØ 2500
         S1L = .20/SCALE
C     S1L IS THE LENGTH ØF LEADER LINE FØR TYPE 3 DIMENSIØN FØRMAT.
         S2L = .30/SCALE
C     S2L IS THE LENGTH ØF AN EXTERNAL ARRØW WITH EXTERNAL WRITING.
         S3L = .45/SCALE
C     S3L IS THE LENGTH ØF AN EXTERNAL ARRØW WITH INTERNAL WRITING.
         S4L = .50/SCALE
C     S4L,THE MINIMUM LENGTH ØF THE INTERNAL ARRØW WITH INTERNAL WRITING.
         S5L = .50/SCALE
C     S5L IS THE LENGTH ØF THE INTERNAL ARRØW WITH EXTERNAL WRITING.
         S6L = .20/SCALE
C     S6L IS THE LENGTH ALLØTTED AN ARRØWHEAD IF PRØVIDING A W. L. BREAK.
         M = NPLM1 + NQ

         DØ 2450 N=1,M

         IF(II(N).LT.0) GØ TØ 1610

         NP = II(N)
         IR = IREF(NP)

         IF(RTØL(NP).LT.(-10.)) GØ TØ 2450

         A = ABS(RNØM(NP))
         ATØL = RTØL(NP)
         JREF = 1

         GØ TØ 1620

C     THE PRESENT LENGTH DIMENSIØN IS A REFERENCE DIMENSIØN.  1610

 1610 NP = -II(N)
         IR = IREFC(NP)
         A = ABS(PØSIT(NP) - PØSIT(IR))
         ATØL = -4.
         JREF = 2
C     TEST FØR PLANES BLØCKED BY ANGLES   1620 TØ 1660

 1620 N1 = MINO(NP,IR)
         N2 = MAXO(NP,IR)
         MA = 0

         DØ 1630 I=1,20

         IF(PABK(1,I).LE.PØSIT(N1)) GØ TØ 1630

         IF(PABK(1,I)-S3A.LT.PØSIT(N1))MA = 1

         DØ 1625 L=1,20

         IF(PABK(1,L).LE.PØSIT(N2)) GØ TØ 1625

         IF(PABK(1,L)-S3A.LT.PØSIT(N2))MA = 1

         GØ TØ 1635

 1625 CØNTINUE

         GØ TØ 1635

 1630 CØNTINUE

 1635 MB = 0

M
```

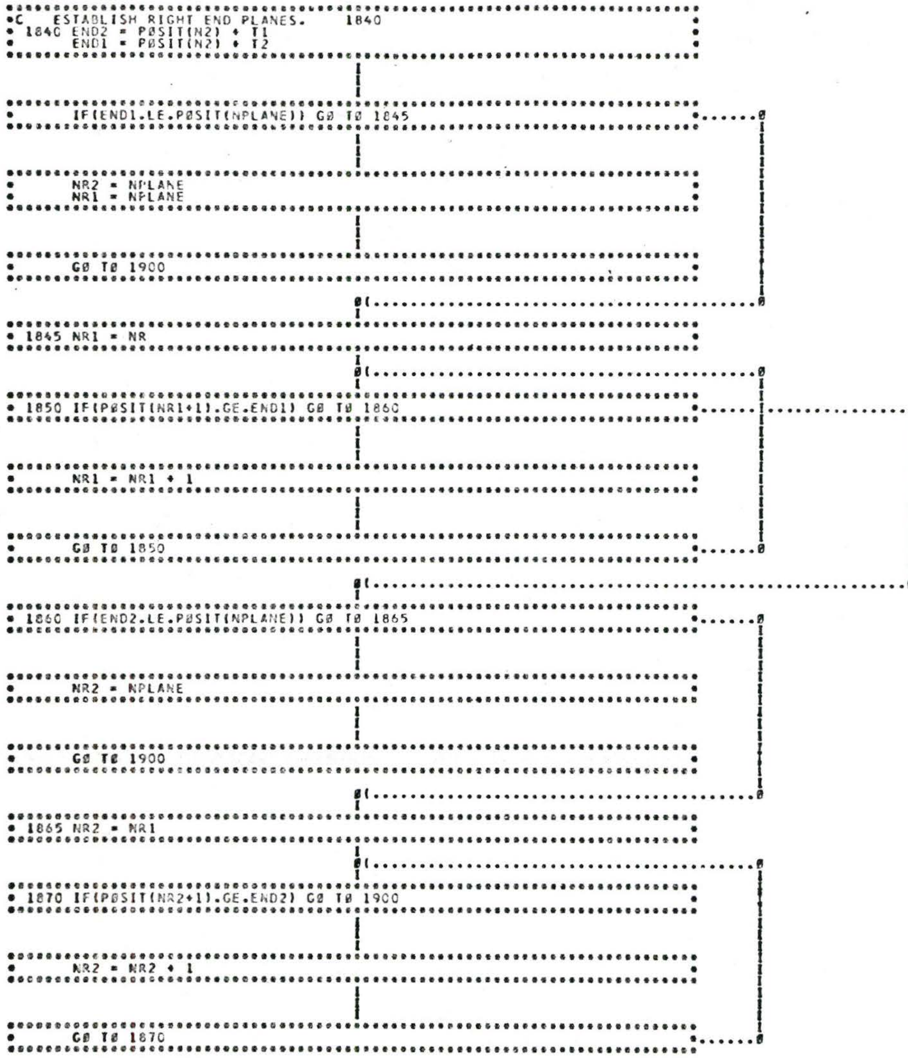
M

```
      DØ 1645 I=1,20

      IF(PABK(2,I).LE.PØSIT(N1)) GØ TØ 1645

      IF(PABK(2,I)-S1A.LT.PØSIT(N1))MB = 1

      DØ 1640 L=1,20

      IF(PABK(2,L).LE.PØSIT(N2)) GØ TØ 1640

      IF(PABK(2,L)-S1A.LT.PØSIT(N2))MB = 1

      GØ TØ 1650

 1640 CØNTINUE

      GØ TØ 1650

 1645 CØNTINUE

 1650 IF(MA+MB.LT.2) GØ TØ 1660

      WRITE(6,1655) NP,IR
 1655 FØRMAT(39HO  UNABLE TØ PLACE DIMENSIØN FRØM PLANE,I3, 9H TØ PLANE,
     1I3,37H BECAUSE ØF INTERFERENCE WITH ANGLES.)
      RTØL(NP) = -50.

      GØ TØ 2450

 1660 JAB = MB
      IF(MA.EQ.1) JAB = 2
C     WC = THE LENGTH ØF THE WRITING (IN DATA UNITS).
      WC = CLEN(A,ATØL,0,JREF-1,.FALSE.)/SCALE
      NL = N1 + 1
      NR = N2 - 1
C     CØUNT THE NUMBER ØF APPLICABLE WITNESS LINES DRAWN ALREADY.
      LA = 0
      LB = 0
      IF(LAB(1,NP)) LA = LA + 1
      IF(LAB(1,IR)) LA = LA + 1
      IF(LAB(2,IR)) LB = LB + 1
      IF(LAB(2,NP)) LB = LB + 1
C     TEST TØ SEE WHICH ARRØW FØRMAT WILL BE USED.
C     T1 AND T2 ARE THE LØNGER AND SHØRTER ØVERHANG DISTANCES ØF THE
C      DIMENSIØNING BEYØND THE PLANES CØNCERNED.

      IF(A.LT.WC+S4L) GØ TØ 1670

C     INTERNAL ARRØWS AND LETTERING.
      LCØDE(NP,JREF) = 1

      IF(NL.GT.NR) GØ TØ 1700

      T1 = 0.0
      T2 = 0.0

      GØ TØ 1800
```

```
 1670 IF(A.LT.WC) GØ TØ 1680

C     INTERNAL WRITING AND EXTERNAL ARRØWS.
         LCØDE(NP,JREF) = 2
         T1 = S3L
         T2 = S3L

         GØ TØ 1800

 1680 IF(A.LT.S5L) GØ TØ 1690

C     INTERNAL ARRØWS AND EXTERNAL WRITING
         LCØDE(NP,JREF) = 3
         T1 = S1L + WC
         T2 = 0.0

         GØ TØ 1800

C     EXTERNAL ARRØWS AND WRITING.   1690

 1690 T1 = S2L + WC
         T2 = S2L
         LCØDE(NP,JREF) = 4

         GØ TØ 1800

C     THE SIMPLE CASE ØF NEIGHBØRING PLANES WITH NØ ØVERHANG. 1700 TØ 1800

 1700 IF(JAB.GT.0) GØ TØ 1740

         IF(LA-LB) 1710,1720,1730

 1710 JAB = 2

         GØ TØ 1740

 1720 IF(KCF(1,NL).GT.KCF(2,NL)) GØ TØ 1710

 1730 JAB = 1

 1740 K = KCF(JAB,NL) + 1
         KCF(JAB,NL) = K
         ASSIGN 1780 TØ M

                                N
```

```
N
C    PROVIDE FØR STARTING NEW WITNESS LINES(WL) AND SIGNALLING
C      THEIR PRESENCE(LAB). 1750

 1750 IF(LAB(JAB,NP)) GØ TØ 1755

        WL(JAB,1,NP) =(.375*FLØAT(K) - .12)/SCALE
        LAB(JAB,NP) = .TRUE.

        GØ TØ 1760

 1755 WL(JAB,1,NP) = AMIN1(WL(JAB,1,NP),(.375*FLØAT(K)-.12)/SCALE)

 1760 IF(LAB(JAB,IR)) GØ TØ 1765

        WL(JAB,1,IR) =(.375*FLØAT(K) - .12)/SCALE
        LAB(JAB,IR) = .TRUE.

        GØ TØ 1770

 1765 WL(JAB,1,IR) = AMIN1(WL(JAB,1,IR),(.375*FLØAT(K)-.12)/SCALE)

 1770 GØ TØ M, (1780,2100)
```

```
 1780 IH(NP,JREF) = KCF(JAB,NL) * (1-2*(JAB-1))

      GØ TØ 2450

C     THE NØN-SIMPLE CASE WITH PØSSIBLE INTERFERENCE.     1800 TØ 2450
C     ESTABLISHE END PLANES FØR PØSSIBLE LEFT AND RIGHT LØCATIØNS,
C       LEFT END FIRST.    1800 TØ 1900
 1800 END1 = PØSIT(N1) - T1
      END2 = PØSIT(N1) - T2

      IF(END2.GE.0.0) GØ TØ 1805

      NL2 = 1
      NL1 = 1

      GØ TØ 1840

 1805 NL2 = NL

 1810 IF(PØSIT(NL2-1).LE.END2) GØ TØ 1820

      NL2 = NL2 - 1

      GØ TØ 1810

 1820 IF(END1.GE.0.0) GØ TØ 1825

      NL1 = 1

      GØ TØ 1840

 1825 NL1 = NL2

 1830 IF(PØSIT(NL1-1).LE.END1) GØ TØ 1840

      NL1 = NL1 - 1

      GØ TØ 1830
```


C ESTABLISH RIGHT END PLANES. 1840
1840 END2 = PØSIT(N2) + T1
END1 = PØSIT(N2) + T2
IF(END1.LE.PØSIT(NPLANE)) GØ TØ 1845
NR2 = NPLANE
NR1 = NPLANE
GØ TØ 1900
1845 NR1 = NR
1850 IF(PØSIT(NR1+1).GE.END1) GØ TØ 1860
NR1 = NR1 + 1
GØ TØ 1850
1860 IF(END2.LE.PØSIT(NPLANE)) GØ TØ 1865
NR2 = NPLANE
GØ TØ 1900
1865 NR2 = NR1
1870 IF(PØSIT(NR2+1).GE.END2) GØ TØ 1900
NR2 = NR2 + 1
GØ TØ 1870

```
C     SURVEY NUMBER ØF INTERFERENCES(M1 AND M2) AND SPACE(KAB1 AND KAB2)
C      1900  TØ 2000

 1900 DØ 2000 L = 1,2

      M1(L) = 0
      M2(L) = 0
      KAB1(L) = KCF(L,NR1+1)
      KAB2(L) = KCF(L,NR2+1)

      DØ 1940 J=NL1,NR1

      KAB1(L) = MAX0(KAB1(L),KCF(L,J))

      IF(J.EQ.NP.ØR.J.EQ.IR) GØ TØ 1940

      IF(LAB(L,J))  M1(L) = M1(L) + 1

 1940 CØNTINUE

      DØ 1980 J=NL2,NR2

      KAB2(L) = MAX0(KAB2(L),KCF(L,J))

      IF(J.EQ.NP.ØR.J.EQ.IR) GØ TØ 1980

      IF(LAB(L,J))  M2(L) = M2(L) + 1

 1980 CØNTINUE

      M3(L) = MIN0(M1(L),M2(L))

 2000 CØNTINUE

C     DETERMINE WHETHER ABØVE(JAB=1) ØR BELØW(JAB=2) IS PREFERABLE.

      IF(JAB.GT.0) GØ TØ 2050

      IF(M3(2)-M3(1)) 2010,2020,2030

 2010 JAB = 2

      GØ TØ 2050

 2020 IF(LA-LB) 2010,2040,2030

 2030 JAB = 1

      GØ TØ 2050

 2040 IF(MIN0(KAB1(1),KAB2(1)) - MIN0(KAB1(2),KAB2(2))) 2030,2030,2010
```

```
C     CHØØSE THE RIGHT ØR LEFT ØPTIØN      2050  TØ  2090

  2050 IF(M1(JAB)-M2(JAB)) 2060,2070,2080

C     LEFT ØPTIØN.     2060

  2060 LØ = -1
       K = KAB1(JAB) + 1
       N1 = NL1
       N2 = NR1 + 1

       GØ TØ 2090

  2070 IF(KAB1(JAB).LE.KAB2(JAB)) GØ TØ 2060

C     RIGHT ØPTIØN.   2080

  2080 LØ = 1
       K = KAB2(JAB) + 1
       N1 = NL2
       N2 = NR2 + 1

  2090 ASSIGN 2100 TØ M

       GØ TØ 1750

C     RETURN FRØM UPDATE LØGICAL ARRAY, LAB.        2100
  2100 IH(NP,JREF) = K*(1-2*(JAB-1))
       LØH(NP,JREF) = LØ
C     UPDATE KCF.

       DØ 2130  I=N1,N2

  2130 KCF(JAB,I) = K
```

```
C   TEST FØR INTERFERENCE FØR PRØVIDING WITNESS LINE BREAKS.

      IF(M3(JAB).EQ.0) GØ TØ 2450

      NM1 = N2 - 1

      DØ 2300 I=N1,NM1

      IF(.NØT.LAB(JAB,I)) GØ TØ 2300

      IF(I-NL+1) 2150,2300,2140

 2140 IF(I-NR-1) 2200,2300,2160

C   PLANE I IS LEFT ØF THE LEFT WITNESS LINE.  2150

 2150 B = PØSIT(NL-1) - PØSIT(I)

      GØ TØ 2170

C   PLANE I IS RIGHT ØF THE RIGHT WITNESS LINE.  2160

 2160 B = PØSIT(I) - PØSIT(NR+1)

 2170 IF(B.GT.T2)  CALL SKIP(JAB,I,K,.10)
      IF(B.LE.S6L) CALL SKIP(JAB,I,K,.05)

      GØ TØ 2300

C   PLANE I IS BETWEEN THE WITNESS LINES.    2200

 2200 IF(T1.NE.T2) GØ TØ 2220

C   T1 = T2 MEANS THE WRITING IS BETWEEN THE WITNESS LINES
C   B IS THE MIDPØINT. (RECALL THAT A IS THE DISTANCE BETWEEN W.L.).
      B = PØSIT(NL-1) + A*.5

      IF(ABS(B-PØSIT(I)).GE.WC*.5) GØ TØ 2220

      CALL SKIP(JAB,I,K,.10)

      GØ TØ 2300

C   TEST FØF INTERNAL ARRØWS.    2220

 2220 IF(A.LT.S5L) GØ TØ 2300

      B = AMIN1(PØSIT(I)-PØSIT(NL-1),PØSIT(NR+1)-PØSIT(I))
      IF(B.LE.S6L) CALL SKIP(JAB,I,K,.05)

 2300 CØNTINUE

 2450 CØNTINUE
```

```
C     LØCATE DIAMETRAL DIMENSIØN LINES     2500  TØ  3499
C     ZERØ ARRAY WLH

 2500 DØ 2550 J=1,30

 2550 WLH(J,1) = 0.0

C     LØCATE DIAMETERS, CENTRAL ØNES FIRST  2600  TØ  3499
         S2D = .30/SCALE
C     S2D IS THE LENGTH ØF HØRIZØNTAL WITNESS LINES

      DØ 3499 I=1,NPLM1

      N = (NPLANE+2)/2 + (1-(MØD(I,2)*2))*I/2

      IF(DTØL(N).GT.(-10.)) GØ TØ 2610

C     TØLERANCE = -99.,INDICATING A TYPICAL DIMENSIØN NØT TØ APPEAR

      GØ TØ 3490

 2610 IF(FRNØM(N).NE.0.0  .ØR.  TANG(N).LE. 0.0) GØ TØ 2620

C     A LEFT CHAMFER

      IF(RNØM(N).EQ.0.0) GØ TØ 2680

C     THE CHAMFER HAS NØ SHØULDER DIAMETER.

      GØ TØ 3490

 2620 IF(FRNØM(N-1).NE.0.0 .ØR. TANG(N-1).GE.0.0) GØ TØ 2630

C     A RIGHT CHAMFER

      IF(RNØM(N-1).EQ.0.0) GØ TØ 2680

C     THE CHAMFER HAS NØ SHØULDER DIAMETER.

      GØ TØ 3490

 2630 A = PØSIT(N)-PØSIT(N-1)

      IF(A.GT..20/SCALE) GØ TØ 2640

C     A SECTIØN NARRØW ENØUGH THAT WITNESS LINES MUST BE USED

      GØ TØ 2690
```

```
 2640 B = PR(N) - PL(N)
C     CØMPLICATED METHØD ØF DEDUCING TWØ PØSSIBLE X LØCATIØNS FØR THE
C      DIAMETER INDICATING ARRØW. CØMPLICATIØN ARISES FRØM EFFØRT TØ BE
C      CØNSISTENT WITH LØWER LIMIT ØN A ØF .20/SCALE.
       XL2 = AMAX1(PL(N)+.1*B,PØSIT(N-1)+.10/SCALE)
       XR1 = AMIN1(PR(N)-.1*B,PØSIT(N)-.10/SCALE)
```

```
       IF(XL2.LT.XR1) GØ TØ 2950
```

```
       XL2 = PL(N) +.5*B
       XR1 = XL2
       S10 = 0.0
```

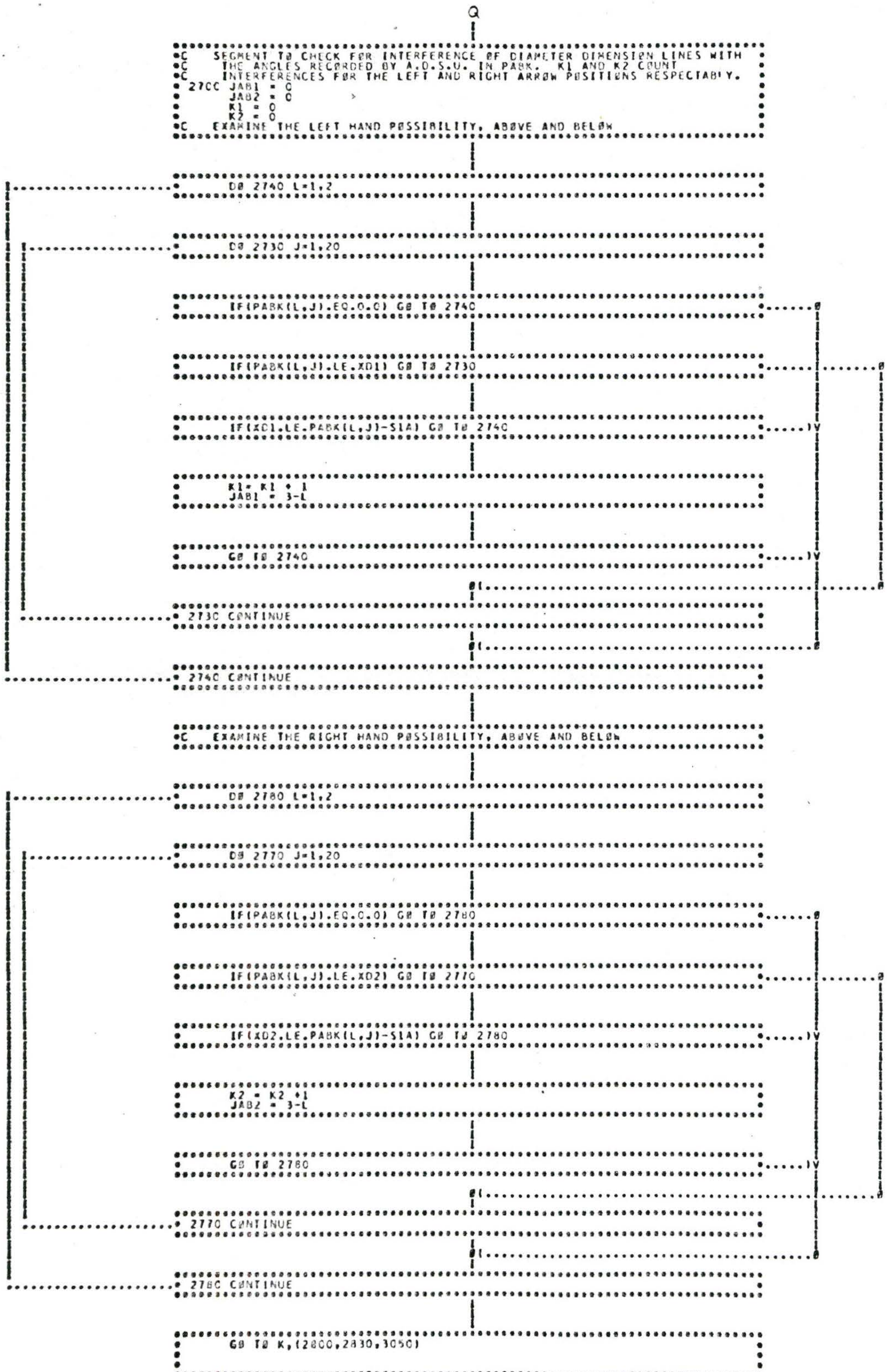
```
       GØ TØ 2950
```

```
C     CHAMFER WITH SHØULDER DIAMETER TØ BE INDICATED.   2680
 2680 XD1 = PØSIT(N) - S2D
       XD2 = PØSIT(N) + S2D
       ASSIGN 2800 TØ K
```

```
       GØ TØ 2700
```

```
C     GRØØVE ØR RIDGE REQUIRING WITNESS LINES. 2690
 2690 XD1 = PØSIT(N-1) - S2D
       XD2 = PØSIT(N) +S2D
       ASSIGN 2830 TØ K
```

```
                              Q
```

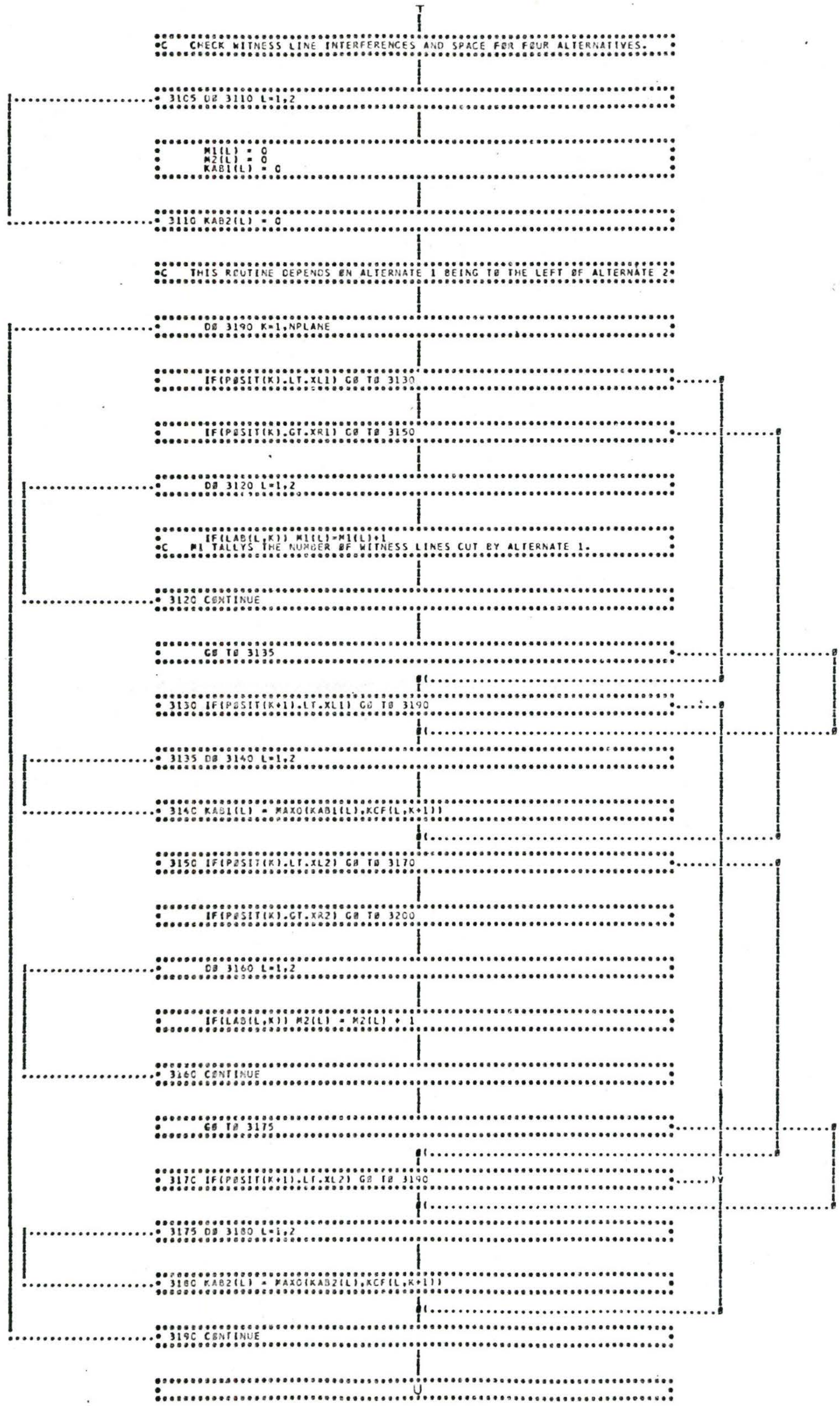

Q
C SEGMENT TØ CHECK FØR INTERFERENCE ØF DIAMETER DIMENSIØN LINES WITH
C THE ANGLES RECØRDED BY A.D.S.U. IN PABK. K1 AND K2 COUNT
C INTERFERENCES FØR THE LEFT AND RIGHT ARRØW PØSITIØNS RESPECTABLY.
2700 JAB1 = 0
JAB2 = 0
K1 = 0
K2 = 0
C EXAMINE THE LEFT HAND PØSSIBILITY, ABØVE AND BELØW
DØ 2740 L=1,2
DØ 2730 J=1,20
IF(PABK(L,J).EQ.0.0) GØ TØ 2740
IF(PABK(L,J).LE.XD1) GØ TØ 2730
IF(XD1.LE.PABK(L,J)-S1A) GØ TØ 2740
K1= K1 + 1
JAB1 = 3-L
GØ TØ 2740
2730 CØNTINUE
2740 CØNTINUE
C EXAMINE THE RIGHT HAND PØSSIBILITY, ABØVE AND BELØW
DØ 2780 L=1,2
DØ 2770 J=1,20
IF(PABK(L,J).EQ.0.0) GØ TØ 2780
IF(PABK(L,J).LE.XD2) GØ TØ 2770
IF(XD2.LE.PABK(L,J)-S1A) GØ TØ 2780
K2 = K2 +1
JAB2 = 3-L
GØ TØ 2780
2770 CØNTINUE
2780 CØNTINUE
GØ TØ K,(2800,2830,3050)

```
C   CHØØSE THE SIDE FØR WITNESS LINE PLACEMENT.   2800  TØ  2899
C   SHØULDER DIAMETER, ØNE SIDE ELIMINATED ØNLY IF INTERFERENCE ABØVE
C     AND BELØW.      2800

 2800 IF(K1-2) 2820,2810,2820

 2810 IF(K2.LT.2) GØ TØ 2890

      WRITE(6,2815) N
 2815 FØRMAT(59H0  CANNØT DIMENSIØN SHØULDER DIAMETER ASSØCIATED WITH PL
     1ANE,I3,39H BECAUSE ØF INTERFERENCE WITH AN ANGLE.)

      GØ TØ 3490

 2820 IF(K2.EQ.2) GØ TØ 2880

      IF(TANG(N)) 2890,2890,2880

C   GRØØVE ØR RIDGE, EITHER SIDE ELIMINATED BECAUSE ØF MØRE
C     INTERFERENCES.  2830

 2830 IF(K1-K2) 2880,2840,2890

 2840 IF(K1.LT.2) GØ TØ 2850

      WRITE(6,2845)N
 2845 FØRMAT(49H0  CANNØT DIMENSIØN DIAMETER TØ THE LEFT ØF PLANE,I3,
     1 39H BECAUSE ØF INTERFERENCE WITH AN ANGLE.)

      GØ TØ 3490

 2850 IF(PØSIT(N+1)-PØSIT(N) .GT. PØSIT(N-1)-PØSIT(N-2)) GØ TØ 2890

C   CHØØSE LEFT HAND WITNESS LINES, FAVØRING WRITING ABØVE.   2880

 2880 SID = -1.0
      IS = N-1
      JAB = MAX0(JAB1,1)

      GØ TØ 2900

C   CHØØSE RIGHT HAND WITNESS LINES, FAVØRING WRITING ABØVE.   2890

 2890 SID = 1.0
      IS = N
      JAB = MAX0(JAB2,1)

                                R
```

```
R

C    SCAN FØR 1ST ØPEN SPØT IN WLH  2900

2900 DØ 2910 J=1,30

     IF(WLH(J,1).EQ.0.0 ) GØ TØ 2920

2910 CØNTINUE

     WRITE(6,2915)
2915 FØRMAT(25H1  WLH DIMENSIØN EXCEEDED)

     GØ TØ 99990

2920 X1 = PØSIT(IS) + S1D*S2D
     WLH(J,1) = DNØM(N)*.5
     WLH(J,2) = X1 + S1D*S2D*.2
     WLH(J,3) = X1 - S1D*S2D*.65
     XL2 = X1
     XR1 = X1
     JSTØRE = J
C    J IS STØRED FØR UPDATING ABC (3330)
2950 WC = CLEN(DNØM(N),DTØL(N),2,0,DTYP(N))/SCALE
     XR2 = XL2+WC+S2D
     XL1 = XR1-WC-S2D
C    LEFT HAND ALTERNATE (XR1,XL1 ARE ITS ENDPØINTS) CØRRESPØNDS TØ
C     RIGHT HAND ARRØW LØCATIØN.
     ID = 1+(N-2)/(NPLANE/2)
C    ID SHØWS LEFT(ID=1) ØR RIGHT(ID=2) CHØICE
     IF(I.EQ.1) ID = 0
     S3D = 1.0

     IF(XL2.GE.XL1) GØ TØ 2960

     S3D = -1.0
C    S3D RECØRDS THE SWITCH MADE TØ ASSURE US THAT 1 IS THE LEFT HAND SET
C     ØF END PØINTS
C    AFTER SWITCHING, LEFT HAND PLACEMENT (ALTERNATE 1) CØRRESPØNDS TØ
C     LEFT HAND ARRØW LØCATIØN.
     XL = XL1
     XR = XR1
     XL1 = XL2
     XR1 = XR2
     XR2 = XR
     XL2 = XL
C    IF A SWITCH IS MADE, THIS MEANS SECTIØN IS LØNG ENØUGH THAT
C     DIAMETER DIMENSIØN INTERFERENCE CANNØT BE A PRØBLEM.
     ID = 0
C    TEST FØR INTERFERENCE WITH ANGLES FØR THØSE NØT ALREADY TESTED  2960

2960 IF(S1D.EQ.0.0 ) GØ TØ 2970

C    DIAMETER WITH WITNESS LINES, ALREADY TESTED

     GØ TØ 3105

2970 IF(XR1.NE.XL2 .ØR. S3D.EQ.(-1.0))  GØ TØ 3000

S
```

```
S
C     THERE IS ØNLY ØNE PØSSIBLE X PØSITIØN FØR THE DIAMETER ARRØW.
C     EXAMINE ABØVE AND BELØW
      K = 0
      JAB = 0

      DØ 2990 L=1,2

      DØ 2980 J=1,20

      IF(PABK(L,J).EQ.0.0) GØ TØ 2990

      IF(PABK(L,J).LE.XR1) GØ TØ 2980

      IF(XR1.LE.PABK(L,J)-S1A) GØ TØ 2990

      K = K+1
      JAB = 3-L

      GØ TØ 2990

 2980 CØNTINUE

 2990 CØNTINUE

      IF(K.LT.2) GØ TØ 3105

      WRITE(6,2995) N
 2995 FØRMAT(32H0  DIAMETER TØ THE LEFT ØF PLANE,I3,73H IS NØT BEING LAB
     1ELED BECAUSE INTERFERENCE WITH AN ANGLE WAS UNAVØIDABLE.)

      GØ TØ 3490

 3000 ASSIGN 3050 TØ K

      IF(S3D)3010,3010,3020

 3010 XD1 = XL1
      XD2 = XR2

      GØ TØ 2700

 3020 XD1 = XL2
      XD2 = XR1

      GØ TØ 2700
```

```
C   RETURN FRØM INTERFERENCE CØUNTING.  3050

 3C5C IF(K1.LT.2 .ØR. K2.LT.2) GØ TØ 3060

      WRITE(6,2995) N

      GØ TØ 3490

C   ØNE SIDE IS BLØCKED CØMPLETELY. THE ØTHER HAS AT LEAST ØNE ØPENING.
C       3060 AND 3C70

 3C6C IF(K2.LT.2) GØ TØ 3C70

      ID = 1
C   ID SHØWS A LEFT ØR RIGHT PREFERENCE AS JAB SHØWS ØNE ABØVE ØR BELØW

      GØ TØ 3100

 3C70 IF(K1.LT.2) GØ TØ 308C

      ID = 2

      GØ TØ 3100

 3080 IF(ID.NE.0) GØ TØ 3100

      IF(JAB1.NE.JAB2) GØ TØ 3090

C   BØTH SIDES IDENTICAL W/ AT LEAST ØNE ØPENING
      JAB = JAB1

      GØ TØ 3100

C   IDA AND IDB ARE INDICATØRS ØF (RIGHT ØR LEFT SIDE) CHØICE ABØVE AND
C     BELØW RESPECTIVELY.  THE TWØ SIDES ARE DIFFERENT.    3090

 3C9C IDA = 0
      IDB = 0
      JAB = 0
      IF(JAB1.EQ.1) IDB = 2
      IF(JAB2.EQ.1) IDB = 1
      IF(JAB1.EQ.2) IDA = 2
      IF(JAB2.EQ.2) IDA = 1

      GØ TØ 3105

 31CC IF(ID.EQ.1)  JAB = JAB1
      IF(ID.EQ.2)  JAB = JAB2
      IDA = ID
      IDB = ID

                               T
```


T
C CHECK WITNESS LINE INTERFERENCES AND SPACE FØR FØUR ALTERNATIVES.
3105 DØ 3110 L=1,2
M1(L) = 0
M2(L) = 0
KAB1(L) = 0
3110 KAB2(L) = 0
C THIS ROUTINE DEPENDS ØN ALTERNATE 1 BEING TØ THE LEFT ØF ALTERNATE 2
DØ 3190 K=1,NPLANE
IF(PØSIT(K).LT.XL1) GØ TØ 3130
IF(PØSIT(K).GT.XR1) GØ TØ 3150
DØ 3120 L=1,2
IF(LAB(L,K)) M1(L)=M1(L)+1
C M1 TALLYS THE NUMBER ØF WITNESS LINES CUT BY ALTERNATE 1.
3120 CØNTINUE
GØ TØ 3135
3130 IF(PØSIT(K+1).LT.XL1) GØ TØ 3190
3135 DØ 3140 L=1,2
3140 KAB1(L) = MAX0(KAB1(L),KCF(L,K+1))
3150 IF(PØSIT(K).LT.XL2) GØ TØ 3170
IF(PØSIT(K).GT.XR2) GØ TØ 3200
DØ 3160 L=1,2
IF(LAB(L,K)) M2(L) = M2(L) + 1
3160 CØNTINUE
GØ TØ 3175
3170 IF(PØSIT(K+1).LT.XL2) GØ TØ 3190
3175 DØ 3180 L=1,2
3180 KAB2(L) = MAX0(KAB2(L),KCF(L,K+1))
3190 CØNTINUE
U

```
C     DETERMINE WHETHER THE DIMENSIØN WILL BE PLACED ABØVE (JAB=1) ØR
C       BELØW (JAB=2).   3200

 3200 IF(JAB.GT.0) GØ TØ 3260

      IF(IDA-1) 3205,3210,3215

 3205 MA = MIN0(M1(1),M2(1))
      KA = MIN0(KAB1(1),KAB2(1))

      GØ TØ 3220

 3210 MA = M1(1)
      KA = KAB1(1)

      GØ TØ 3220

 3215 MA = M2(1)
      KA = KAB2(1)

 3220 IF(IDB-1) 3225,3230,3235

 3225 MB = MIN0(M1(2),M2(2))
      KB  = MIN0(KAB1(2),KAB2(2))

      GØ TØ 3240

 3230 MB = M1(2)
      KB = KAB1(2)

      GØ TØ 3240

 3235 MB = M2(2)
      KB = KAB2(2)
C     FIRST CRITERIØN IS MINIMIZING INTERFERENCE.   3240

 3240 IF(MA-MB) 3255,3250,3245

 3245 JAB = 2
      IF(ID.EQ.0)  ID=IDB

      GØ TØ 3260

 3250 IF(KA-KB) 3255,3255,3245

 3255 JAB = 1
      IF(ID.EQ.0)  ID=IDA
C     DETERMINE WHETHER DIMENSIØN WILL BE GIVEN LEFT ØF RIGHT PLACEMENT.

 3260 IF(ID-1) 3265,3290,3280

 3265 IF(M1(JAB)-M2(JAB)) 3290,3270,3280

 3270 IF(KAB1(JAB)-KAB2(JAB)) 3290,3290,3280
```

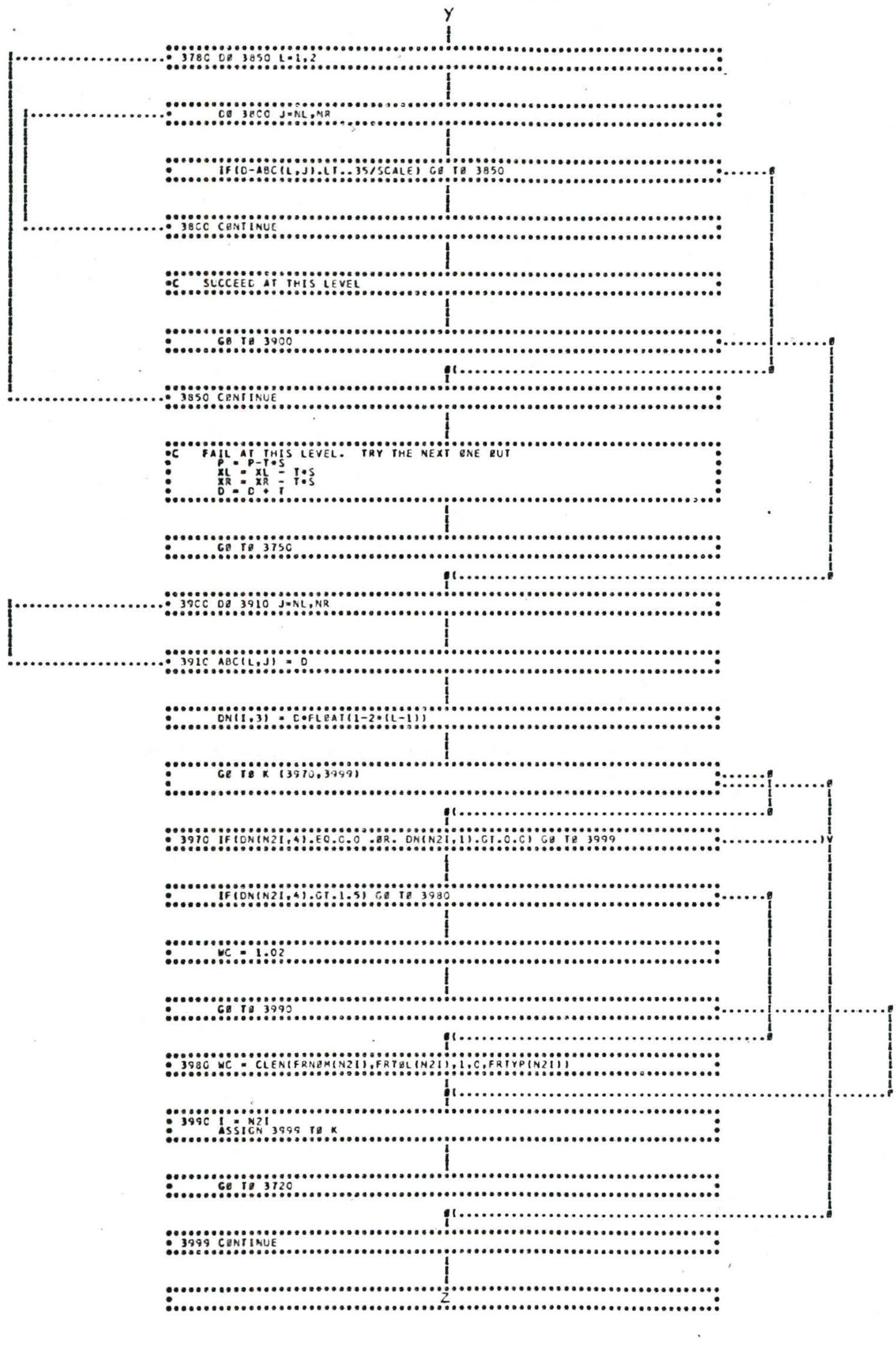
```
C    RECØRD HAND ØF TURN(AS ØPPØSED TØ HAND ØF PLACEMENT) IN DL(N,3)
C    RIGHT HAND PLACEMENT (ALTERNATE 2).    3280
 3280 DL(N,3) = S3D
      K = KAB2(JAB) + 1

      IF(S3D.GT.0.0) GØ TØ 3285

C    LEFT HAND TURN
      DL(N,1) = XR2

      GØ TØ 3300

C    RIGHT HAND TURN 3285

 3285 DL(N,1) = XL2

      GØ TØ 3300

C    LEFT HAND PLACEMENT(ALTERNATE 1).    3290
 3290 DL(N,3) = -S3D
      K = KAB1(JAB) + 1

      IF(S3D.GT.0.0) GØ TØ 3295

C    RIGHT HAND TURN
      DL(N,1) = XL1

      GØ TØ 3300

C    LEFT HAND TURN  3295

 3295 DL(N,1) = XR1

                                 V
```

```
V

C     RECØRD PLACEMENT LEVEL ØF WRITING IN DL(N,2).   3300
 33CC DL(N,2) = FLØAT(K*(1-2*(JAB-1)))
C     X1 AND X2 ARE THE TWØ ENDS ØF THE WRITING.
         X1 = DL(N,1) + DL(N,3)*S2D
         X2 = DL(N,1) + DL(N,3)*WC
C     XLØ AND XRØ ARE THE ENDS ØF WRITING PLUS LEADER.
         XLØ = AMIN1(X2,DL(N,1))
         XRØ = AMAX1(X2,DL(N,1))
C     XLI AND XRI ARE THE LEFT AND RIGHT ENDS ØF THE WRITING.
         XLI = AMIN1(X1,X2)
         XRI = AMAX1(X1,X2)
C     UPDATE LEVELS FILLED ARRAY, KCF  AND PRØVIDE FØR W.L. BREAKS

         DØ 3325 J=1,NPLANE

         IF(PØSIT(J).LE.XLØ) GØ TØ 3325

         KCF(JAB,J) = K

         IF(PØSIT(J).GE.XRØ) GØ TØ 3330

         IF(XLI.GE.PØSIT(J) .ØR. XRI.LE.PØSIT(J)) GØ TØ 3325

         IF(.NØT.LAB(JAB,J)) GØ TØ 3325

         CALL SKIP(JAB,J,K,.1C)

 3325 CØNTINUE

         KCF(JAB,NPLP1) = K

 3330 IF(S1D.EQ.0.0) GØ TØ 3499

C     FØR CASES WITH WITNESS LINES, ABC IS UPDATED
         J = JSTØRE
         XL = AMIN1(WLH(J,2),WLH(J,3))
         XR = AMAX1(WLH(J,2),WLH(J,3))

         DØ 3350 L=1,NPLANE

         IF(PØSIT(L).LE.XL) GØ TØ 3350

         ABC(1,L) = AMAX1(WLH(J,1),ABC(1,L))
         ABC(2,L) = AMAX1(WLH(J,1),ABC(2,L))

         IF(PØSIT(L).GE.XR) GØ TØ 3499

 3350 CØNTINUE

         ABC(1,NPLP1) = AMAX1(WLH(J,1),ABC(1,NPLP1))
         ABC(2,NPLP1) = AMAX1(WLH(J,1),ABC(2,NPLP1))

         GØ TØ 3499

 3490 DL(N,2) = 0.0

 3499 CØNTINUE

W
```

```
W

C
C    SET-UP NØTES WITH LEADERS        3500    TØ  3999
C
        SIN = .30/SCALE
C    SIN IS THE LENGTH ØF THE HØRIZØNTAL TAIL ØF THE LEADER ARRØWS.
C    LØCATE LEADER ARRØW PØINT        3500    TØ  3700

        DØ 3700 I=2,NPLANE

        IF(DN(I,4).NE.1.0) GØ TØ 3600

C    THIS IS A NØTED CHAMFER.
        J = IFIX(SIGN(1.,TANG(I)))
        IMJ = I-J
        DN(I,1) = PØSIT(I)-.5*(PØSIT(I)-PØSIT(IMJ))
C    DN(I,1) IS PØSITIVE FØR RIGHT PØINTING ARRØWS
        DN(I,1) = DN(I,1) * FLØAT(J)
        DN(I,2) = (DNØM(I)+DNØM(I+1)) * .25

        GØ TØ 3700

C    TEST FØR CHAMFER(NØT A NØTED ØNE).   3600

  3600 IF(FRNØM(I).EQ.0.0) GØ TØ 3690

C    TEST FØR DIMENSIØN TØ BE ØMITTED.

        IF(FRTØL(I).LT.(-10.)) GØ TØ 3690

        IF(DNØM(I)-DNØM(I+1)) 3610,3690,3620
```

```
 3610 RB = DNØM(I+1)*.5
      RS = DNØM(I)*.5
      J = 1

      GØ TØ 3630

 3620 RB = DNØM(I) * .5
      RS = DNØM(I+1) * .5
      J = -1
C   TEST FØR A TAPER.    3630

 3630 IF(TANG(I).EQ.0.0) GØ TØ 3640

      IF(ABS(TANG(I)).LE..7854) GØ TØ 3650

C   LØCATE AT 45DEG. PT. WITH TAPER ENTERING PRØBLEM.
      DN(I,1) = (PØSIT(I)-(RB-RS)/TAN(TANG(I))-FRNØM(I)*TAN(TANG(I)*.5)
     1   + FRNØM(I)*SIN(TANG(I)*.5)) * FLØAT(J)
      DN(I,2) = RS + FRNØM(I)*(1.-CØS(TANG(I)*.5))

      GØ TØ 3660

C   NØ TAPER INVØLVED.    3640

 3640 IF(RB-RS.LT..293*FRNØM(I)) GØ TØ 3650

      DN(I,1) =(PØSIT(I) - .293*FRNØM(I)*FLØAT(J)) * FLØAT(J)
      DN(I,2) = RS + .293 * FRNØM(I)

      GØ TØ 3660

C   FØR LACK ØF BETTER TØ DØ THE ARRØW IS DIRECTED TØ THE ØUTLINE.  3650

 3650 DN(I,1) = PØSIT(I) * FLØAT(J)
      DN(I,2) = RB

 3660 DN(I,4) = 2.0

      GØ TØ 3700

C   SET DN4 = 0 FØR NØ NØTE.    3690
 3690 DN(I,4) = 0.0

 3700 CØNTINUE

X
```

```
X

C     LØCATE THE WRITING LEVEL FØR NØTES.  3700 TØ 3999

      DØ 3999 N=2,NPLANE

      N2I = NPLANE + 2 - N

      IF(DN(N,4).EQ.0.0 .ØR.  DN(N,1).LT.0.0) GØ TØ 3970

C     A RIGHT PØINTING NØTE TØ BE LØCATED.
      ASSIGN 3970 TØ K
      I = N

      IF(DN(N,4).GT.1.5) GØ TØ 3710

      WC = 1.02

      GØ TØ 3720

 3710 WC = CLEN(FRNØM(N),FRTØL(N),1,0,FRTYP(N))
C     LØCATING RØUTINE.    3720 TØ 3950

 3720 S = SIGN(1.,DN(I,1))
C     S IS PØSITIVE FØR A RIGHT-PØINTING NØTE.
      IS = IFIX(S+.5)
      T = .375/SCALE
      D = FLØAT(IFIX((DN(I,2) + .40/SCALE)/T)+1) * T
C     D IS THE 1ST PØSSIBLE LEVEL
      P = ABS(DN(I,1)) - S*(D-DN(I,2))
C     P IS THE PRØPØSED CRØØK PØINT X LØCATIØN.
      Q = P - S*(WC/SCALE + SIN)
C     Q IS THE CØRRESPØNDING X LØC ØF THE END ØF THE WRITING.
      XL = AMIN1(P,Q)
      XR = AMAX1(P,Q)
 3750 NL = 1

      DØ 3770 J=1,NPLANE

      IF(PØSIT(J).GT.XL) GØ TØ 3760

      NL = NL + 1

      GØ TØ 3770

 3760 NR = J

      IF(PØSIT(J).GT.XR) GØ TØ 3780

 3770 CØNTINUE

      NR = NPLP1

Y
```


Y
378C DØ 3850 L=1,2
DØ 38CO J=NL,NR
IF(D-ABC(L,J).LT..35/SCALE) GØ TØ 3850
38CC CØNTINUE
C SUCCEED AT THIS LEVEL
GØ TØ 3900
3850 CØNTINUE
C FAIL AT THIS LEVEL. TRY THE NEXT ØNE BUT
P = P-T*S
XL = XL - T*S
XR = XR - T*S
D = D + T
GØ TØ 3750
39CC DØ 3910 J=NL,NR
391C ABC(L,J) = D
DN(I,3) = D*FLØAT(1-2*(L-1))
GØ TØ K (3970,3999)
3970 IF(DN(N2I,4).EQ.0.0 .ØR. DN(N2I,1).GT.0.0) GØ TØ 3999
IF(DN(N2I,4).GT.1.5) GØ TØ 3980
WC = 1.02
GØ TØ 3990
3980 WC = CLEN(FRNØM(N2I),FRTØL(N2I),1,C,FRTYP(N2I))
399C I = N2I
ASSIGN 3999 TØ K
GØ TØ 3720
3999 CØNTINUE
Z

Z

```
C
C    MERGE THE INNER AND ØUTER LEVELS.    4CCO TØ 4CS9
C
        S1M = .375
C    S1M IS THE DISTANCE BETWEEN LEVELS ØF DIMENSIØNING (IN INCHES).
        S2M = .40/SCALE
C    S2M IS THE MIN. ALLØWABLE APPRØACH ØF DIMENSIØN LINES TØ
C    PART ØUTLINE IN DATA UNITS.
C  CØUNT THE NUMBER ØF LEVELS ØF DIMENSIØNING REQUIRED.
        NLEV(1) = C
        NLEV(2) = C
```

```
        DØ 4020 I = 1,NPLP1
```

```
        DØ 4010 L=1,2
```

```
  401C NLEV(L) = MAXO(NLEV(L),KCF(L,I)+IFIX((ABC(L,I)+S2M)*SCALE/S1M))
```

```
  402C CØNTINUE
```

```
        NPW = IFIX(PWIDTH/S1M)
```

```
        IF(NLEV(1) + NLEV(2) + 2 .LE. NPW) GØ TØ 4050
```

```
        P = S1M * FLØAT(NLEV(1)+NLEV(2)+2-NPW)
        WRITE(6,403C) P
  403C FØRMAT(3OHO PAPER WIDTH IS TØØ NARRØW BY,F6.1,8H INCHES.)
```

```
        GØ TØ 9000
```

```
  405C IF(NLEV(1).LT.(NPW-1)/2) GØ TØ 4060
```

```
        YAXIS = S1M * FLØAT(NPW-NLEV(1)-1)
```

```
        GØ TØ 4099
```

```
  406C IF(NLEV(2).LT.(NPW-1)/2) GØ TØ 4070
```

```
        YAXIS = S1M * FLØAT(NLEV(2)+1)
```

```
        GØ TØ 4099
```

```
  407C YAXIS = PWIDTH * .5
```

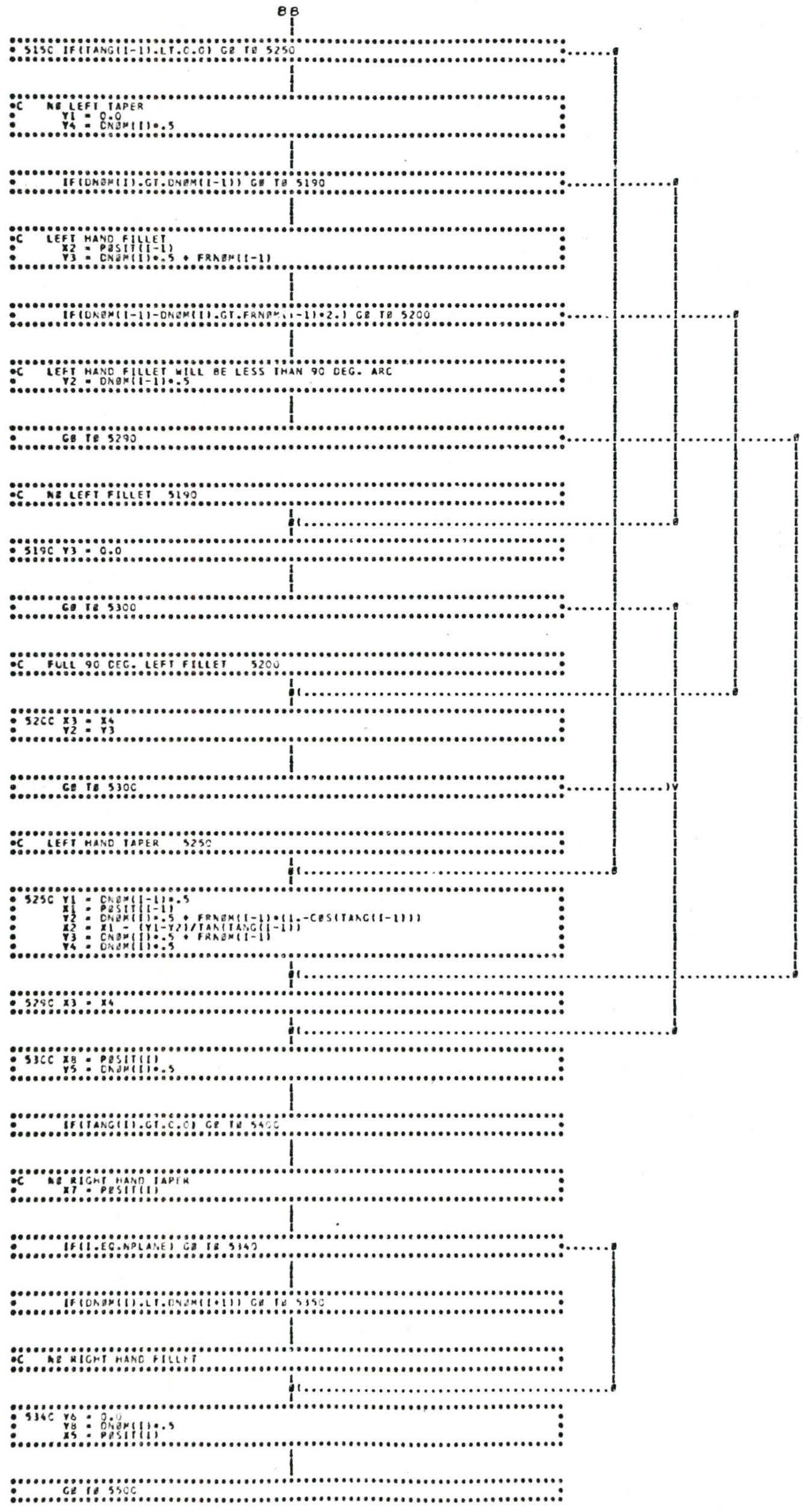
```
  4099 CØNTINUE
```

AA

AA

```
C
C   DRAW THE ØUTLINE ØF THE SHAFT       5001  TØ  5999
C

 5000 IF(PLTØ) GØ TØ 5002

      WRITE (6,5001)
 5001 FØRMAT(49HC      A PLØT IS NØT BEING PRØDUCED   --  NØ PLØT.///)

      GØ TØ 9000

C   PLTSW IS TURNED TRUE ØNLY IF THERE IS SØME PLØTTING DØNE.

 5002 IF(.NØT.PLTSW)  CALL IDPLØT
      PLTSW = .TRUE.
      CALL PLTIN(1./SCALE, 1./SCALE, -10.      /SCALE, -YAXIS/SCALE,
     1-10.      /SCALE, PØSIT(NPLANE)+10.      /SCALE, -YAXIS/SCALE,
     2(PWIDTH-YAXIS)/SCALE)
      CALL FØRMAT
      DEV = .01
      WLY(1) = DNØM(2)*.5
      CALL PLTLN(0.0,WLY(1),0.0,-WLY(1))

      DØ 5999 I=2,NPLANE

C
C   SET UP DRAWING SECTIØN  5100  TØ  5499
C
      X4 = PL(I)
      X5 = PR(I)

      IF(FRNØM(I).NE.0.0) GØ TØ 5140

      IF(TANG(I)) 5145,5140,5120

C   CHAMFER SECTIØN TØ THE LEFT ØF PLANE I   5120

 5120 IF(I.LT.NPLANE) GØ TØ 5125

      WRITE (6,5122)
 5122 FØRMAT(53H1  THE LAST PLANE CANNØT HAVE A PØSITIVE TAPER ANGLE.)

      GØ TØ 99990

 5125 Y4 = DNØM(I)*.5
      Y8 = DNØM(I+1)*.5
      Y5 = DNØM(I+1)*.5

      GØ TØ 5135

 5130 Y4 = DNØM(I-1)*.5
      Y5 = DNØM(I)*.5
      Y8 = DNØM(I)*.5

 5135 Y1 = 0.0
      Y3 = 0.0
      Y6 = 0.0
      X7 = PØSIT(I)
      X8 = PØSIT(I)
      MFER = 0
C   MFER IS USED TØ DETECT A RIGHT CHAMFER FRØM ITS SHØULDER PLANE.

      GØ TØ 5500

 5140 IF(MFER) 5130,5150,5130

 5145 MFER = 1
```

BB


BB
5150 IF(TANG(I-1).LT.0.0) GØ TØ 5250
C NØ LEFT TAPER
Y1 = 0.0
Y4 = DNØM(I)*.5
IF(DNØM(I).GT.DNØM(I-1)) GØ TØ 5190
C LEFT HAND FILLET
X2 = PØSIT(I-1)
Y3 = DNØM(I)*.5 + FRNØM(I-1)
IF(DNØM(I-1)-DNØM(I).GT.FRNØM(I-1)*2.) GØ TØ 5200
C LEFT HAND FILLET WILL BE LESS THAN 90 DEG. ARC
Y2 = DNØM(I-1)*.5
GØ TØ 5290
C NØ LEFT FILLET 5190
5190 Y3 = 0.0
GØ TØ 5300
C FULL 90 DEG. LEFT FILLET 5200
5200 X3 = X4
Y2 = Y3
GØ TØ 5300
C LEFT HAND TAPER 5250
5250 Y1 = DNØM(I-1)*.5
X1 = PØSIT(I-1)
Y2 = DNØM(I)*.5 + FRNØM(I-1)*(1.-CØS(TANG(I-1)))
X2 = X1 - (Y1-Y2)/TAN(TANG(I-1))
Y3 = DNØM(I)*.5 + FRNØM(I-1)
Y4 = DNØM(I)*.5
5290 X3 = X4
5300 X8 = PØSIT(I)
Y5 = DNØM(I)*.5
IF(TANG(I).GT.0.0) GØ TØ 5400
C NØ RIGHT HAND TAPER
X7 = PØSIT(I)
IF(I.EQ.NPLANE) GØ TØ 5340
IF(DNØM(I).LT.DNØM(I+1)) GØ TØ 5350
C NØ RIGHT HAND FILLET
5340 Y6 = 0.0
Y8 = DNØM(I)*.5
X5 = PØSIT(I)
GØ TØ 5500

```
C     RIGHT HAND FILLET  5350
 5350 Y6 = DNØM(I)*.5 + FRNØM(I)
      Y8 = DNØM(I+1)*.5

      IF(DNØM(I+1)-DNØM(I).GE.FRNØM(I)*2.) GØ TØ 5370

C     RIGHT HAND FILLET WILL BE LESS THAN 90 DEG. ARC
      Y7 = DNØM(I+1)*.5

      GØ TØ 5450

C     RIGHT FILLET ARC IS FULL 90 DEG.

 5370 X6 = X5
      Y7 = Y6

      GØ TØ 5500

C     RIGHT HAND TAPER  5400
 5400 Y8 = DNØM(I+1)*.5
      Y7 = DNØM(I)*.5 + FRNØM(I)*(1.-CØS(TANG(I)))
      Y6 = DNØM(I)*.5 + FRNØM(I)
      X7 = X8 - (Y8-Y7)/TAN(TANG(I))

 5450 X6 = X5

 5500 CØNTINUE
 5600 IF(Y1.GT.0.0)CALL PLTLN(X1,Y1,X2,Y2)
      IF(Y3.GT.0.0) CALL PLTARC(X2,Y2,X4,Y4,X3,Y3,DEV)
      IF(X5.GT.X4) CALL PLTLN(X4,Y4,X5,Y5)
      IF(Y6.GT.0.0) CALL PLTARC(X5,Y5,X7,Y7,X6,Y6,DEV)
      IF(X8.GT.X7)  CALL PLTLN(X7,Y7,X8,Y8)
      CALL PLTLN(X8,Y8,X8,-Y8)
      IF(X8.GT.X7)  CALL PLTLN(X8,-Y8,X7,-Y7)
      IF(Y6.GT.0.0) CALL PLTARC(X7,-Y7,X5,-Y5,X6,-Y6,DEV)
      IF(X5.GT.X4)  CALL PLTLN(X5,-Y5,X4,-Y4)
      IF(Y3.GT.0.0) CALL PLTARC(X4,-Y4,X2,-Y2,X3,-Y3,DEV)
      IF(Y1.GT.0.0) CALL PLTLN(X2,-Y2,X1,-Y1)
C     Y8 IS RECØRDED IN WLY TØ USE IN PLØTTING VERTICAL WITNESS LINES.
      WLY(I) = Y8
 5999 CØNTINUE

                    CC
```

```
CC

C
C     PLØT THE ANGULAR DIMENSIØNS           6000 TØ 6099
C
        SIP = SIM/SCALE
C     SIP IS THE DISTANCE BETWEEN LEVELS ØF DIMENSIØNING IN DATA UNITS.

        DØ 6099 N=2,NPLANE

        IF(YACL(N).EQ.0.0) GØ TØ 6099

C     ANGLE TØ BE PLØTTED FØR PLANE N
        CALL ANGPLT(TANG(N),PØSIT(N),YACL(N))

  6099 CØNTINUE

C
C     PLØT VERTICAL WITNESS LINES     6100 TØ 6199
C

        DØ 6199 L=1,2

        Y = FLØAT(NLEV(L)+1)*SIP
C     Y IS THE DATA UNITS DISTANCE TØ THE BASE LINE (L=1 FØR ABØVE.)
        S = FLØAT(1-2*(L-1))
C     S IS PØSITIVE WHEN ABØVE, NEGATIVE WHEN BELØW.

        DØ 6180 I=1,NPLANE

        IF(.NØT.LAB(L,I)) GØ TØ 6180

        DØ 6150 K=2,16,2

        IF(WL(L,K,I).EQ.0.0) GØ TØ 6170

        CALL PLTLN(PØSIT(I),(Y-WL(L,K-1,I))*S,PØSIT(I),(Y-WL(L,K,I))*S)

  6150 CØNTINUE

        K = 18
C     PLØT THE SEGMENT ØF THE WITNESS LINE CLØSEST TØ THE SHAFT.

  6170 CALL PLTLN(PØSIT(I),(Y-WL(L,K-1,I))*S,PØSIT(I),(WLY(I)+S2L)*S)

  6180 CØNTINUE

  6199 CØNTINUE

C
C     PLØT HØRIZØNTAL WITNESS LINES  6200 TØ 6299
C

        DØ 6290 I=1,30

        IF(WLH(I,1).EQ.0.0) GØ TØ 6299

        CALL PLTLN(WLH(I,2),-WLH(I,1),WLH(I,3),-WLH(I,1))

  6290 CALL PLTLN(WLH(I,2),WLH(I,1),WLH(I,3),WLH(I,1))

  6299 CØNTINUE

DD
```

DD

```
C
C   PLØT THE LENGTH DIMENSIØNS   6300 TØ 6499
C

      DØ 6499 L=1,2

C   PLØT THE LØCATING DIMENSIØNS (L=1) BEFØRE THE REFERENCE DIMENSIØNS.
      J = L-1

      DØ 6490 I=2,NPLANE

      IF(L.EQ.2) GØ TØ 6310

      IF(RTØL(I).LT.(-10.)) GØ TØ 6490

      IR = IREF(I)
      A = ABS(RNØM(I))
      ATØL = RTØL(I)

      GØ TØ 6320

C   TEST FØR LACK ØF REFERENCE DIMENSIØN.

 6310 IF(IREFO(I).EQ.1) GØ TØ 6490

      IR = IREFO(I)
      A = ABS(PØSIT(I) - PØSIT(IR))
      ATØL = -4.0

 6320 NL = MINO(IR,I)
      NR = MAXO(IR,I)

      IF(IH(I,L)) 6330,6330,6340

C   IH IS NEGATIVE FØR DIMENSIØNS PLØTTED BELØW THE SHAFT.

 6330 Y = FLØAT(NLEV(2)+1+IH(I,L)) + (-SIP)

      GØ TØ 6350

 6340 Y = FLØAT(NLEV(1)+1-IH(I,L)) + SIP

 6350 IF(LCØDE(I,L).GT.1) GØ TØ 6380

C   INTERNAL ARRØWS AND WRITING.
      CALL ARRØW(PØSIT(NL),Y,0)
      XS = PØSIT(NL) + (PØSIT(NR)-PØSIT(NL)-CLEN(A,ATØL,0,J,.FALSE.)
     1/SCALE)*.5
      CALL PLTPT(PØSIT(NL),Y)
      CALL PLTLN(PØSIT(NL),Y,XS,Y)
      CALL ALLPLT(XS,Y,A,ATØL,0,J,.FALSE.,XF)
      CALL PLTLN(XF,Y,PØSIT(NR),Y)
      CALL ARRØW(PØSIT(NR),Y,4)

      GØ TØ 6490

 6380 IF(LCØDE(I,L)-3) 6400,6420,6460
```

```
C     INTERNAL WRITING AND EXTERNAL ARRØWS.          6400
 6400 CALL PLTLN(PØSIT(NL)-S3L,Y,PØSIT(NL),Y)
      CALL ARRØW(PØSIT(NL),Y,4)
      CALL ALLPLT(PØSIT(NL)+(PØSIT(NR)-PØSIT(NL)-CLEN(A,ATØL,0,J,.FALSE.
     1)/SCALE)*.5,Y,A,ATØL,0,J,.FALSE.,XF)
      CALL ARRØW(PØSIT(NR),Y,0)
      CALL PLTLN(PØSIT(NR),Y,PØSIT(NR)+S3L,Y)

      GØ TØ 6490

C     INTERNAL ARRØW, AND EXTERNAL WRITING.     6420
 6420 CALL ARRØW(PØSIT(NL),Y,0)
      CALL ARRØW(PØSIT(NR),Y,4)

      IF(LØH(I,L).LT.0) GØ TØ 6440

C     WRITING IS ØN THE RIGHT HAND
      CALL PLTLN(PØSIT(NL),Y,PØSIT(NR)+S1L,Y)
      CALL ALLPLT(PØSIT(NR)+S1L,Y,A,ATØL,0,J,.FALSE.,XF)

      GØ TØ 6490

C     WRITING IS ØN THE LEFT HAND.    6440

 6440 CALL ALLPLT(PØSIT(NL)-S1L-CLEN(A,ATØL,0,J,.FALSE.)/SCALE,Y,A,ATØL,
     10,J,.FALSE.,XF)
      CALL PLTLN(XF,Y,PØSIT(NR),Y)

      GØ TØ 6490

C     EXTERNAL ARRØWS AND WRITING.    6460
 6460 CALL PLTLN(PØSIT(NL)-S2L,Y,PØSIT(NL),Y)
      CALL ARRØW(PØSIT(NL),Y,4)
      CALL ARRØW(PØSIT(NR),Y,0)
      CALL PLTLN(PØSIT(NR)+S2L,Y,PØSIT(NR),Y)

      IF(LØH(I,L).LT.0) GØ TØ 6480

C     RIGHT HAND WRITING
      CALL ALLPLT(PØSIT(NR)+S2L,Y,A,ATØL,0,J,.FALSE.,XF)

      GØ TØ 6490

C     LEFT HAND WRITING.    6480

 6480 CALL ALLPLT(PØSIT(NL)-S2L-CLEN(A,ATØL,0,J,.FALSE.)/SCALE,Y,A,ATØL,
     10,J,.FALSE.,XF)

 6490 CØNTINUE
 6499 CØNTINUE

EE
```

```
EE

C
C     PLØT DIAMETRAL DIMENSIØNS.       6500 TØ 6599
C

      DØ 6599 I=2,NPLANE

      IF(DL(I,2))6510,6599,6520

C     DIMENSIØN WRITTEN BELØW.        6510

 6510 Y = (FLØAT(NLEV(2)+1)+DL(I,2))*(-S1P)
      S = .5

      GØ TØ 6530

C     DIMENSIØN WRITTEN ABØVE.  6520

 6520 Y = (FLØAT(NLEV(1)+1)-DL(I,2)) * S1P
      S = -.5

 6530 CALL ARRØW(DL(I,1),DNØM(I)*.5,6)
      CALL ARRØW(DL(I,1),-DNØM(I)*.5,2)
      CALL PLTLN(DL(I,1),DNØM(I)*S,DL(I,1),Y)

      IF(DL(I,3).GT.0.0) GØ TØ 6570

C     LEFT HAND WRITING.
      XS = DL(I,1) - S2D - CLEN(DNØM(I),DTØL(I),2,0,DTYP(I))/SCALE
      CALL ALLPLT(XS,Y,DNØM(I),DTØL(I),2,0,DTYP(I),XF)
      CALL PLTPT(XF,Y)
      CALL PLTLN(XF,Y,DL(I,1),Y)

      GØ TØ 6599

C     RIGHT HAND WRITING.  6570

 6570 XS = DL(I,1) + S2D
      CALL PLTLN(DL(I,1),Y,XS,Y)
      CALL ALLPLT(XS,Y,DNØM(I),DTØL(I),2,0,DTYP(I),XF)

 6599 CØNTINUE

FF
```

```
FF

C
C   PLØT NØTES WITH LEADERS    6600  TØ  6699
C

      DØ 6699 I=2,NPLANE

      IF(DN(I,4).EQ.0.0) GØ TØ 6699

      S1 = SIGN(1.,DN(I,1))
C   S1 IS PØSITIVE FØR A RIGHT PØINTING LEADER.
      S2 = SIGN(1.,DN(I,3))
C   S2 IS PØSITIVE FØR DIMENSIØN PLACED ABØVE.
      J = IFIX(4.5 - S2*(2.-S1))
C   J IS THE CØDE FØR ARRØW DIRECTIØN (1 = SW, 3 = SE, 5 = NE, 7 = NW).
      XS = DN(I,1) * S1
      YS = DN(I,2) * S2
      CALL ARRØW(XS,YS,J)
      XF = XS - S1*(DN(I,3)*S2 - DN(I,2))
      CALL PLTLN(XS,YS,XF,DN(I,3))
      XD = XF - SIN*S1
      CALL PLTLN(XF,DN(I,3),XD,DN(I,3))

      IF(DN(I,4).LT.1.5) GØ TØ 6660

C   THE PRESENT NØTE IS A FILLET RADIUS.
      IF(S1.GT.0.0) XD = XD-CLEN(FRNØM(I),FRTØL(I),1,0,FRTYP(I))/SCALE
      CALL ALLPLT(XD,DN(I,3),FRNØM(I),FRTØL(I),1,0,FRTYP(I),XF)

      GØ TØ 6699

C   THE PRESENT NØTE IS A CHAMFER.       6660

 6660 CALL UNITTØ(XD,DN(I,3),X,Y)

      IF(S1.GT.0.0) GØ TØ 6680

C   THIS IS A RIGHT CHAMFER(A LEFT PØINTING LEADER).
      WIDTH = PØSIT(I+1) - PØSIT(I)

      GØ TØ 6690

C   THIS IS A LEFT CHAMFER(A RIGHT PØINTING LEADER).   6680

 6680 X = X - 1.02
      WIDTH = PØSIT(I) - PØSIT(I-1)

 6690 CALL CHPLT(X,Y,.7854,WIDTH)

 6699 CONTINUE

                                  GG
```

GS

```
C     END PLØT, MØVING PEN 2C INCHES PAST RIGHT END ØF SHAFT.
 8CCC CALL PLØT(PØSIT(NPLANE)*SCALE + 20.      ,0.0,-3)
      WRITE (6,8C05)
 8CC5 FØRMAT(32HC         A PLØT IS BEING PRØDUCED.////)
 9CCC CØNTINUE

      GØ TØ 99995

99990 WRITE (6,99991)
99991 FØRMAT(61HC  A FATAL ERRØR HAS BEEN CØMMITTED AND THE PRØGRAM ABØ
     1RTED.)

99995 WRITE (6,99996)
99996 FØRMAT(109HC  I      PØSIT      DNØM        DTØL     IREF   RNØM       RT
     1ØL        FRNØM       FRTØL     TANG    IREFQ   DTYP  FRTYP /)

      DØ 99997 I=2,NPLANE

99997 WRITE(6,99998)I,PØSIT(I),DNØM(I),DTØL(I),IREF(I),RNØM(I),RTØL(I),

99998 FØRMAT (X,I3,3F10.4,3X,I2,4F10.4,F7.2,4X,I2,2(5X,L3)/)
99999 CØNTINUE

      IF(LAST.EQ.1) GØ TØ 1

C     WRITE 999 TØ END ALL PLØTTING (IF THERE HAS BEEN ANY PLØTTING).
      IF(PLTSW)  CALL PLØT(0.0,0.0,999)

      STØP

      END
```

```
(ENTRANCE)

      FUNCTIØN CLEN(ANØM,ATØL,I,J,TYP)
C   CALCULATES THE LENGTH(IN INCHES) ØF THE WRITING(SIZE 2 = .12IN.HIGH)
C   THIS FUNCTIØN IS THE CØMPANIØN TØ SUBRØUTINE ALLPLT
C   I IS THE TYPE CØDE -- 0 FØR LENGTH, 1 FØR RADIUS, 2 FØR DIAMETER.
C   J IS ØNE FØR REFERENCE DIMENSIØNS, ZERØ ØTHERWISE.
C   ATØL MUST NØT BE -99.
C   TYP IS TRUE IF THIS IS A TYPICAL DIMENSIØN.
      LØGICAL TYP

      DØ 10 L=1,3

      LM1 = L-1

      IF(ANØM.LT.10.**LM1) GØ TØ 20

   10 CØNTINUE

      LM1 = 3
C   ANØM MUST BE LESS THAN 1000

   20 K = 9 + 7*LM1

      IF(I.EQ.1) GØ TØ 60

      IF(ATØL.LT.0.0) GØ TØ 30

      K = K + 69

      GØ TØ 40

C   CØDE FØR NUMBER ØF DECIMAL PLACES   30

   30 K = K + 7*IFIX(.5-ATØL)

   40 IF(J.EQ.1) GØ TØ 90

      IF(I.NE.2) GØ TØ 100

      K = K + 32

      GØ TØ 80

   60 IF(ATØL.GT.0.0) GØ TØ 70

      K = K + 7*IFIX(.5-ATØL) + 8

      GØ TØ 80

   70 K = K + 63
      IF(ANØM-ATØL.LT.1E-3) K=K+3

   80 IF(.NØT.TYP) GØ TØ 100

   90 K = K + 32

  100 CLEN = .02*FLØAT(K)

      RETURN

      END
```

(ENTRANCE)

```
      SUBRØUTINE ALLPLT(X,Y,ABØM,ATØL,I,J,TYP,XF)
C     PLØTS THE DIMENSIØN WRITING
C     THIS IS THE CØMPANIØN TØ FUNCTIØN CLEN AND CØDES ARE THE SAME.
C     I IS THE TYPE CØDE -- 0 FØR LENGTH, 1 FØR RADIUS, 2 FØR DIAMETER.
C     J IS ØNE FØR REFERENCE DIMENSIØNS, ZERØ ØTHERWISE.
C     ATØL MUST NØT BE -99.
C     TYP IS TRUE IF THIS IS A TYPICAL DIMENSIØN.
C     X AND Y GIVE THE LEFT HAND PØINT AT MID-LETTER HEIGHT, IN DATA UNITS
C      WHERE WRITING BEGINS.
C     XF(ØUTPUT) GIVES THE RIGHT HAND END ØF THE WRITING.
      DIMENSIØN A(2)
      DATA   PM/0776060606060/
      LØGICAL TYP
      CALL UNITTØ(X,Y,XS,YS)
      YS = YS - .06
      XS = XS + .08
      ANØM = ABØM
      IF(ANØM-ATØL.LT.1E-3) ANØM = 2.*ANØM

      IF(ANØM.LT.1.0) GØ TØ 100

C     WRITE THE NUMBERS TØ THE LEFT ØF THE DECIMAL IN ANØM(LT 1000)

      DØ 25 L=1,3

      IF(ANØM.LT.10.**L) GØ TØ 50

   25 CØNTINUE

C     IF ANØM .GE.1000 IT IS SET EQUAL TØ 0.0 AND WRITTEN AS SUCH
      ANØM = 0.0

   50 CALL ØCØNVF(ANØM,A,12,5-L)
      CALL LETTER(L,2,0,XS,YS,A(2))
      XS = XS + .14*FLØAT(L)
C     SEPARATE THE RADIUSES FØR WHICH ØNLY 3(MAXIMUM) DECIMAL PLACES ARE
C      WRITTEN   100

  100 IF(I.NE.1) GØ TØ 200

      IF(ATØL.LT.0.0) GØ TØ 110

      L = 3

      GØ TØ 120

C     ATØL GIVES CØDE FØR NUMBER ØF DECIMAL PLACES.   110

  110 L = IFIX(.5-ATØL)

  120 TEMP = ANØM + .5*10.**(-L)
      CALL ØCØNVF(TEMP,A,12,5)
      CALL LETTER(L+1,2,0,XS,YS,A(2))
      XS = XS + .10 + .14*FLØAT(L)

      IF(ATØL.LT.0.0) GØ TØ 190

      IF(ABØM-ATØL.GT.1E-3) GØ TØ 130

      CALL LETTER(4,2,0,XS,YS,4H MAX)
      XS = XS + .64

      GØ TØ 190

  130 CALL LETTER(1,2,0,XS,YS,PM)
      TEMP = ATØL + .0005
      CALL ØCØNVF(TEMP,A,12,5)
      CALL LETTER(4,2,0,XS+.16,YS,A(2))
      XS = XS + .70

  190 CALL LETTER(1,2,0,XS,YS,1HR)
      XS = XS + .16

      GØ TØ 350
```

```
C   CIAMETERS AND LENGTHS WITH UP TØ 4 DECIMAL PLACES.  200
  2CC IF(ATØL.LT.0.0) GØ TØ 21C
      L = 4
      GØ TØ 220
  21C L = IFIX(.5-ATØL)
  22C TEMP = ANØM + .5*10.**(-L)
      CALL ØCØNVF(TEMP,A,12,5)
      CALL LETTER(L+1,2,0,XS,YS,A(2))
      XS = XS + .10 + .14*FLØAT(L)
      IF(ATØL.LT.C.0) GØ TØ 3C0
      CALL LETTER(1,2,0,XS,YS,PM)
      TEMP = ATØL + .00005
      CALL ØCØNVF(TEMP,A,12,5)
      CALL LETTER(5,2,0,XS+.16,YS,A(2))
      XS = XS + .84
  3CC IF(I.NE.2) GØ TØ 375
      CALL LETTER(4,2,0,XS,YS,4H DIA)
      XS = XS + .64
  35C IF(.NØT.TYP) GØ TØ 4C0
      CALL LETTER(4,2,0,XS,YS,4H TYP)
      XS = XS + .64
      GØ TØ 400
  375 IF(J.EC.0) GØ TØ 4C0
      CALL LETTER(4,2,0,XS,YS,4H REF)
      XS = XS + .64
  4CC CALL INCHTØ(XS,YS,XF,DUMMY)
      RETURN
      END
```

```
(ENTRANCE)

      SUBROUTINE SKIP(JAB,I,K,GAP)
C   PRØVIDES SKIPS IN WITNESS LINES RECØRDED IN WL
      CØMMØN SCALE,DUMMY(3),WL(2,17,50)

      DØ 10 L=2,16,2

      IF(WL(JAB,L ,I).EQ.0.0) GØ TØ 50

   10 CØNTINUE

      WRITE(6,20)  JAB,I
   20 FØRMAT(43H1  WL DIMENSIØN IS NØT SUFFICIENT FØR JAB =,I3,5H, I =,
     1 I3)

      RETURN

   50 WL(JAB,L,I) = (.375*FLØAT(K) - GAP)/SCALE
      WL(JAB,L+1,I)=(.375*FLØAT(K) + GAP)/SCALE

      RETURN

      END
```

(ENTRANCE)

```
      SUBRØUTINE ANGPLT(TANG,X,Y)
C     PLØTS DIMENSIØN LINES, WITNESS LINES, AND NUMBERS FØR ANGLES
      CØMMØN /BLAST/ DEV
      S1 = TANG/ABS(TANG)
      S2 = Y/ABS(Y)
      CALL UNITTØ (X,Y,XC,YC)
      AX = CØS(TANG)*S1
      AY = S1*SIN(TANG)*S2
      CALL PLØT(XC+.1*AX,YC+.1*AY,3)
      CALL PLØT(XC+.87*AX,YC+.87*AY,2)
      IF(S1*TANG.LT..79) GØ TØ 10
C     INTERNAL ARRØWS
      S3 = 1.0
      BX = CØS(TANG*.5)*S1*.75+XC
      BY = S1*SIN(TANG*.5)*S2*.75 + YC
      CY = BY - .12*S2
      DY = BY - .05*S2
      CX = SQRT(.5625 - (CY-YC)**2)*S1 + XC
      DX = SQRT(.5625 - (DY-YC)**2)*S1 + XC
      IF(S1*S2) 50,50,60
C     EXTERNAL ARRØWS     10
   10 S3 = -1.0
      BX = CØS(TANG + S1*.3)*S1*.75 + XC
      BY = S1*SIN(TANG + S1*.3)*S2*.75 + YC
      CX = .92106*S1*.75 + XC
      CY = -.38942*S2*.75 + YC
      IF(S1*TANG.LT..35) GØ TØ 20
C     INTERNAL NUMBERS
      DX = CØS(TANG*.5)*S1*.75 + XC
      DY = S1*SIN(TANG*.5)*S2*.75 + YC
      GØ TØ 30
C     EXTERNAL NUMBERS     20
   20 DX = BX
      DY = BY + S2*.07
   30 IF(S1*S2) 60,60,50
C     ARGUMENTS FØR ARC ARE ØRDERED TØ MAINTAIN CØUNTER-CLØCKWISE DRAWING
   50 CALL ARC(XC+.75*AX,YC+.75*AY,BX,BY,XC,YC,DEV)
      CALL ARC(CX,CY,XC+.75*S1,YC,XC,YC,DEV)
      GØ TØ 70
   60 CALL ARC(BX,BY,XC+.75*AX,YC+.75*AY,XC,YC,DEV)
      CALL ARC(XC+.75*S1,YC,CX,CY,XC,YC,DEV)
   70 CALL ARRØWA(XC+.75*S1,YC,3.1416-S2*S3*(1.5708-S1*.12))
      CALL ARRØWA(XC+.75*AX,YC+.75*AY,3.1416+S2*(TANG+S3*(1.5708-S1*
     1 .12)))
      CALL CHPLT(DX-S1*.06-.24, DY-S2*.01,ABS(TANG),0.0)
      RETURN
      END
```

```
                                   (ENTRANCE)
                                       I
                                       I
**************************************************************************
*        SUBRØUTINE CHPLT(X,Y,ANG,WIDTH)                                 *
*C    PLØTS THE LETTERS FØR AN ANGLE IF WIDTH .EQ. 0  AND THØSE FØR A     *
*C      CHAMFER NØTE ØTHERWISE.  ARGUMENTS IN PLØTTER UNITS INDICATE LEFT *
*C        MID-PØINT ØF 1ST WØRD. ANGLE GIVEN IN RADIANS.                  *
*        DIMENSIØN C(2)                                                  *
*        XS = X + .08                                                    *
*        YS = Y - .06                                                    *
**************************************************************************
                                       I
                                       I
                                       I
**************************************************************************
*        IF(WIDTH.EQ.0.0) GØ TØ 10                                       *......Ø
**************************************************************************      I
                                       I                                        I
                                       I                                        I
                                       I                                        I
**************************************************************************      I
*        TEMP = WIDTH                                                    *      I
*        CALL ØCØNVF(TEMP,C,9,2)                                         *      I
*        CALL LETTER(3,2,0,XS,YS,C(2))                                   *      I
*        CALL LETTER(1,2,0,XS+.38,YS,1HX)                                *      I
*        XS = XS + .54                                                   *      I
**************************************************************************      I
                                       I                                        I
                                       Ø(.......................................Ø
                                       I
**************************************************************************
*     10 A = ANG*57.29578 + .5                                           *
*        CALL ØCØNVF(A,C,12,3)                                           *
*        CALL LETTER(2,2,0,XS,YS,C(2))                                   *
*        CALL LETTER(1,1,0,XS+.28,YS+.06,1HØ)                            *
**************************************************************************
                                       I
                                       I
                                       I
**************************************************************************
*        RETURN                                                          *
**************************************************************************


**************************************************************************
*        END                                                             *
**************************************************************************
```

```
                              (ENTRANCE)
                                  I
                                  I
*************************************************************************
*      SUBRØUTINE ARRØWA(X,Y,A)                                         *
*C   PLØTS AN ARRØW HEAD PØINTING IN DIRECTIØN A, WITH ITS PØINT AT X,Y *
*C   X,Y - IN PLØTTER UNITS      A - RADIANS IN USUAL SENSE             *
*      X1 = .15*CØS(A+.1652244) + X                                     *
*      X2 = .15*CØS(A-.1652244) + X                                     *
*      Y2 = .15*SIN(A-.1652244) + Y                                     *
*      Y1 = .15*SIN(A+.1652244) + Y                                     *
*      CALL ARRA(X1,Y1,X2,Y2,X,Y)                                       *
*************************************************************************
                                  I
                                  I
*************************************************************************
*      RETURN                                                           *
*************************************************************************

*************************************************************************
*      END                                                              *
*************************************************************************
```

```
(ENTRANCE)

      SUBROUTINE ARROW(X,Y,I)
C     DRAWS ARROW HEAD IN ONE OF EIGHT DIRECTIONS, ACCORDING TO I, WITH
C     ITS TIP AT POINT X,Y.
C     X,Y, IN DATA UNITS.  I=0 GIVES A LEFT-POINTING ARROW, 1 A SOUTHWEST,
C     2 A DOWNPOINTING,3 A SOUTHEAST-POINTING, ETC.
      CALL UNITTO(X,Y,XP,YP)

      IF(MOD(I,2)) 10,10,3C

   10 IF(MOD(I,4).EQ.2) GO TO 20

C     I=ZERO OR FOUR
      X1 = XP + .07398*FLOAT(2-I)
      Y1 = YP + .02465
      Y2 = YP - .02465
      CALL ARRA(X1,Y1,X1,Y2,XP,YP)

      RETURN

C     I=TWO OR SIX  20

   2C Y1 = YP + .07398*FLOAT(4-I)
      X1 = XP + .C2465
      X2 = XP - .02465
      CALL ARRA(X1,Y1,X2,Y1,XP,YP)

      RETURN

C     I IS ODD     30

   30 C = .15
      IF(I.GT.4) C=-.15
C     C IS POSITIVE FOR I = 1 OR 3
      Y1 = YP + C*.81378
      Y2 = YP + C*.58118
      IF(MOD(I+1,4).EQ.0) C=-C
C     C IS POSITIVE FOR I = 1 OR 7
      X1 = XP + C*.58118
      X2 = XP + C*.81378
      CALL ARRA(X1,Y1,X2,Y2,XP,YP)

      RETURN

      END
```

```
                              (ENTRANCE)
                                  I
                                  I
*****************************************************************
*       SUBRØUTINE ARRA(X1,Y1,X2,Y2,XP,YP)                      *
*       CALL PLØT(XP,YP,3)                                      *
*       CALL PLØT(X1,Y1,2)                                      *
*       CALL PLØT(XP,YP,3)                                      *
*       CALL PLØT(X2,Y2,2)                                      *
*       CALL PLØT(X1,Y1,2)                                      *
*****************************************************************
                                  I
                                  I
                                  I
*****************************************************************
*       RETURN                                                  *
*****************************************************************


*****************************************************************
*       END                                                     *
*****************************************************************
```

```
(ENTRANCE)
    I
    I
      SUBRØUTINE FØRMAT
      CØMMØN  SCALE,PWIDTH,SN(2)
      CALL PLØT(1.625,2.625,3)
      CALL PLØT(2.625,2.625,2)
      CALL PLØT(2.625,7.625,2)
      CALL PLØT(1.625,7.625,2)
      CALL PLØT(1.625,2.625,2)
      CALL PLØT(2.125,2.625,3)
      CALL PLØT(2.125,5.625,2)
      CALL PLØT(1.625,5.625,3)
      CALL PLØT(2.625,5.75C,2)
      CALL PLØT(1.625,5.75C,2)
      CALL PLØT(2.625,5.625,2)
      CALL PLØT(1.625,5.625,2)
      CALL PLØT(2.125,5.75C,3)
      CALL PLØT(2.125,7.625,2)
      CALL ØCØNVF(SCALE,SC,6,3)
      CALL LETTER(10,4,90,2.0,3.0,10HSHAFT NAME)
      CALL LETTER( 5,4,90,2.0,6.125,5HSCALE)
      CALL LETTER(12,4,90,2.5,3.C,SN)
      CALL LETTER( 6,4,90,2.5,6.125,SC)
    I
    I
    I
      RETURN

      END
```

(ENTRANCE)

```
      SUBROUTINE PLTARC (X1,Y1,X2,Y2,XC,YC,DEV)
C     MODIFICATION OF PLTARC TO ALLOW FOR A CENTER PT OFF THE PAGE
      COMMON  SCALE,YAXIS
      CALL UNITTO (X1,Y1,XX1,YY1)
      CALL UNITTO (X2,Y2,XX2,YY2)
      XXC = XC*SCALE + 10.
      YYC = YC*SCALE  +  YAXIS
      CALL ARC (XX1,YY1,XX2,YY2,XXC,YYC,DEV)
```

```
      RETURN
```

```
      END
```

APPENDIX VI

Condensed Input Instructions

This appendix has been provided for the convenience of the reader who wishes to try using the program.

Figure 1 is a one-page condensation of Section III, pp. 5 - 14, "The Description of the Input."

Finally, a key-punch form has been designed which follows the input format described on page 14 of the text.

NOTES FOR DATA INPUT TO SHFTDW

I. Definition of Planes: Every dimension is associated with an adjacent plane of the shaft. These planes are perpendicular to the center line of the shaft. They appear as vertical lines across the shaft in the drawing. (In accordance with standard drawing practice, a plane appears only where there is a sharp change in the surface contour.)

II. Numbering of the Planes: With one exception, each plane is assigned an arbitrary, unique, two-digit number in the input. That exception is the leftmost plane which defines the left end of the shaft. That plane is assumed to have the number zero and has no dimensions associated with it.

III. The Dimension Parameters: Each plane (except plane zero) has associated with it;

I	--	the plane number (integers only)
DNOM	--	the nominal diameter of the section to the left
IREF	--	# of the plane from which it is located
RNOM	--	# the nominal distance from plane IREF to plane I (RNOM is negative when plane I is to the left of plane IREF)
FRNOM	--	a nominal fillet radius
IREFQ	--	# of the plane to which a reference dimension labelled REF appears. IREFQ can be left blank. Therefore, -1 must be used instead of zero for a reference dimension to the leftmost plane.
TANG	--	a taper angle (positive for diameter increasing to the right). May be left blank.

Each nominal dimension has with it a tolerance. These are DTOL, RTOL, FRTOL. They are of the equal (plus or minus) type, i.e., DNOM $\pm$ DTOL, etc.

IV. Chamfers: The chamfer has been treated as a unit. One can be added to or removed from a shoulder without altering the other parameters. In the case of the right hand shoulder, this necessitates one exception to the rule that DNOM is the diameter to the left of the plane. In this case when the chamfer is present DNOM of the shoulder plane is the diameter to the left of the chamfer plane.

Chamfer planes should always be located relative to adjacent shoulder planes.

V. Blanks: No field may be left blank in the input unless mentioned above or on the input sheet.

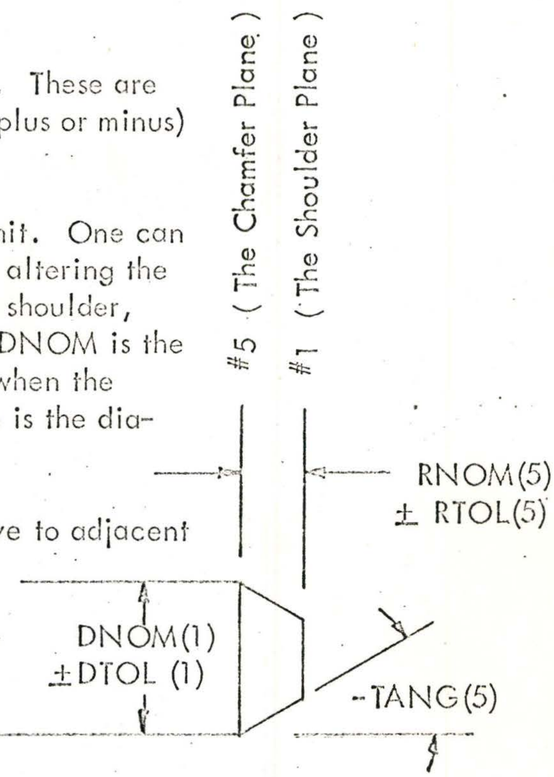


FIGURE 1

I		DIAMETER				IREF		LOCATING LENGTH				FILLET RADIUS				IREFQ		TAPER ANGLE		LAST
		NOMINAL		TOLERANCE				NOMINAL		TOLERANCE		NOMINAL		TOLERANCE						

↑ ARROWS ↑ SHOW COLUMNS ↑ IN WHICH ↑ DECIMAL ↑ IS ↑ ASSUMED ↑

Duplication of Dimensions: For a duplicate Diameter or Fillet Radius the nominal dimension may be omitted in which case the number of the duplicated plane should be entered in the Tolerance Field. The duplicated planes must preceed the duplicating plane.

Blank Fields: No blank fields are permitted except those mentioned specifically on this form.

Standard Tolerances: .X = ±.060

.XX = ±.030

.XXX = ±.010

This convention can be utilized by giving -n instead of a tolerance, where n is the number of digits to the right of the decimal. (one, two, or three, as above)

Chamfer Planes: Have a taper angle but the fillet radius is left blank. Normally Diameter is also blank. If diameter given then this is the shoulder Diameter and the Locating Length Dimension must be blank. The Fillet Radius must also be blank on the right most plane.

IREFQ is blank unless a reference dimension is desired between I and IREFQ.

LAST = 1 on last card of a shaft

LAST = 2 on last card of last shaft

LAST is blank otherwise.