SHARE PROGRAM LIBRARY AGENCY

SHARE

PROGRAM NUMBER

234004

University of Miami

1365 MEMORIAL DRIVE - CORAL GABLES, FLORIDA
(305) - 284-6257

SHARE PROGRAM LIBRARY SUBMITTAL FORM

SHARE PROGRAM LIBRARY AGENCY
Triangle Universities Computation Center
Post Office Box 12076
Research Triangle Park, North Carolina
27709 USA

SPLA CONTROL NUMBER:

This form should be completed and submitted with the program package to the SHARE Program Library Agency at the address shown above. Standards and instructions for submitting programs are in the "SHARE Program Library Standards Manual".

(1) Program Number (to be filled in by SPLA) 360D-23.4.004

(2) System Type (machine) .... 360/370 - 50 & up

(3) Search Key ....

(4) Programming Language .... Fortran IV

(5) Author's Name and Address .... Quint Rygh
Douglas Aircraft Co.
Long Beach, CA 90846

(6) Direct Inquiries to Name and Address ..... T. J. Casey, CI-648 (2-56)
(if different than Author)
Douglas Aircraft Co.
3855 Lakewood Blvd.
Long Beach, CA 90846

(7) Title of Program .... 360 APT - V4M3/SSX3A/SSIP

(8) Submitter's Installation Membership Code........ DA

(9) Submitter's Own Program Identification and Suffix(Optional)... APT/NC 1

(10) Primary Subject Code.................................. 23 4

(11) Operating or Monitor System Required OS or OS/VS - any version

(12) New or Revision Code (if revision, show prior Program Number in Item 1).. New

(13) Year Completed.. 1974

(14) Date of Submittal...................................... 4-1-74

(15) Documentation (number of original pages submitted)............ 100

(16) Abstract (should contain sufficient information for a reader to determine the value of the program). Listed on the reverse side of this form are subjects which may serve as a guide for a descriptive abstract.

360D-23.4.004

APTNC1 IMPLEMENTATION INSTRUCTIONS

I. MAGNETIC TAPE DESCRIPTION:

9-TRACK, 800 BPI, NON-LABEL. FOUR FILES CONTAIN UNLOADED PARTITIONED DATA SETS AS FOLLOWS:

FILE 1: SYS4.SSP.X3A.LM, LOAD MODULES FOR THE SS PROCESSOR. MEMBERS ARE ASECT0, ASECT1, ASECT2, ASECT3, ASECT4.

FILE 2: SYS4.SSP.X3A.SM, SOURCE MODULES OF SSX3A PROCESSOR WHICH WERE CHANGED FOR THIS PROCESSOR. MEMBERS ARE APT107, APT108, GETSCA, LINEAR, MESH, SSPLIN, SSPRE.

FILE 3: SYS4.SSP.X3A.UPDOVLY, UPDATES TO V4M3 SECTION I WHICH WERE NEEDED TO IMPLEMENT SS ON V4M3, AND OVERLAY STRUCTURES FOR ASECT1 AND ASECT2. MEMBERS ARE HOOKS, SEC1OVLY, SEC2OVLY.

FILE 4: SYS4.SSP.X3A.TEST, NINE TESTS FOR VERIFYING IMPLEMENTATION AND DEMONSTRATING CAPABILITIES. MEMBERS ARE TEST1, TEST2, TEST2A, TEST3, TEST4, TEST5, TEST6, TEST7, TEST8.

II. IMPLEMENTATION:

THE FOLLOWING CONTROL STATEMENTS ILLUSTRATE THE IMPLEMENTATION OF THE PROGRAM PACKAGE:

```
//TODISK  EXEC PGM=IEHMOVE
//SYSPRINT  DD SYSOUT=A
//SYSUT1    DD UNIT=2134,VOL=SER=UTIL03,DISP=OLD
//DISK      DD UNIT=2314,VOL=SER=APTPAK,DISP=OLD
//TAPE      DD UNIT=2400,LABEL=(,NL),VOL=(,RETAIN,SER=APTNC1),
//             DISP=OLD,DCB=(RECFM=FB,LRECL=80,BLKSIZE=800,DEN=2)
//SYSIN     DD *
 COPY FROMDD=TAPE,TO=2314=APTPAK,                                    *
               FROM=2400=(APTNC1,1),PDS=SYS4.SSP.X3A.LM
 COPY FROMDD=TAPE,TO=2314=APTPAK,                                    *
               FROM=2400=(APTNC1,2),PDS=SYS4.SSP.X3A.SM
 COPY FROMDD=TAPE,TO=2314=APTPAK,                                    *
               FROM=2400=(APTNC1,3),PDS=SYS4.SSP.X3A.UPDOVLY
 COPY FROMDD=TAPE,TO=2314=APTPAK,                                    *
               FROM=2400=(APTNC1,4),PDS=SYS4.SSP.X3A.TEST
/*
```

III. OPERATION:

NORMAL JCL FOR NC360 APT EXECUTION IS USED FOR THE EXECUTION OF THIS PROCESSOR. REFER TO IBM NC360 APT OPERATION MANUAL H20-0331 FOR V4M3.

360D-23.4.004

PART PROGRAMMING DOCUMENTATION

FOR

V4M3/SSX3A/SSIP

SHARE - APT/NC PROJECT

APRIL 1974

FOREWORD

Much of what appears in the pages which follow has been harvested from a field cultivated by the Sculptured Surface Project of CAM-I. Permission to use was graciously extended. Acknowledgment is also made of the sizeable efforts in this endeavor by the following companies: Rohr Industries, Bell Helicopter, McDonnell Douglas Astronautics, Teledyne Ryan Aeronautical and Douglas Aircraft.

INTRODUCTION

The APT/NC Project met for the first time at the SHARE Interim Meeting in Houston, 5/22-23/72. In discussing goals the proposal was made and warmly accepted that the Project adopt as one of its immediate goals the incorporation of CAM-I's Sculptured Surface Package into IBM System/360 APT. The feeling of the members was that the CAM-I package was indeed good and one which, with further refinement, might well accomodate the needs of the NC community but it was suffering from lack of exposure. It was hoped that by attaining the above goal this situation could be rectified.

The merger, V4M3 and SSX2A, was effected with dispatch and distributed to some 10 or 12 users. Extensive testing was performed between August, 1972 and August, 1973. Many benefits were reaped, many more were envisioned but standing in the way was an obstacle, the V4M3 ARELEM and its inability to handle a sculptured surface. At the Project's Miami Meeting, 8/13-17/73, it was decided to replace this faulty member with an interim or temporary member which would do the job. So it was that the Sculptured Surface Interim Processor (SSIP) came to be. It is not a full blown, production-type ARELEM but it is a package which will allow Sculptured Surface R & D to continue in an efficient manner. Those who have used it thus far have marvelled at its simplicity and praised its reliability. They have commended its adherence to the concept of maximum processing or, in other words, its ability to recover and continue in the face of minor programming incongruities. Most of all they are amazed at its efficiency, as much as ten times faster than its predecessor.

The sections which follow are intended for the part programmer. Section 1, basically, is Chapter II of the SSX3 documentation released by CAM-I in April, 1973. The modifications necessary were few. Section 2 documents SSIP. Section 3 consists of a group of nine test cases which may be used to verify implementation and demonstrate capabilities.

TABLE OF CONTENTS

SECTION 1

PART PROGRAMMER GUIDE

TO

SSX3A/SSIP

1. PART PROGRAMMERS GUIDE TO SSX3A/SSIP

The following describes concepts, language and capabilities of sculptured surfaces for potential users of the system.

1.1 Basic Structure of Sculptured Geometry

There are a large number of sculptured definitions available in the present system. These include two major types of geometry -- a synthetic curve and a sculptured surface. While many alternate definitions are available, the underlying structure of the resulting geometry is very similar in all cases. Most geometry proceeds from programmer supplied points (for positional control) and, if desirable, vectors (for slope control).

1.1.1 Synthetic Curve -- Segment Structure

The synthetic curve structure consists of a set of interlinked cubic curve segments. The part programmer, however, does not have to know all about the cubic form. A variety of input formats allow him to create this structure in ways which are both useful and understandable. In a structural sense, the synthetic curve is similar to a Tabulated Cylinder. However, there are two major differences--the synthetic curve is a true space curve rather than a cylindrical surface--and it has the capability to include linear, circular and conic curves as well as curves which are twisted in space (i.e., do not lie in a plane). In addition, a synthetic curve includes cross and normal vector constraints which can be used in subsequent surface definitions.

At present, synthetic curves have no direct type of tool control available. In the future, some type of control may be available but the concepts are presently awaiting firm definition. In the SSX3 system, the only way in which a synthetic curve can be milled is by defining a sculptured surface from it--this consisting of either a surface of revolution or a cylinder. This sculptured surface may then be used as a normal APT surface for purposes of guiding a tool.

1.1.2 Sculptured Surfaces

A large variety of definitions is available to part programmers of sculptured surfaces. All of these definitions create a surface which is defined by an interconnected structure of bicubic surface patches. These patches are basically each four-sided surfaces of limited flexibility. They can capture points of inflection and can describe quadric forms as well as more general sculptured shapes.

Surfaces of revolution and cylindrical and ruled shapes are also within the range of describable geometry.

1.1.3 Extensions Beyond Surface Boundaries

Since sculptured surfaces are bounded, a basic problem which occurs is that of interpolating the surface beyond its boundaries. Generally speaking, the surface bounds are specified by the part programmer in his definition of the surface. However, it is of some utility to the programmer to be able to ignore the 'natural' bounds when positioning a tool. For this reason a convention has been adopted that an infinite ruled surface is extrapolated beyond the bounds of all sculptured surfaces (see Figure 1). The rulings of this surface have directions identical to the natural surface tangents as they cross the boundaries. A doubly ruled surface is extrapolated beyond the corner sections of the structure. At present, the ruled extrapolation is designed for patch structures which exhibit four or fewer sides. These extensions extend infinitely. Consequently, if an APT programmer should miss a check surface in a tool motion command, the tool will move out along the sculptured surface until a preset maximum distance is exceeded.

1.1.4 Part Programmer Oriented Definitions

The fundamental structure of sculptured geometry has been briefly described. What makes the concept truly useful to a part programmer, is a large array of simple-to-use definitions. The principal idea which is used for defining a synthetic curve or a sculptured surface is a stream of input points which lie on the desired shape. These points may be developed from a coordinate measuring device which traces a model in a systematic direction, from design equations, or else from design specified geometry. Under the present APT structure, data which comes from the latter two sources can be applied more easily. However, this data must be encoded (usually on computer cards) in an APT readable format. For example, in order to define a synthetic curve through an ordered set of points (see Figure 2a), the part programmer must present these points in the proper order to the synthetic curve processor by a statement, such as

```
C = SCURV/SPLINE, P1, P2, P3, P4, P5
```

In English, the programmer is saying, define a space curve called C that moves smoothly through the consecutive points P1, P2, P3, P4 and P5. This processor then interpolates cubic arcs between consecutive points such that the overall profile is smooth. Hence four interconnected cubic arcs are generated. The programmer will receive a printed analysis of each of the arcs (curvature, tangents, etc) with his normal APT output.

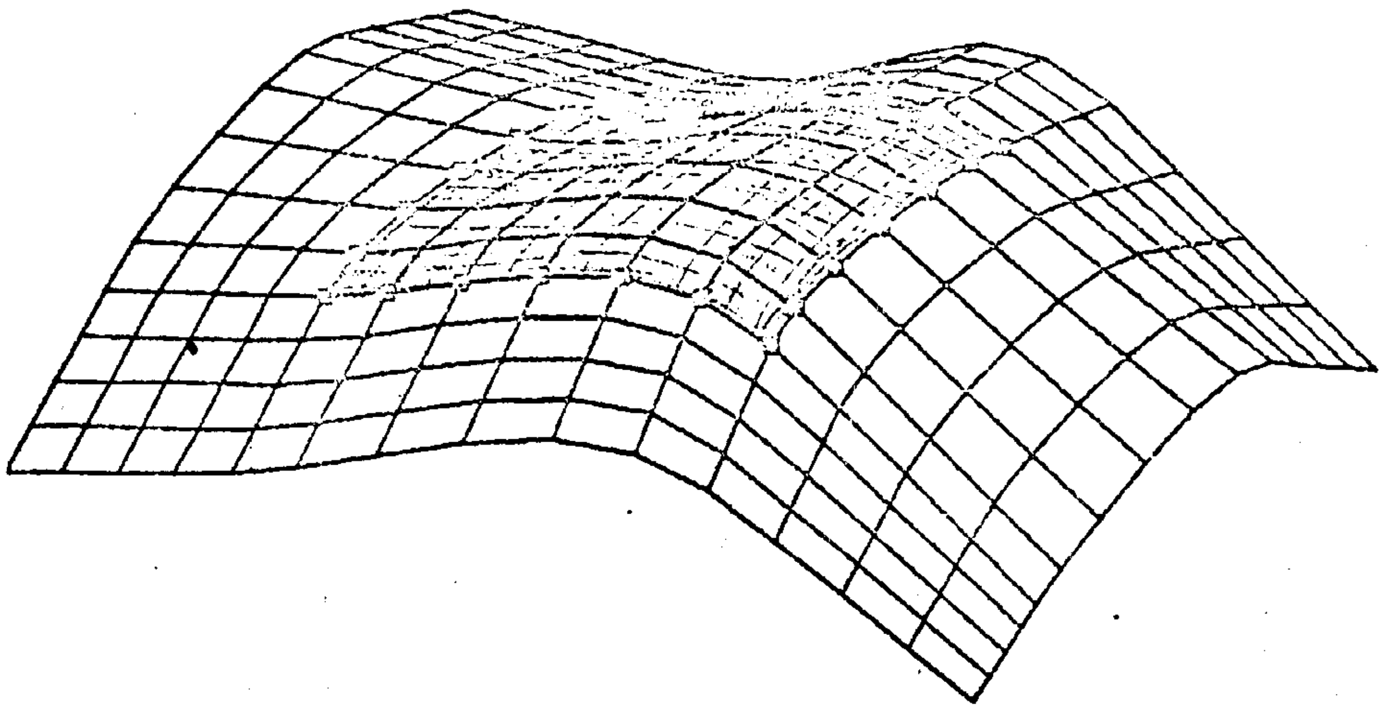


Figure 1. Example of Ruled Extensions Beyond the Boundaries of a Sculptured Surface

The principal method of defining a surface is by a network of streams of points (Figure 2b). This, however, imposes the added difficulty that points have to be coordinated in two directions simultaneously. However, the input of a network of points (Figure 2c) creates a patch structure where a bicubic patch is interpolated through the space marked by every four adjacent points. Therefore, the part programmer does not need to be acquainted with the details of patch canonical forms. However, since the resulting surface is analyzed patch-by-patch on the print file, a programmer's use of the system is enhanced by a knowledge of the structure of its geometry.

1.1.5 Splining Techniques

One of the important methods for determining a continuous curve or surface in space through discrete points is by splining techniques. Traditionally, ship loftsmen produced splines by passing an elastic beam through fixed positions on a loft floor. The shape assumed by the beam is a spline fit to the fixed positions. Mathematically, the spline position is an elastic curve with the least stored energy which passes through the fixed points. This condition is approximately realized whenever the interconnecting cubic segments have similar radii of curvature at their juntions. The spline concept can also be generalized so that the input points will move in such a way as to achieve a final shape which is more 'relaxed'. This concept is effected by attaching springs to the fixed starting positions of the points. The spline shape is then balanced (see Figure 3) between minimum stored energy of the elastic beam and the stored energy of the springs which are stretched from their fixed pivot points. When this concept is applied to input with inherent 'ripples', these ripples tend to be reduced into an overall 'smoother' configuration.

1.1.6 Vector Constraints

A part programmer can impart additional slope information with discrete points. He does this by inputting vector constraints at any desired input point (a vector indicated by three coordinates is a direction in space). In the case of a curve, tangent or normal vector constraints are applicable at any point, while in the case of a surface-tangent, normal or crossspline constraints may be applied. In any case, a vector constraint is treated as an essential condition. That is, the final curve or surface will possess the direction as indicated by the programmers's tangent or normal constraint. In fact, the vector constraint, inserted in the middle of a spline, will actually split a spline into two separate splines. There is a possibility of conflict if a part programmer should enter conflicting vector constraints, such as tangent and normal constraints at the

Figure 2a. Programmer's Input Points

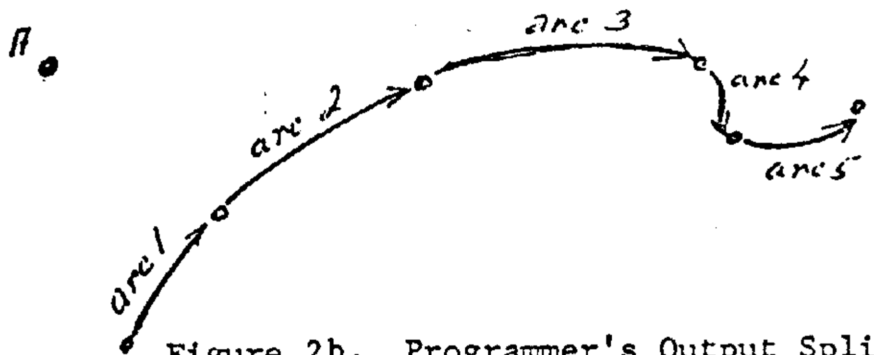


Figure 2b. Programmer's Output Spline Curve

Figure 2c. Programmer's Input Points

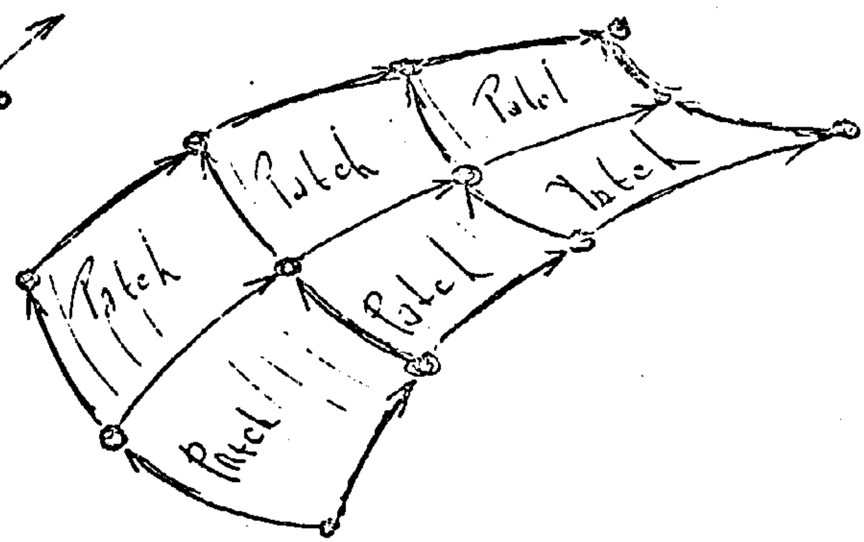


Figure 2d. Programmer's Output Surface

same point which are not actually perpendicular. In this case, the normal vector takes precedence and the tangent vector is forced into the plane of the normal vector.

1.1.7 Concept of Smoothing with Weights and Limits

As mentioned in Section 1.4, a concept of smoothing applies to splining which will actually move the input points into an overall smoother configuration. This concept is effected by attaching a 'weight' or degree of certitude to each point which controls the spring tension there. If a point is to remain fixed, a weight of 1.0 (or certainty) is attached to it and the spline must move through the point. If the weight is reduced to a value between '0' and '1', then the point will tend to move more in line with other points of the spline (see Figure 3). A further capability is available which will limit the maximum movement of any point during splining. This is called the 'LIMIT' constraint. The LIMIT factor is a spherical radius scribed around an input point. Whenever weights less than 1 have been applied, a point is restricted to moving within the confines of the limiting sphere placed around the input point. It should be noted that the splining technique applies only to curves (not complete surfaces) at the present time. Consequently, for a 'MESH' surface, which is defined by a criss-cross pattern of splines and cross splines, the concept of smoothing applies only to spline curves in the major spline direction. Also whenever weights are applied together with vector constraints, then points are 'moved' or 'relaxed' in a first pass and slopes are adjusted in a second pass.

1.1.8 Error Diagnostics

Number	Description
129	Incorrect amount of data (NUMPTS/SPLINE)
133	Illegal vocabulary code
	KERR=1 Illegal word in definition
	KERR=2 Surface type (PATCH, MESH, SPLINE, CURSEG)
	KERR=3 Mesh type no match (XYZ, XYPLAN, etc)
	KERR=4 Illegal word in definition
4002	Word count no good PATCH definition
4003	Plus or minus not found in expected position
5000	Word count no good
5001	Surface too large
5016	Error found when processing PATCH in APT 107
5017	Error found when processing MESH in APT 107
5018	Error found when processing SCURV in APT 108

1.1.9 Size Limits on Sculptured Geometry

There is no practical limit, small or large, to the physical size of sculptured geometry. There are, however,

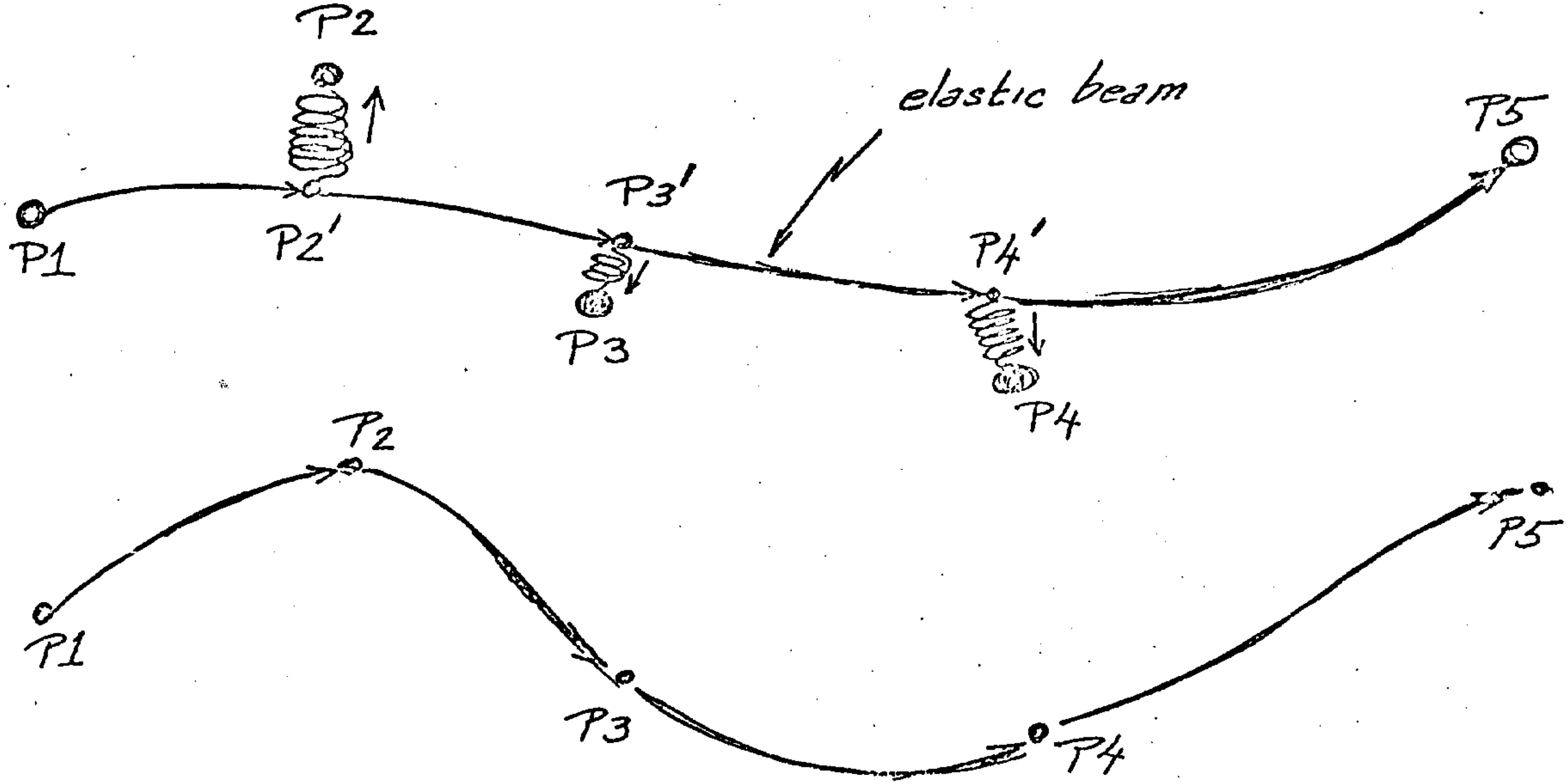


Figure 3. Splining -- With Springs and Without Springs

limits on the amount of physical data which can be stored and used for describing geometry. Basically, surfaces are composed of patches. Each patch is defined by a combination of sixteen points and vectors and each of these are described by three numbers (X, Y, and Z coordinates). In round numbers, a patch requires about 50 numbers for its analytic description--so that if a surface were defined by 50 patches, then 2500 numbers have to be stored.

Recognizing this problem SSIP had provided a compressed storage format which allows for the processing of sculptured surfaces as large as 66 patches within a 1000 word table. This could be expanded in the future, but appears to be adequate at this present period in the development of sculptured surfaces.

1.1.10 Tool Control

The usual objective in defining geometry in APT, is to reproduce it by using that geometry to control the motion of a cutter. Sculptured geometry is composed of two geometries -- surfaces and synthetic curves. Synthetic curves are unique in APT in that they represent a one-dimensional space curve rather than a two-dimensional surface. No concept has been implemented for using this entity directly for tool control -- in the future a concept will most likely be available. Sculptured surfaces, on the other hand, should be treated as other APT surfaces for tool control. In fact, this feature of full APT tool control is the very heart of the sculptured capability in APT.

Previous systems, such as FMILL/APTLFT, though very useful, basically controlled the tool relative to only a set of points extracted from a single surface. In the SSX3 system a general APT cutter can be positioned to simultaneously contact two, or even three, sculptured surfaces.

1.2 Summary of Language

The system has all the standard language features of APT, and, in addition, provides special language for defining:

- synthetic curves or
- sculptured surfaces.

In the subsequent language definitions the following conventions are used: a brace {a, b} is used to indicate an item which may or may not appear in the input and a bracket $\begin{bmatrix}\underline{A}\\ B\\ C\end{bmatrix}$ indicates information for which an optional choice of items is presented. If one of the items is underlined, then this item is the assumed value if none of the options is input. Otherwise, the absence of an option will produce an error.

1.2.1 Synthetic Curve

A synthetic curve is defined by the following language:

$$C = SCURV \ / \begin{bmatrix}SPLINE\\ CURSEG\end{bmatrix}, \ P1, \ P2, \ \ldots, \ PN$$

Here CURSEG (a curve segment) indicates a conic arc is to be defined 'through' the points P1, P2, ..., PN, while SPLINE indicates a sequence of cubic arcs is to be passed through the consecutive points.

Further options are available -- namely, constraints can be attached to any point by entering them immediately after the point in question. For example:

P2 {,TANSPL,VT} {,NORMAL,VN} {,CRSSPL,VC,} $

{WEIGHT,W} {,LIMIT,L}

indicates that

- VT is a vector tangent to C at P2 in the spline direction
- VN is a vector normal to C at P2 in the spline direction
- VC is a vector which may be tangent to a surface through C at P2
- W is a scalar weight value to be used to move the point P2 to obtain a smoother curve
- L is a scalar limit which limits the maximum movement of P2.

A third option is available -- namely, assigning global weights and limits to all points.

$$C = SCURV \ / \begin{bmatrix}SPLINE\\ CURSEG\end{bmatrix} \ ,WEIGHT,.5 \ \ ,LIMIT,.3 \ \ ,P1 \ \ ,P2 \ \$$$

P3 ,WEIGHT,1.0 ,P4, ..., PN

In the above case all points carry a weight of .5 and a limit of .3 except that this global condition is overridden at the single point P3, which has a weight factor of 1.0. The curve will pass through P3, but the other points will move.

A great deal of language is planned for the future with synthetic curves -- in fact, as the name indicates, the primary future idea is to combine (synthesize) several synthetic arcs into a single curve. This feature has not been implemented in the present system as yet.

1.2.2 Surfaces

Sculptured Surfaces permit a variety of definitions -- basically they fall into two categories, namely, definitions which use points and vectors and definitions which use previously defined synthetic curves. The major definition forms are presented below:

MESH Definition

```
                                  ┌XYZ    ┐
                                  │XYPLAN │
S = SSURF / MESH*,                │YZPLAN │ ,    $
                                  └ZXPLAN ┘
         SPLINE, P1, P2, ..., PN,    $
         SPLINE, Q1, Q2, ..., QN,    $
         SPLINE, R1, R2, ..., RN
```

Here the P's, Q's, and R's are all points (see Figure 4). The stream of points P1, P2, ... through PN forms a spline curve as does the 'Q' stream and 'R' stream. In addition, the cross streams -- namely, P1, Q1, R1 for example -- also should form smooth splines. The word MESH indicates that the points form a coordinated, but flexible grid in space. The word 'XYZ' indicates that the grid is completely general, while the presence of another word (such as XYPLAN) indicates that the grid is somewhat specialized and the surface will occupy less storage and process more quickly.

As in the case of the synthetic curve a number of optional constraints may be attached to any point of a MESH, i.e.,

```
Q1 {,TANSPL,VT} {,CRSSPL,VC} {,NORMAL,VN} {,WEIGHT,W}
   {,LIMIT,L}
```

----

* The word MESH is a temporary word. It has been used as a MAJOR word in some APT systems. Other words such as 'NET' or 'GRID' may possibly replace it in the future.

Here

- VT is a vector tangent to the surface and spline at Q1

- VC is a vector tangent to the surface and cross spline at Q1

- VN is a vector normal to the surface (and splines) at Q1. If VT and VC are not perpendicular to the normal VN, then they are forced to be perpendicular

- W is a scalar weight which is attached to Q1 to be used in moving Q1 to obtain a smoother curve. Smoothing is conducted only on the splines in the major direction and at least three points not on a line must be present for smoothing to occur

- L is a scalar limit which controls the maximum movement of Q1 from its original position during smoothing.

The default value of all weights is 1.0 (certitude) and of all limits is infinity. Weights can vary reasonably anywhere from 0.0 to 1.0.

Patch Definition

A second type of surface definition is the individual PATCH input. This process permits a programmer to directly create the individual patches which define a surface. There are three basic patch inputs: PNTVEC, POLYGN and PNTSON.

The syntax is as follows:

$$S = SSURF \;/\; PATCH, \begin{bmatrix} PNTVEC \\ POLYGN \\ PNTSON \end{bmatrix}, 4, 2, \$$$

$$\begin{bmatrix} PLUS \\ MINUS \end{bmatrix}, PV1, PV2, \ldots, PV16, \$$$

$$\begin{bmatrix} PLUS \\ MINUS \end{bmatrix}, QV1, QV2, \ldots, QV16, \$$$

$$\begin{bmatrix} PLUS \\ MINUS \end{bmatrix}, RV1, RV2, \ldots, RV16, \$ \qquad \text{(See Figure 4a)}$$

A three patch surface is defined above. The word 'PATCH' means that patches are individually described. The selection of PNTVEC, POLYGN or PNTSON indicates the definition form for the individual patches. 4 is the number of 'U' boundaries in the total patch structure. 2 is the number of 'V' boundaries. PLUS or MINUS are primarily markers between

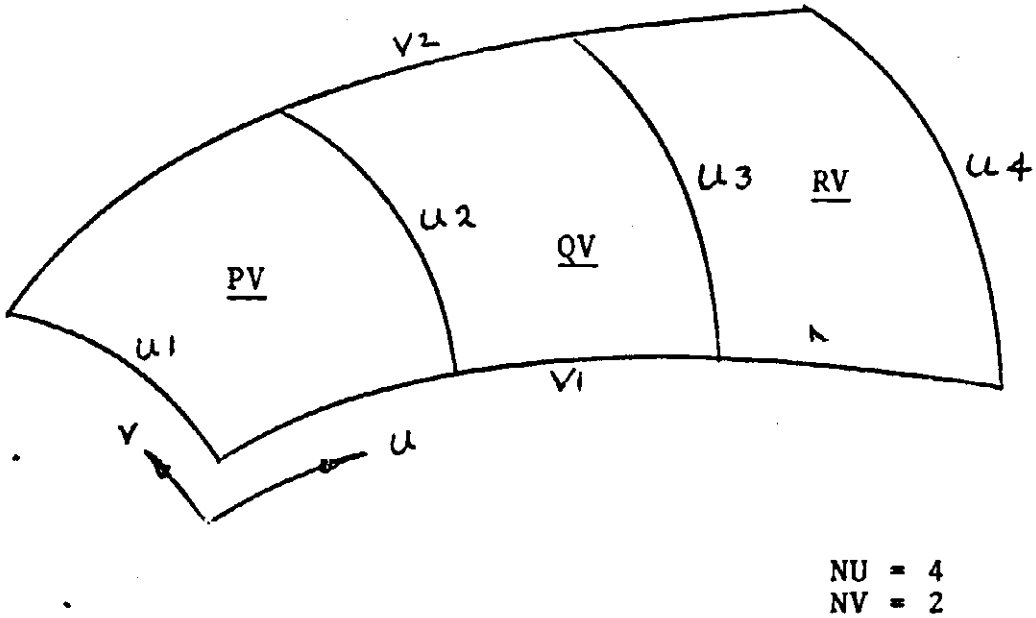


Figure 4a

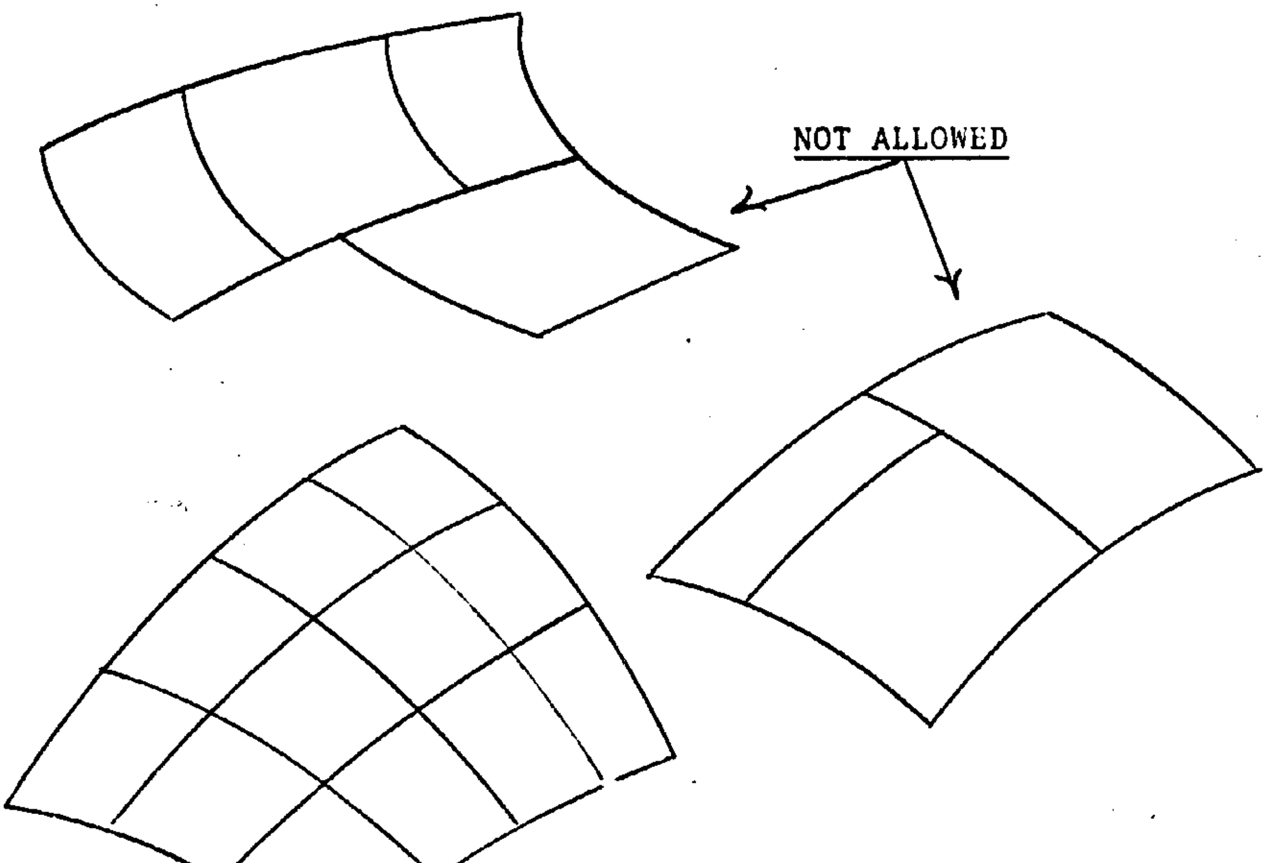


Figure 4b

patches. Each patch is then defined by a combination of sixteen points and vectors. Please note that it is possible to define patches which do not match at their boundaries and also surfaces which do not have smooth slopes where they meet. It is intended that the programmer will guarantee a proper match when defining surfaces in this way. Note: the addition of the number of U's and number of V's in the PATCH definition is the only sculptured surface syntax change required for SSIP processing. It should also be noted that the PATCH structure shall be rectangular (see Figure 4b).

The 'PNTVEC' option indicates that the patches are defined by Coon's point/vector canonical form as shown in Figure 4.

```
SP00, SP10, FD00, FD10, $

SP01, SP11, FD01, FD11, $

SD00, SD10, TW00, TW10, $

SD01, SD11, TW01, TW11
```

Here SP is mnemonic for surface point, FD for first direction vector, SD for second direction vector and TW for twist vector. Each patch is visualized as a unit square which has been stretched and formed. The corners are still referred to by their original values, i.e., SP00 is the surface point at the $u = 0$, $v = 0$ corner. Again, the programmer must input these vectors in the correct order or unexpected problems may occur in the desired surface representation.

The 'POLYGN' option signifies that the patches are defined by points which form a 'Bezier' polygonal umbrella (see Figure 5a). The points must be entered in the order shown in the figure. In this representation, the points (other than corner points) do not <u>lie</u> on the surface, but do define tangents and normals to the surface at the corners. Renault Motor Works in France (under P. Bezier) has used this form successfully for interactive graphics design of car bodies.

The 'PNTSON' option signifies that patches are defined so as to pass through sixteen points as shown in Figure 5b. If several interconnected patches are so defined, they are subsequently modified to achieve closer slope continuity across boundaries. The 'PNTSON' system was originally installed before the 'MESH' definition and still presents opportunities of special utility.

The third major type of sculptured definitions are those which define surfaces by using synthetic curves. The following definitions are available:

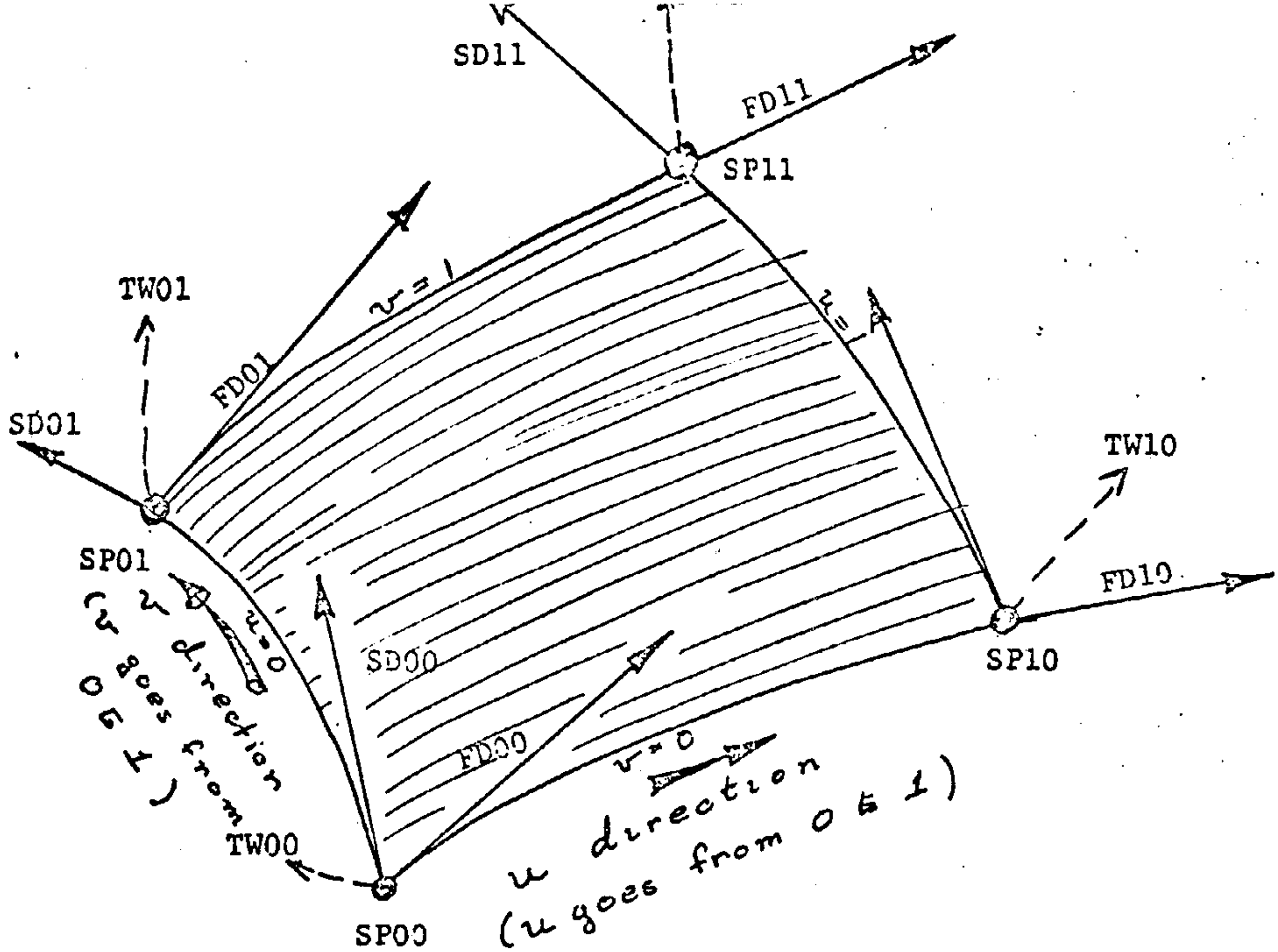


Figure 4. Example of 'PNTVEC' Canonical Definition of Patch

SP = Surface point; FD = first direction tangent

SD = Second direction tangent; TW = Twist vector

```
 S = SSURF/PATCH, PNTVEC, PLUS,$
     SP00, SP10, FD00, FD10, $
     SP01, SP11, FD01, FD11, $
     SD00, SD10, TW00, TW10, $
     SD01, SD11, TW01, TW11
```

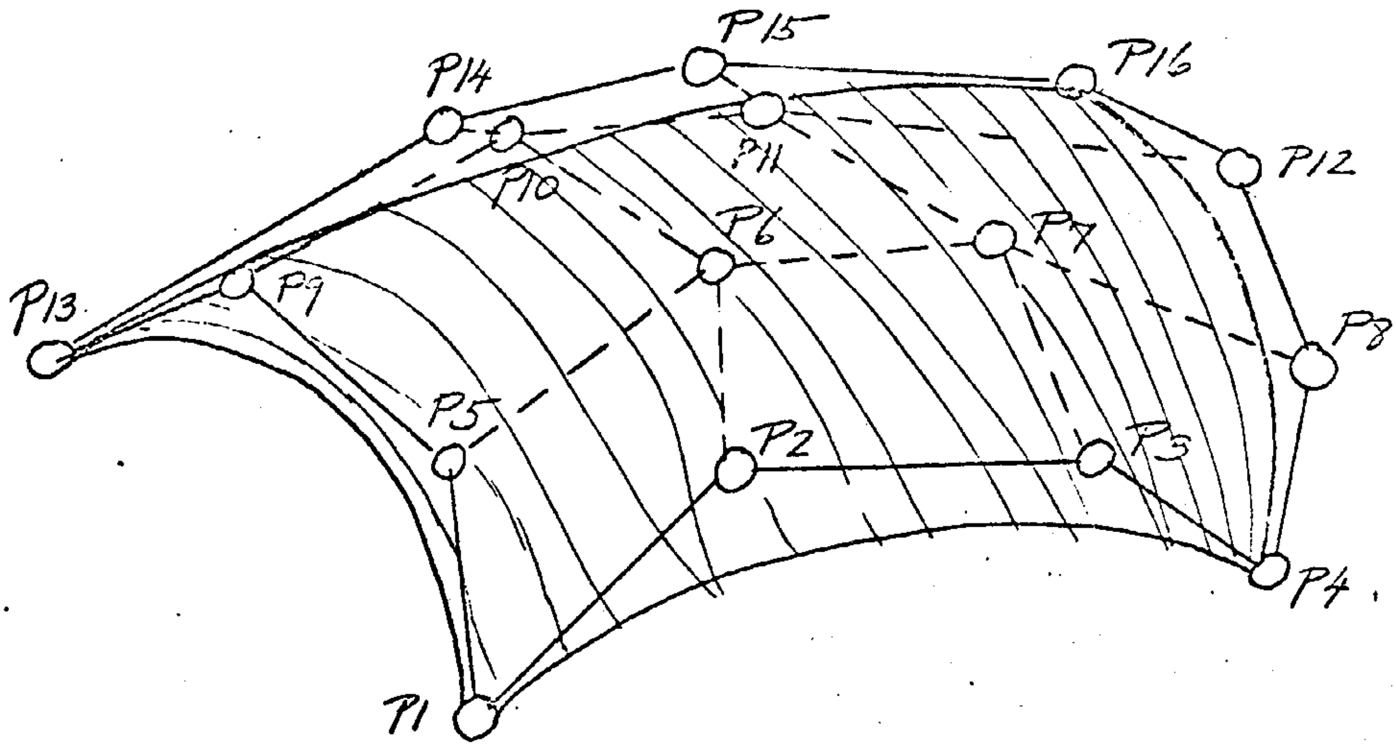


Fig. 5a. B = SSURF/PATCH, POLYGN, PLUS, P1, P2, ..., P16

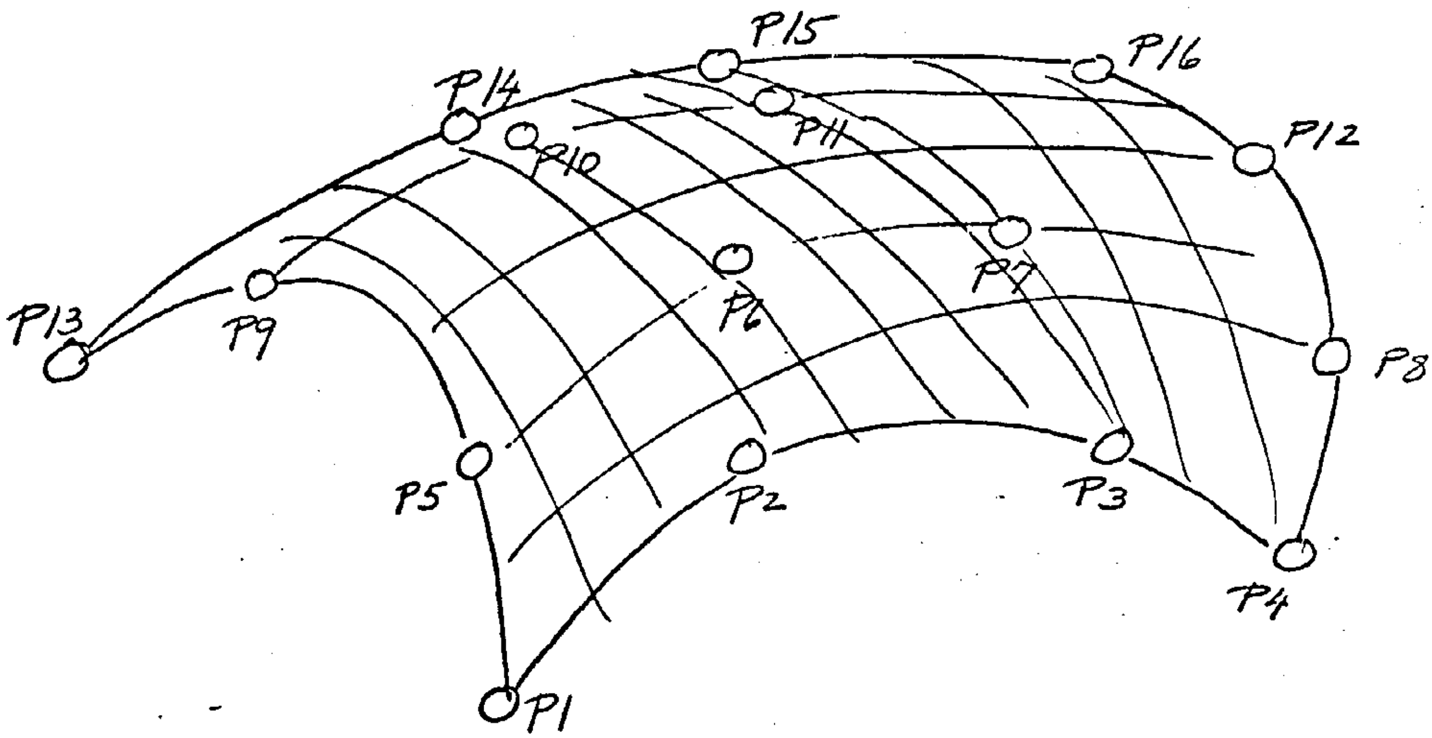


Fig. 5b. S = SSURF/PATCH, PNTSON, PLUS, P1, P2, ..., P16

Figure 5. Contrasting Examples of 'POLYGN' and 'PNTSON' Definitions

Surface of Revolution

$$S = SSURF\ /\ REVOLV,\ C,\ AXIS^*,\ P1,\ \begin{bmatrix} P2 \\ V \end{bmatrix},\ \begin{bmatrix} CLW \\ CCLW \end{bmatrix},\ A,\ B$$

This statement reads, S is a sculptured surface formed by revolving curve C around an AXIS defined by the two points P1 and P2 in a clo direction from an angle A to an angle B (in degrees). The AXIS may by a point and vector. It is noteworthy here that the curve C is

- either a conic or spline, and
- can touch the axis, or
- be twisted out of the plane of the axis, and
- form a closed loop (in which case the surface is a form of donut).

These capabilities, combined with the fact that the surface can be revolved through a finite range of angles and has linear extensions beyond all boundaries make this surface of revolution one of unparalleled flexibility.

Cylindrical Surface

$$S = SSURF\ /\ RULED,\ C,\ AXIS^{**},\ \begin{bmatrix} V \\ P1,\ P2 \end{bmatrix}$$

This statement reads that S is a sculptured surface which is RULED (that is, generated by moving a straight line through space) and that the rulings pass through the synthetic curve C, with a fixed axial direction parallel to the vector V. The axial direction of the rulings may also be defined by two points. since the axis is fixed, the resulting surface is cylindrical. It is planned that in the future, variable rulings will be allowed under this format. This will be possible since varying cross tangent vectors can be attached to a synthetic curve during its definition.

---

* In the SSX3 system the programmer should use the word 'XAXIS' instead of 'AXIS'. The word 'AXIS' has not yet been loaded into the system.

** In the SSX3 system, the word 'AXIS' has not yet been loaded into the Translator Table. For the interim period the word 'XAXIS' is being used instead.

1.2.3 Language Techniques for Easy Definition

At times there are complaints that since APT is a symbolic processor, too much data preparation is required. In the case of sculptured surfaces, since large numbers of points are required for definitions, this problem is more acute. It is well to remember, however, that the APT system has many work-arounds so that these problems are not as great as they seem.

For example, consider a 'MESH' surface defined by twenty-one points as follows:

```
Q1 = POINT/ 0.   , 1.238, 1.745
Q2 = POINT/  .1  , 2.1  , 1.3
Q3 = POINT/  .15, 2.9   , 1.2
          .
          .
          .
Q21 = POINT/10, 5.8, 3.7
S = SSURF / MESH, XYZ, SPLINE, Q1, Q2, Q3, Q4, Q5,$
                               Q6, Q7,  $
                       SPLINE, Q8, Q9, Q10, Q11, Q12, $
                               Q13, Q14,  $
                       SPLINE, Q15, Q16, Q17, Q18, Q19, $
                               Q20, Q21
```

If the above text were prepared on a keypunch with an automatic duplicating feature then the following formulation would reduce much of the encoding to duplication and number entry only. Furthermore, a synonym may be used for the word POINT to further reduce data. Thus the part program would appear as simple as:

```
SYN/POINT, P, SPLINE, SPL
RESERV/Q, 21
I = 0
Q(I = I+1) = P/0.   , 1.238, 1.745
Q(I = I+1) = P/ .1  , 2.1  , 1.3
Q(I = I+1) = P/ .15, 2.9   , 1.3
               .
               .
               .
Q(I = I+1) = P/10, 5.8, 3.7
S = SSURF / MESH, XYZ, SPL, Q(1, THRU, 7), $
                       SPL, Q(8, THRU, 14), $
                       SPL, Q(15, THRU, 21)
```

If the surface needs to be transformed in space all that is required is that a 'REFSYS/M' statement be placed around the points before the sculptured surface is defined or around the surface definition itself. The latter is not true for the SCURV/ and SSURF/REVOLV. Either form of 'REFSYS/M' may be used but not both.

A particularly useful feature of the sculptured processor is that surfaces which are represented by equations can be rapidly approximated. For example, consider the surface defined by an equation

$$Z = X^2 + Y^3.$$

This is not in the standard APT repertoire of surfaces. Suppose the area of interest is in first quadrant of the XY plane where X is between 1 and 5 and Y is between 1 and 6. Then the following loop in APT can be used to define a network of points on the surface and approximate the surface by a sculptured mesh. Please note that the greater the number of points, the more accurate the surface representation. Also the representation is only accurate within the bounds of the point mesh.

```
      RESERV/Q, 30
      J = 0
      X = 0
ID1)  X = X + 1
      Y = 0
      IF (X - 5) ID3, ID3, IDEND
ID3)  Y = Y + 1
      IF(Y - 6) ID4, ID4, ID1
ID4)  Z = X*X + Y*Y*Y
      J = J + 1
      Q(J) = POINT/X, Y, Z
      JUMPTO/ID3
IDEND)   S = SSURF/MESH, XYPLAN, SPLINE, Q(1, THRU 6), $
                                 SPLINE, Q(7, THRU, 12), $
                                 SPLINE, Q(13, THRU, 18), $
                                 SPLINE, Q(19, THRU, 24), $
                                 SPLINE, Q(25, THRU, 30)
```

In general, the utilization of the system is limited mainly by the programmer's knowledge and resourcefulness in applying the full capabilities of APT to the problem at hand.

1.3 Synthetic Curve Capability

The new type of geometry in the SSX3 system is the so-called synthetic curve or 'SCURV'. The word synthetic is descriptive of the ultimate objective for this curve -- namely, that it be possible to describe curves which are made up (or synthesized) from many different arcs. However, at present a synthetic curve can be either

- a twisted spline in space or else

- a planar conic curve in space.

Synthetic curves mathematically are interlinked cubic (parametric rational cubic) arcs. In the case of a conic -- a single arc is used for the description, while in the case of a spline, a cubic arc is inserted between consecutive input points along the spline. Beyond the boundaries, these curves are treated as linear extensions of the final tangent vector.

At present, there is no direct interface between synthetic curves and other APT definitions (i.e., lines, circles, ellipses, etc.). There is an indirect interface -- for example, if an APT circle is known, then points can be obtained from it by geometric construction and these points can then be used to define the synthetic form of the same circle.

The programmer should be aware of the following conceptual differences between a synthetic curve and an APT surface.

- A synthetic curve is a true space curve, i.e., it is not a cylinder. It should be conceived as a thin wire in space.
- A synthetic curve is bounded -- that is, it extends from one finite position in space to another. On the other hand, APT surfaces are infinite or extent.
- Ordinary concepts of tool control do not apply directly to synthetic curves.

1.3.1 Conic Arcs

A variety of conic arcs can be defined by the curve-segment (CURSEG) option of the synthetic curve language. All of the definitions depend on an orderly stream of points which approximately lie on the arc in question. In particular the programmer must provide the starting and ending points of the arc. A curve segment can only be a finite, contiguous arc. For example, it cannot represent two branches of a hyperbola at the same time. As mentioned earlier, the basic definition statement for a curve segment is of the following form:

```
C = SCURV / CURSEG, P1, P2, P3, P4, ..., PN
```

The points should proceed in an orderly stream from one end of the arc to the other. As mentioned earlier constraints may be entered at any point, including tangent or normal vector constraints and weights or limits.

In the case of CURSEG definition, the meaning of weights and limits is that the points should first be moved and then a conic should subsequently be fit through the new positions.

A conic arc is uniquely defined by five conditions (three points and two vectors). Any additional data amounts to overkill -- so that if a programmer inputs a total of more than six points and vector conditions he will receive a diagnostic and the definition will fail.

- The points must lie in a plane.
- All tangent vectors must lie in the same plane as the points.
- The sum of all points and tangent vectors must not exceed six.
- The points and tangents must not force an inflection in the resulting arc.
- No two points in the input should be identical.
- At present a conic arc must span less than 180 degrees of arc. (Hence a complete circle or even semi-circle cannot be defined as a single curve segment).
- In the case of an SCURV, the direction of flow of the curve is determined by the flow of the input points. Any tangent or normal constraints can point in either direction and will be automatically adjusted to point in the same direction as the point flow.
- Finally, the processor generally attempts to fit a conic closely through all of the data, but an exact fit is guaranteed only under special conditions. In all cases, the arc does match the end points and also end vector directions if any directions are provided.

The following describes the system action for various inputs.

Two Conditions

If two points are input, a straight line segment from the first to the second is generated.

Three Conditions

There are five possible ways in which three conditions can be input. They are:

C = SCURV/CURSEG, P1, P2, P3

C = SCURV/CURSEG, P1, TANSPL, V1, P3

C = SCURV/CURSEG, P1, NORMAL, VN1, P3

C = SCURV/CURSEG, P1, P3, TANSPL, V3

C = SCURV/CURSEG, P1, P3, NORMAL, VN3

In all of the above cases an exact circular arc is fit through the input data. The use of a normal vector is particularly useful when the center of the circle is known, since a normal vector is then the difference between the center and one end point of the arc.

Four Conditions

Four conditions are in general too many to define a circle, and too few to specify a unique conic. Consequently, the processor tries to determine a '<u>smooth</u>' conic which approximates the data. If three or four points are input, the conic does not pass exactly through the interior points. However, any end slopes are matched. Finally, if the four points should lie on an exact circle, then a circular segment will be defined.

Five Conditions or More

The one input format which <u>guarantees</u> a unique general conic is the format where three points and slope vectors (tangent or normal) are provided at each end. In this case a unique conic matching all of the input conditions is generated.

If five or even six points are input, only one of the interior points will be matched. The others will be used only to determine approximate slopes at the ends and the resulting conic will be an approximation to some of the input. For example, up to six points may be chosen on an ellipse, but the resulting 'CURSEG' arc will be a conic which only approximately fits the interior positions.

1.3.2 Splines

A synthetic spline curve is defined by the format

S = SCURV/SPLINE, P1, P2, ..., PN

As mentioned earlier, various global and local constraints can be applied at the input points. The resulting spline which is generated consists of a set of interlinked cubic arcs, each spanning the space between consecutive input points.

The presence of a TANSPL or NORMAL vector constraint at any interior point of the spline actually causes the spline to be split there for purposes of slope determination. This feature allows a spline to negotiate a sharp corner without universal wrinkles by the addition of only one or two constraints. However, if weights are used, the spline is first treated as a whole for purposes of smoothing and split later for final slope determination.

The input points do not have to lie in a plane. For example, they could lie on the outer periphery of a winding helicoid. In general, two consecutive points should not be equal. However, the first and last point of the spline can agree in which case a smooth, continuous closed curve is automatically generated.

The points in a spline do not have to be spaced at equal intervals. However, programmer experience is definitely an asset regarding practical choices in the number and distribution of points required for a good fit.

1.3.3 Verification Listing

The programmer receives a verification listing each time a synthetic curve is defined. This consists of three components:

- the blocked input data before the input is subjected to splining and conic fitting algorithms;
- a display and analysis of the curve segment by segment, showing points, tangent, vectors, normals and radii of curvature at five positions on each segment. This part of the display is most useful to part programmers;
- the blocked output structure which is the final canonical form of the curve. This part of the output is of little use to the part programmer and is included temporarily as an aid in systems validation, since the curve is a brand new feature in the system.

A sample curve definition and listing are shown in Figure 6. Note in the listing that all spline points have been translated to a local coordinte system with the first spline point the origin. This is for purposes of compression (see section 1.1.9).

1.3.4 Size Considerations

A synthetic curve can describe curves of any practicable physical size, either small or large. However, there are

```
1  PARTNO SCULPTURED SURFACE TEST (SCURV)
2  SSPRNT=SSURF/DISPLY,ON
3  P1    =POINT/0,15,0
4  P2    =POINT/5,14.524,C
5  P4    =POINT/15,9.922,C
6  P5  =POINT/2C,0,0
7  C7    =SCURV/CURSEG,P1,P2,P4,P5


SCULPTURED( OR SYNTHETIC) CURVE     C7


CANONICAL ARRAY FOLLOWS
BLKNUM=   1     0.0      1.0C0    3.000   4.000    0.C     0.C    0.0    0.0     0.0    0.0    0.0    0.0
                0.0      C.0      0.0     0.0      0.0     C.0    0.0    0.0     0.0    0.0    0.0    0.0
BLKNUM=   2     0.0      0.0      0.0     0.0      0.0     0.0    0.C    0.C     0.0    0.0    0.0    0.0
                0.0      0.0      0.0     0.0      1.000   0.C    0.0    0.0     0.0    0.0    0.0    0.0
BLKNUM=   3     5.000   -0.476    0.0     C.0      0.0     0.0    0.0    0.0     0.0    0.0    0.0    0.0
                0.0      0.0      0.0     0.0      1.000   0.0    0.0    0.C     0.0    0.0    0.0    0.0
BLKNUM=   4    15.000   -5.078    0.0     0.0      0.C     0.0    0.C    0.C     0.0    0.0    0.0    0.0
                0.0      0.0      0.0     0.0      1.0C0   0.0    0.0    0.C     0.0    0.0    0.0    0.0
BLKNUM=   5    20.000  -15.000    0.0     0.0      0.0     0.0    0.0    0.C     0.0    0.0    0.0    0.C
                0.0      0.C      0.0     0.0      1.000   0.C    0.0    0.C     0.0    0.0    0.0    0.0

SCULPTURED( OR SYNTHETIC) CURVE     C7

   PARAM    XCOORD    YCOORD    ZCOORD   UTAN-I   UTAN-J   UTAN-K   UNORMI   UNORMJ   UNORMK   CURVATURE RADIUS
SEGMENT NUMBER   1
   0.C      0.0       0.0       0.0      1.0000   0.0022   0.0      0.0      0.0     -1.C000   0.0367   27.278
   0.2500   6.7743   -0.9050    0.0      0.9616  -0.2744   0.C      0.0      C.0     -1.CCC0   C.0455   21.989
   0.5C00  12.9321   -3.8423    0.0      0.8121  -0.5835   0.0      0.0      0.0     -1.C000   0.0548   18.255
   0.7500  17.5511   -8.7323    0.C      0.5328  -0.8463   0.C      0.0      0.0     -1.C000   0.0575   17.385
   1.CC00  20.C000  -15.0C00    0.0      0.1901  -0.9818   0.C      0.0      0.0     -1.C000   0.0509   19.646

CANONICAL ARRAY FOLLOWS
BLKNUM=   1     0.0      1.CC0    3.000   1.000    0.0     0.0    0.0    0.0     0.0    0.0    0.0    0.0
                0.0      0.C      0.0     0.0      0.0     0.C    0.0    0.C     0.0    0.0    0.0    0.0
BLKNUM=   2     0.0      4.797    1.634   6.054    0.0    -3.598  0.004 -10.C00   0.0    0.0    0.0    0.0
                0.060    0.240    0.183   0.241    0.0     C.C    0.0    0.C     0.0    0.0    0.0    0.0
```

FIGURE 6. Verification Listing - Spline

limits on the amount of data which can be used in a single curve definition.

- The system cannot handle more than 54 points in one definition.

- The numerical size of the synthetic curve is usually not a problem. However, more important is the amount of data generated when using it in a large sculptured surface definition. The size limitations are as follows:

  CURSEG case: there are no overflow problems; the largest SSURF that can be defined would be one with only 3 patches.

  SPLINE case: maximum of 54 points when combined in a RULED surface definition. The maximum for a REVOLV surface is 39 points for .GT. 120°; 23 points for .LT. 120° and .GT. 240°; 16 points for .LT. 240°.

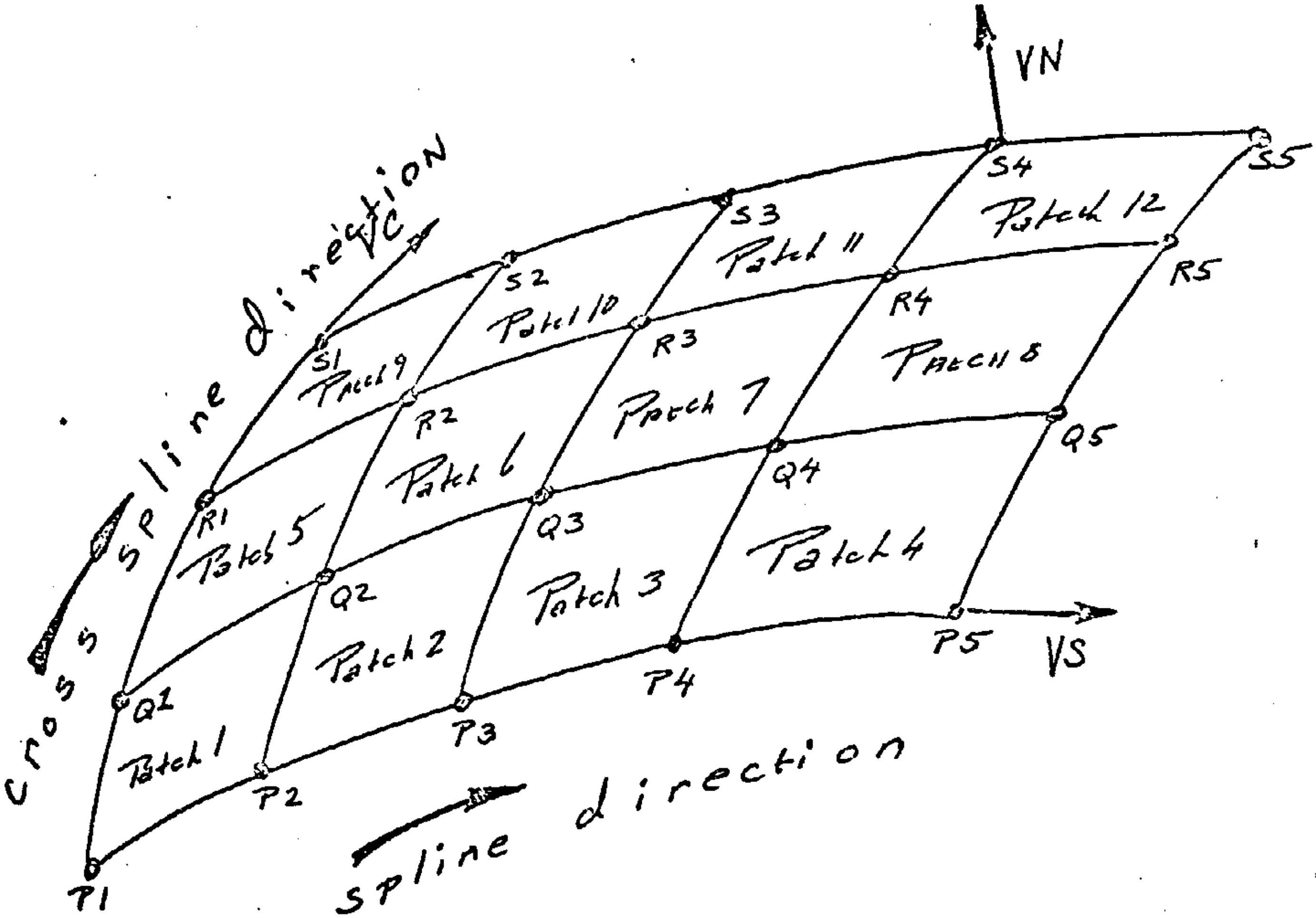
1.4 Surfaces

There are four major categories of definition for sculptured surfaces, namely, MESH, PATCH, REVOLV and RULED. All of these definition methods have a great deal in common -- they have all been developed to give the programmer useful ways in which to define a surface, and the resulting surface is always a patch structure and is displayed and analyzed as such.

1.4.1 MESH-of-Points

The general MESH-of-points surface (see Figure 7) is still the most important capability in the SSX3 system. A set of points defining roughly parallel curves across the surface are input to the MESH processor. These determine the major spline direction. Each spline must contain the same number of points. There is an added difficulty in the MESH definition that these points must also be coordinated in the transverse or cross-spline direction so that smooth splines can be generated in that direction as well. In the example shown, four splines are fit in the major spline direction and five splines are fit in the cross direction. The resulting slopes at the junction points are used to interpolate surface patches among neighboring points.

The resulting number of patches is twelve, which can be computed as the product of the number of points per spline less one times the number of parts per cross-spline less one (i.e., 12 = (5-1) x (4-1)). The patches are numbered sequentially first in the direction of the major spline and next in the direction of the consecutive cross splines.




```
S = SSURF/MESH, XYZ, $
    SPLINE, P1, P2, P3, P4, P5, TANSPL, VS, $
    SPLINE, Q1, O1, Q3, Q4, Q5, $
    SPLINE, R1, R2, R3, R4, R5, $
    SPLINE, S1, CRSSPL, VC, S2, S3, S4, $
         NORMAL, VN, S5
```

Figure 7. Mesh-of-Points Illustration

Since patch verification listings provide a display of patches in their sequential order, it is useful to a programmer to know the logic used in determining this sequential layout.

The concepts of vector constraints and smoothing have been treated in previous sections. In the mesh definition there is presently no way in which to effect a global setting of weights and/or limits. In other words, weight or limit constraints must be entered at every point where the programmer wishes to change them from the default values. Also, the programmer should be aware that the smoothing concept for a mesh applies only to individual curves in the major spline direction. These curves are each smoothed individually without regard to the position of other curves in the mesh. The basic problem here is that unwanted fluctuations may then occur in the cross direction because the smoothing does not presently consider the cross fitting problem when adjusting the major splines.

All of the mesh definitions have a striking capability to describe a ruled surface which is generated by a straight line sliding along one or two curves in space. The original 'RLDSRF' in the APT system is an example of this. However, a different group of ruled surfaces can be generated using the sculptured surface 'MESH' definition. The idea is that any time the programmer inputs only two spline curves in a mesh definition, his implication is that he wishes to define a ruled surface where the rulings span the space between the two splines. There is a high degree of control of the rulings by the programmer in that each two corresponding points of the two splines are connected by a ruling. A perfect ruled surface is not created if either a CRSSPL or a NORMAL constraint is present. For an example of a ruled MESH surface, the helicoidal surface in Figure 8 was defined by a mesh of points.

A specialized version of the mesh surface can be invoked whenever the defining grid points form a parallel grid in one of the coordinate planes. If the parallel condition is satisfied, the part programmer requests the special mode with one of the words 'XYPLAN', 'YZPLAN' or 'ZXPLAN', whichever is appropriate. Figure 9a illustrates a case where the 'XYPLAN' can be used successfully and Figure 9b, a case where it would be invalid. Relevant to SSX3A/SSIP, this specialized version of the mesh surface offers no advantage in data size or APELEM efficiency. Also it shoudl be noted that 'XYPLAN' option produces a slightly different surface that the more general 'XYZ' mode.

A number of constraints are imposed on the input points -- namely, that two consecutive input points on a spline or cross spline may not be identical, or else, for any three

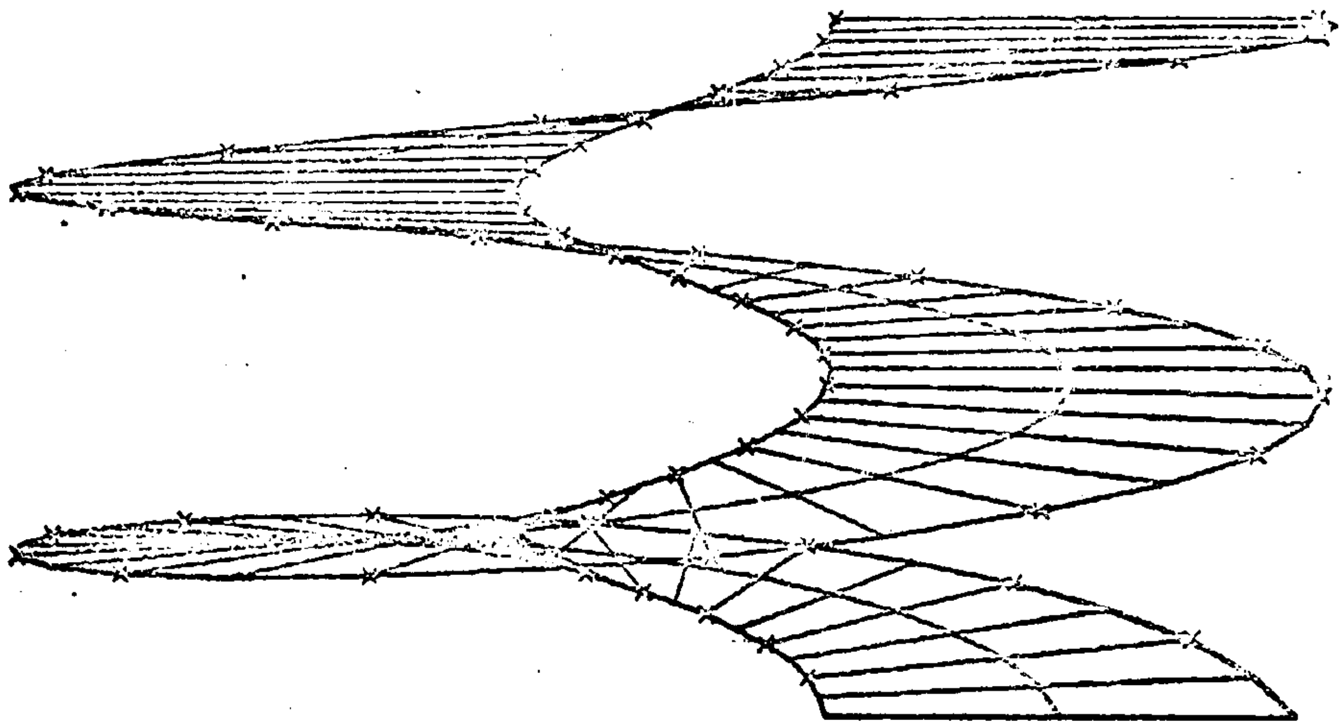


Figure 8. A Helicoidal Surface Defined by a Sculptured Ruled MESH of Points.

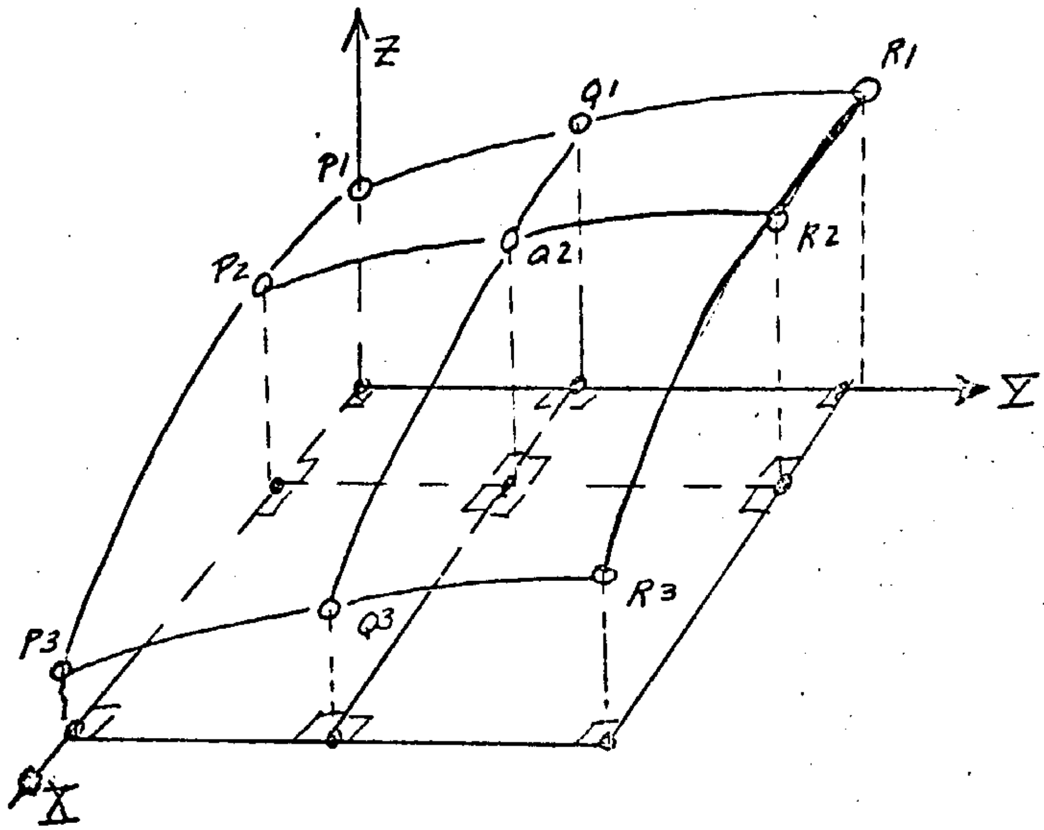


Figure 9a. A Valid XYPLAN Mesh of Points

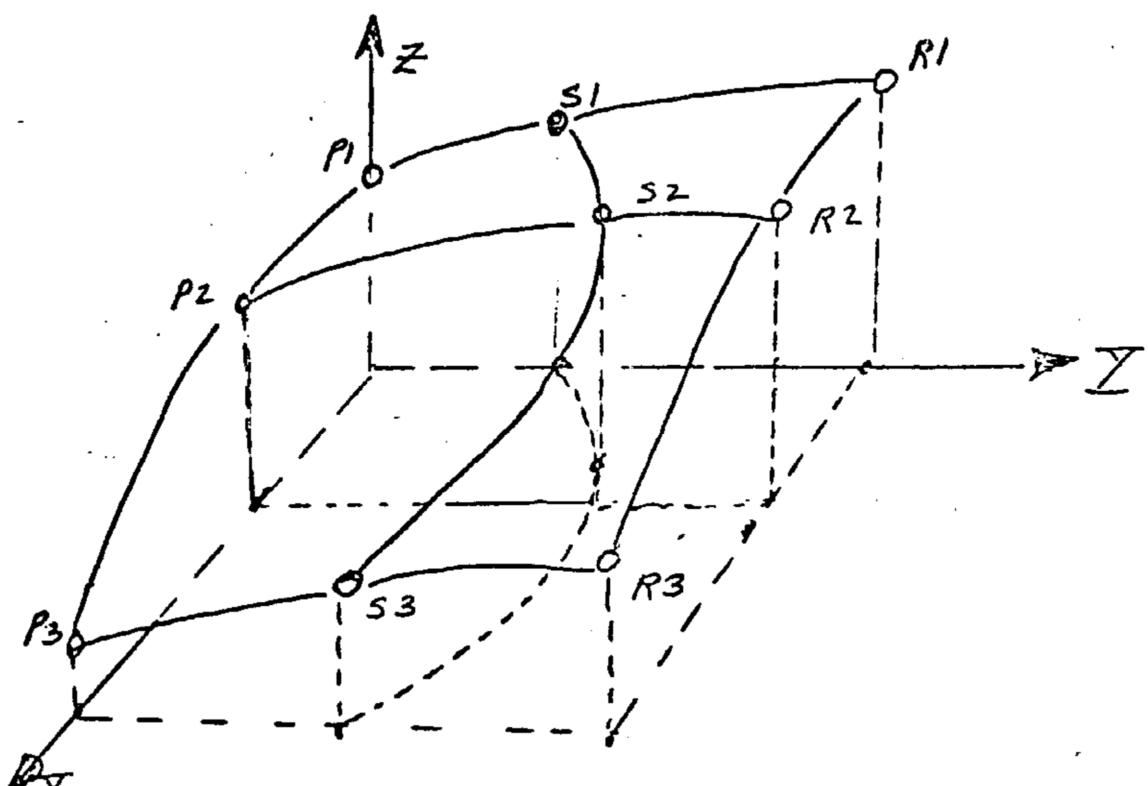


Figure 9b. An Invalid XYPLAN Mesh of Points

consecutive points of a spline, the middle point must project somewhere in the interior of the line segment between the outermost pair.

The programmer must be aware of certain size limitations built into the SSX3/SSIP system -- otherwise, he may easily define surfaces which are too large for the system to handle. The maximum mesh sizes are tabulated below.

SURFACE SIZE NPS x NPC OR NPC x NPS	NUMBER OF POINTS	NUMBER OF PATCHES (NPS-1) x (NPC-1)	NUMBER OF WORDS (4 x NP + 18)
2 x 54	108	53	450
3 x 31	93	60	390
4 x 22	88	63	370
5 x 17	85	64	358
6 x 14	84	65	354
7 x 12	84	66	354
8 x 10	80	63	338
9 x 9	81	64	342

NP = NUMBER OF POINTS
NPS = NUMBER OF POINTS PER SPLINE
NPC = NUMBER OF POINTS PER CROSS SPLINE

A large number of constraints applied within a mesh definition will increase the word count and could cause an overflow within 1000 word table (DEFTAB). Therefore, the following algorithm should be applied for predicting the maximum word count.

(NUMBER OF WORDS) + (NUMBER OF VECTOR CONSTRAINTS x 5) +

(NUMBER OF WEIGHT CONSTRAINTS x 2) = TOTAL NUMBER OF WORDS

As an example, a general XYZ MESH surface defined by 7 splines with 12 points each and 72 vector constraints and 50 weight constraints, would yield a numerical surface size of:

354 + (72 x 5) + (50 x 2) = 814 words

Consequently the above mesh definition fits well within the confines of the 1000 word limits. There are no restrictions on the number of sculptured surfaces defined in a single part program.

1.4.2 PATCH Definitions

The 'PATCH' definitions were the first implemented in the sculptured processor. At the time of formulation, they were desigred to prove the feasibility of using the patch technology in APT. However, these definitions still serve three major funcions -- namely,

- they provide a learning tool to aid programmers in building an understanding of the patch concept

- they provide a means of interfacing between special design preprocessors and the APT manufacturing system

- in many cases, they provide the means of describing complex shapes with very few patches. However, the programmer must possess greater understanding of the underlying technology than with the MESH approach.

There are three major types of PATCH definitions -- namely, the PNTVEC (mnemonic for point-vector) format, the POLYGN (mnemonic for the Bezier Polygon approach) and PNTSON (mnemonic for the points-on-processor).

Patch Definition - PNTVEC

In the 'PNTVEC' version of the patch definition (Figure 4) four corner points, major first direction tangents, cross or second direction tangents and twist vectors are introduced at each corner. Both the length and magnitude of these vectors are significant to the surface shape. These points and vectors actually represent the canonical form of Coon's patch -- that is, in every sculptured surface the final representation of surface patches is by means of Coon's canonical form points and vectors. Accordingly, the PNTVEC method can, at times, be used as a direct override of sculptured surface mathematical smoothing techniques. In general, the corner points and tangent vectors control the shape of boundary curves of a patch. For good general results the length of these tangents approximate the length of the curves. Finally, the twist vectors modify slopes across the boundaries of patches and accordingly modify the interior shape of a patch. A mathematical definition of the surface represented by Coon's canonical points and vectors was provided as an appendix to the release documentation of the SSX2 system.

Patch Definition - POLYGN

The essential idea of the 'POLYGN' input is to define boundary curves not by points on the curves directly, but by 'characteristic' polygons -- which define the important shape characteristics of the curves and patches. This definition of patches adapts itself particularly well to on-line interactive design using only 'small' computers. The definition has been particularly included in this processor so that surfaces which are defined elsewhere in such modes of operation, can then be input directly to APT by the sculptured surface processor to perform the manufacturing function.

Figure 5a shows a typical characteristic polygon of points and the resulting surface. Notice that only four corner points actually lie on the patch. The remaining points determine a polygonal 'umbrella' over the surface which controls tangent directions at patch corners. The 'POLYGN' input always demands sixteen points per patch. However, if the eight central points are identical, a 'ruled' patch will be generated across two parallel boundaries, and if the four 'central' points are identical, then a patch with zero twist vectors will be generated (see Figure 10a and Figure 10b).

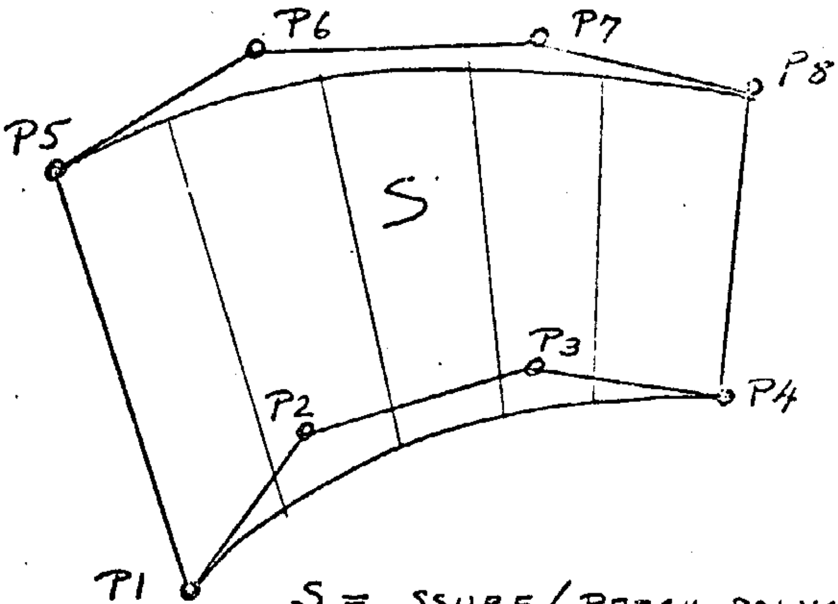
Patch Definition - PNTSON

A third method of defining individual patches is by the 'PNTSON' method. An example of this type of definition is shown in Figure 5b. The 'PNTSON' method accepts a small mesh of sixteen points to define each patch. There is a difference, though, between this definition and the other patch definitions -- namely, that after processing the definition of each individual patch, the system performs global processing to insure that patches blend smoothly together. During this process of blending, there is no longer a guarantee that the final surface actually passes through all of the interior points -- only that the surface is blended and makes an approximation to the input. There are many default conditions available in the PNTSON definition for a single patch. If two of four points which define a boundary are identical, then the boundary will be a straight line. If three instead of four distinct points are defined for a boundary, a parabolic-type curve on the boundary will be generated. If the eight central points are identical, a ruled patch will result and if the four central points are identical, a zero twist patch will result. In a sense then, the 'PNTSON' processor can be used as a learning tool for Coon's canonical form. There are also times when a single-patch is all the sculptured surface required for a given job. It should be noted that the input for a 'PNTSON' patch is far more randomized than a mesh. The illustration tries to convey this concept by placing the sixteen points in a non-uniform spatial relationship.

The maximum number of patches allowed within a 'PATCH' surface are 16.

1.4.3 Surface of Revolution

The surface of revolution is a new surface in the sculptured system. It is defined by revolving a synthetic curve around an axis in space. This processor actually




```
S = SSURF/PATCH, POLYGON, PLUS, $
        P1,P2,P3,P4, PZ,PZ,PZ,PZ $
        PZ,PZ,PZ,PZ, P5,P6,P7,P8
```

Figure 10a. Ruled Option for POLYGON Input

```
S = SSURF/PATCH, POLYGON, PLUS, $
    P1, P2, P3, P4, Q1, PZ, PZ, Q4, $
    R1, PZ, PZ, R4, S1, S2, S3, S4
```

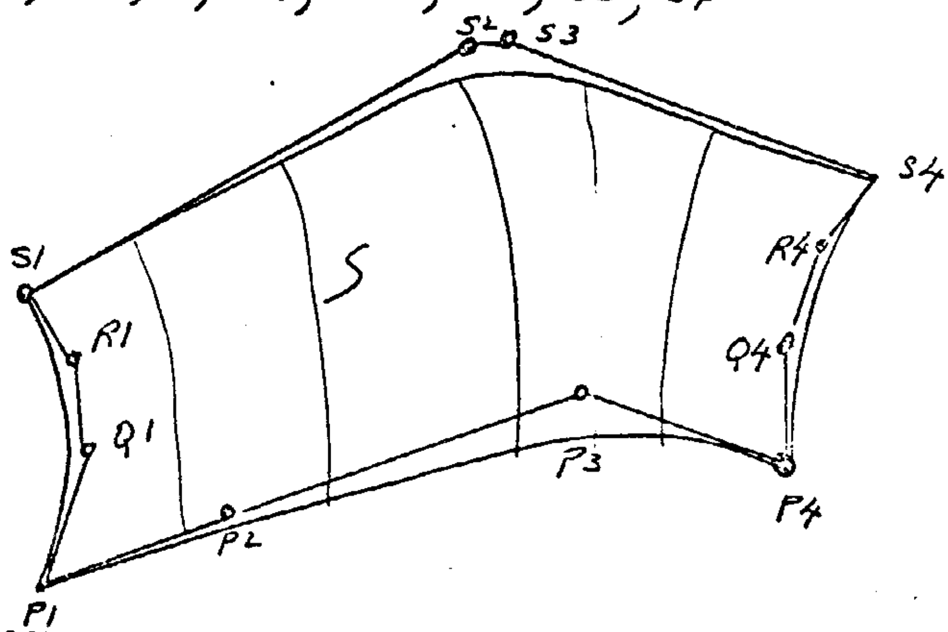


Figure 10b. Zero Twist Vector Option for POLYGON Input

generates a mesh-like sculptured surface. Each segment of the synthetic curve sweeps out from one to three patches as it revolves around the axis. A single patch can span up to one hundred and twenty degrees of revolution. As mentioned in the language section, this surface of revolution has few practical constraints -- that is, the curve need not lie in the plane of the axis, the curve can touch or cross the axis, the curve can be multi-valued with respect to the axis (for example, a closed circle) and the angular extent of the revolution can be restrained within finite bounds. A limited angle of revolution is very useful for die sinking a cavity -- by generating only the lower half of the surface the part programmer does not have to contend with interference caused by the presence of an undesired upper half of the surface.

The language for defining a surface of revolution (as described earlier) is

$$S = SSURF \ / \ REVOLV, \ C, \ AXIS, \ \begin{bmatrix} P1, \ V \\ P1, \ P2 \end{bmatrix} \begin{bmatrix} CLW \\ CCLW \end{bmatrix}, \ A, \ B$$

Where C is a previously defined synthetic curve. The axis is a directed space line passing from the point P1 to P2. In order to determine the rotational sense for the surface the programmer should imagine placing his eye at P2 and looking toward P1 or the vector directed towards the programmer. The first point of the curve C which does not lie on the axis determines a zero-angle for referencing other angular positions. Angles are measured as clockwise or counterclockwise from the programmer's sighting point at P2. In case a full surface of revolution is to be generated, the angles 0 and 360 must be entered, however, a full 360 degree surface is not recommended within SSIP. Difficulties may arise during arelem processing.

While a curve can cross the axis of revolution and still define a surface, a programmer should treat such a surface with caution since the surface reduces to a point at the axial intersection and surface normals perform a dangerous reversal around this point.

It is perfectly valid that the profile curve be a straight line. In general, if a straight line is revolved around an axis, a section of a cone is generated. Since extensions are linear, the extended surface is a complete cone. However, an attempt to move from one branch of the cone to another will meet with failure, since the apex of the cone is a singularity of the surface. If the profile line does <u>not</u> lie in the same plane as the axis, then a section of a hyperbola of revolution is generated. What this shows is that the concept of rotating a twisted profile curve around an axis can be highly relevant to real problems. Finally, it should be noted that the profile curve can be multivalued with respect to the axis. For example, the closed curve in Figure 11 may be rotated around the axis to form a donut -- or more practically, the surface of an auto tire.

For size limitations see <u>Size Considerations</u> (Section 1.3.4).

1.4.4 <u>Cylindrical Surfaces</u>

The format for generating a cylindrical surface from a synthetic curve C is

S = SCURV / RULED, C, AXIS, V

This feature is not extremely significant at the present time. However, it will be very significant in the future -- for, when a synthetic curve becomes a composite of conics and splines, this cylinder will represent a general 2-D contour. Secondly, in the future synthetic curves will be used extensively for defining general surfaces. Then, it will be very useful to construct cylinders or ruled surfaces through these same boundary curves to use them as bounding surfaces for cutter control.

For size limitations see <u>Size Considerations</u> (Section 1.3.4).

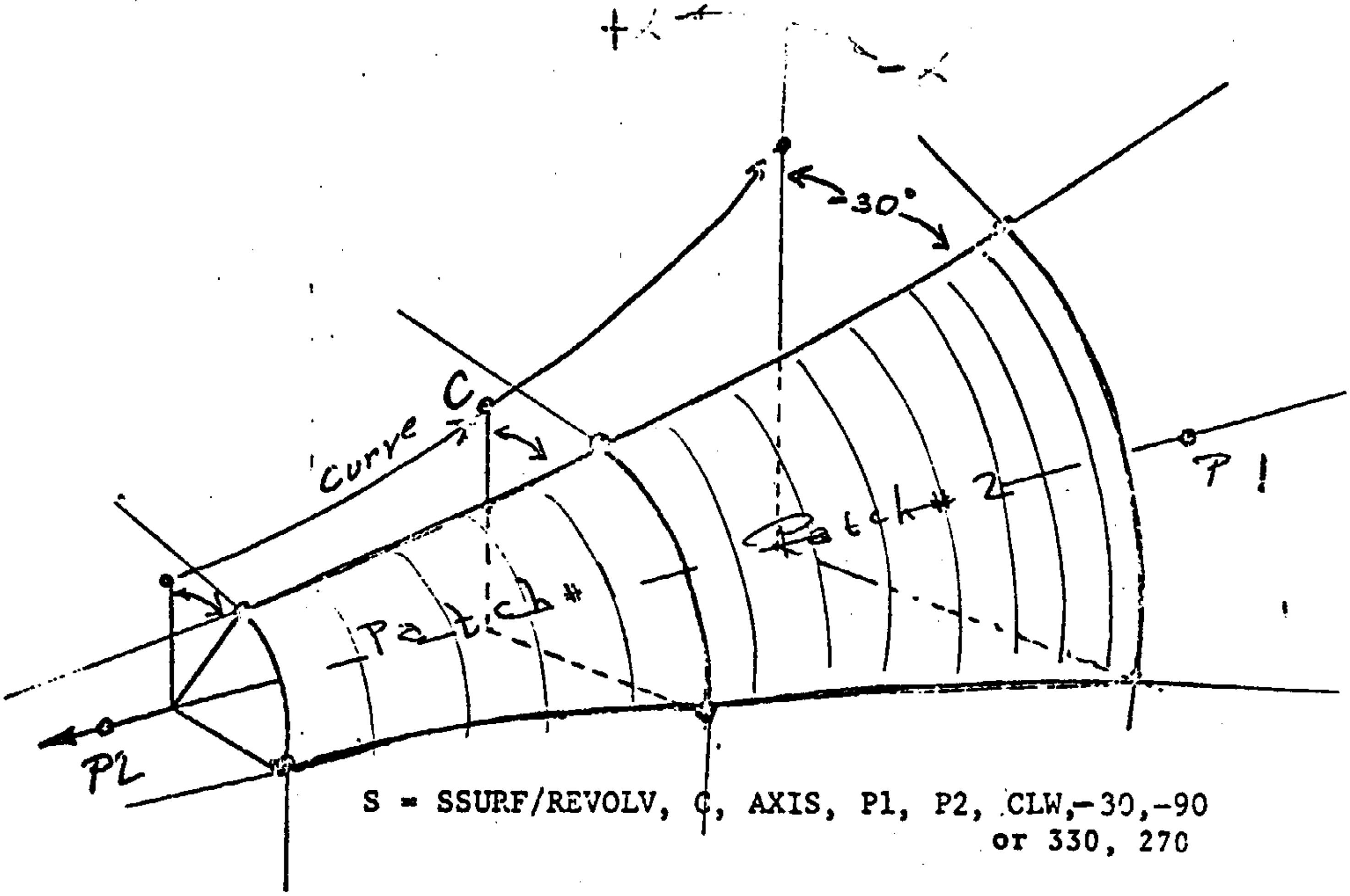
1.4.5 <u>Verification Listing for Surfaces</u>

All sculptured surfaces, whatever the definition -- generate a patch structure. Subsequent to definition, the programmer receives an analysis of the surface on his print output. This analysis proceeds by first displaying corner points, slopes, surface normals and curvatures for each patch. When used intelligently, this listing can tell the programmer a great deal about his surface and can also be used to detect errors in the input. For example, if one point in a mesh of points is grossly out of position (say from an encoding error) the surface area of the affected patches and possibly the entire surface will be abnormal. The verification listing for a sample part program is explained below.

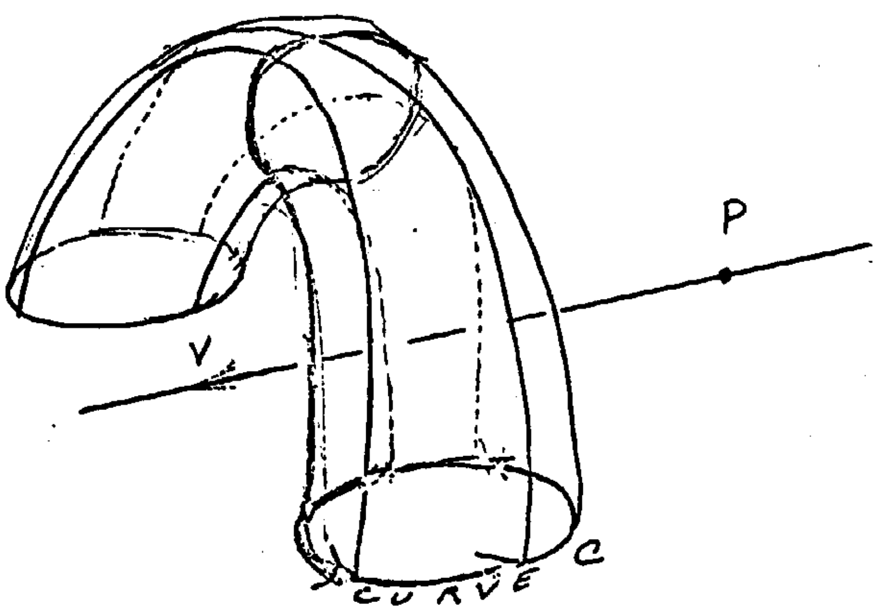
1.4.5.1 <u>General Description of a Sample Part Program</u>

A complete verification listing of a part program execution using sculptured surfaces is shown on five pages in Figure 12. The surface defined is a simple two-spline sculptured ruled surface.

Figure 12a lists the input part program, and Figures 12b and 12c show the surface analysis and display data for the generated sculptured surface. This display will be explained



S = SSURF/REVOLV, C, AXIS, P1, P2, CLW,-30,-90
or 330, 270



S = SSURF/REVOLV, C, AXIS, P, V, CCLW, 0, 180

Figure 11. Examples of Surfaces of Revolution

```
*** SCULPTURED SURFACE SYSTEM - SSX3 ( BASED ON A4X3 )   IIT RESEARCH

ISN
 1.   PARTNO DEMONSTRATION OF A SIMPLE RULED MESH SURFACE --SSX3 1973
 2.          P1 = POINT/ 0, 0, 0
 3.          P2 = POINT/ 1, 0, 1
 4.          P3 = POINT/ 2, 0, 1
 5.          P4 = POINT/ 3, 0, 0
 6.          S1 = POINT/ 0, 3, 0.1
 7.          S2 = POINT/ 1, 3, 0.6
 8.          S3 = POINT/ 2, 3, 0.4
 9.          S4 = POINT/ 3, 3, 0
10.          SM = SSURF/ MESH, XYPLAN, SPLINE, P1, P2, P3, P4,  $
10.                                   SPLINE, S1, S2, S3, S4
11.          FINI

APT IV - A4X3    TRANSLAT PHASE - RUN TIME=  0.39 MIN.    ENTRY POINT = 0019F3E=

       IER            0        0
```

FIGURE 12A. Test Part Program to demonstrate verification listing

SCULPTURED SURFACE

```
SP=SURFACE POINT,FD=FIRST(SPLINE) DIRECTION, SD=SECOND(CROSS SPLINE) DIRECTION, SN= SURFACE NORMAL
EACH PATCH HAS FOUR CORNERS  SP00,SP10,SP01,SP11.  BOUNDARY B1=SP00 TO SP10, B2=SP00 TO SP01, B3=SP10 TO SP11, B4 =SP01 TO SP11
SN50,SN05, ETC REPRESENT SURFACE NORMALS AT INTERIOR PTS.  SN05 REPRESENTS A NORMAL AT U=0.0,V=0.5 MIDWAY BETWEEN SP00 - SP01
POSITIVE RADII AND CURVATURES ARE CUPPED IN THE DIRECTION OF  THE SURFACE NORMALS SNI,SNJ,SNK

SS HEADER TABLE
VALUE TABLE SIZE                    75.
TOPOLOGY TABLE SIZE                 12.
NUMBER OF PATCHES                    3.
TYPE OF SURFACE(NONMESH OR MESH)     2.
PNTVEC, POLYGN, OR PNTSON            1.
SIGN OF SURF NORMAL +=FD00XSD00      1.
TOTAL SIZE OF STRUCTURE            115.
NUMBER OF SPLINES                    2.
NUMBER OF POINTS PER SPLINE          4.

PATCH      X         Y         Z         W              X         Y         Z         W            SNI       SNJ       SNK
PATCH NUMBER    1
SP00     0.0       0.0       0.0       1.0000   SP01   0.0       3.0000    0.1000    1.0000   SN00 -0.67532 -0.02456  0.73712
SP10     1.0000    0.0       1.0000    1.0000   SP11   1.0000    3.0000    0.6000    1.0000   SN50 -0.76368  0.03133  0.64483
FD00     1.0000    0.0       0.9162    0.0      FD01   1.0000    0.0       0.6420    0.0      SN10 -0.32498  0.12391  0.93757
FD10     1.0000    0.0       0.3466    0.0      FD11   1.0000    0.0       0.0922    0.0      SN01 -0.56320 -0.02751  0.82586
SD00     0.0       3.0000    0.0499    0.0      SD01   0.0       3.0000    0.0499    0.0      SN51 -0.48581  0.04241  0.87304
SD10     0.0       3.0000   -0.3965    0.0      SD11   0.0       3.0000   -0.3965    0.0      SN11 -0.09103  0.13048  0.98726
TW00     0.0       0.0      -0.2342    0.0      TW01   0.0       0.0      -0.2342    0.0      SN05 -0.62403 -0.02504  0.78096
TW10     0.0       0.0      -0.2544    0.0      TW11   0.0       0.0      -0.2544    0.0      SN15 -0.21250  0.12970  0.96851
FLAGS=2,2,4,1, B1=   0 B2=    0 B3=    2 B4=    0 SP55   0.5000    1.5000    0.4975    1.0000   SN55= -0.65607  0.03727  0.75378
PATCH AREA=        3.74475 CURVATURES=   0.0470   -0.3036 MEAN & GAUSS=   -0.2566   -0.0143 RADII=   21.284    -3.294

PATCH NUMBER    2
SP00     1.0000    0.0       1.0000    1.0000   SP01   1.0000    3.0000    0.6000    1.0000   SN00 -0.32498  0.12391  0.93757
SP10     2.0000    0.0       1.0000    1.0000   SP11   2.0000    3.0000    0.4000    1.0000   SN50 -0.00000  0.17285  0.98494
FD00     1.0000    0.0       0.3466    0.0      FD01   1.0000    0.0       0.0922    0.0      SN10  0.32202  0.18220  0.92903
FD10     1.0000    0.0      -0.3456    0.0      FD11   1.0000    0.0      -0.3279    0.0      SN01 -0.09103  0.13048  0.98726
SD00     0.0       3.0000   -0.3965    0.0      SD01   0.0       3.0000   -0.3965    0.0      SN51  0.23102  0.16820  0.95830
SD10     0.0       3.0000   -0.5883    0.0      SD11   0.0       3.0000   -0.5883    0.0      SN11  0.30632  0.18320  0.93413
TW00     0.0       0.0      -0.2544    0.0      TW01   0.0       0.0      -0.2544    0.0      SN05 -0.21250  0.12970  0.96851
TW10     0.0       0.0       0.0187    0.0      TW11   0.0       0.0       0.0187    0.0      SN15  0.31389  0.18794  0.93067
FLAGS=2,2,4,1, B1=   0 B2=    1 B3=    3 B4=    0 SP55   1.5000    1.5000    0.6196    1.0000   SN55=  0.11782  0.17527  0.97745
PATCH AREA=        3.12193 CURVATURES=   0.0114   -0.5446 MEAN & GAUSS=   -0.5332   -0.0062 RADII=   87.814    -1.836

PATCH NUMBER    3
SP00     2.0000    0.0       1.0000    1.0000   SP01   2.0000    3.0000    0.4000    1.0000   SN00  0.32202  0.18220  0.92903
SP10     3.0000    0.0       0.0       1.0000   SP11   3.0000    3.0000    0.0       1.0000   SN50  0.76191  0.07493  0.64334
FD00     1.0000    0.0      -0.3466    0.0      FD01   1.0000    0.0      -0.3279    0.0      SN10  0.67552  0.0      0.73734
FD10     1.0000    0.0      -0.9162    0.0      FD11   1.0000    0.0      -0.4556    0.0      SN01  0.30632  0.18320  0.93413
SD00     0.0       3.0000   -0.5883    0.0      SD01   0.0       3.0000   -0.5883    0.0      SN51  0.37251  0.10735  0.92160
SD10     0.0       3.0000    0.0       0.0      SD11   0.0       3.0000    0.0       0.0      SN11  0.41461  0.0      0.91000
TW00     0.0       0.0       0.0187    0.0      TW01   0.0       0.0       0.0187    0.0      SN05  0.31389  0.18794  0.93067
TW10     0.0       0.0       0.4605    0.0      TW11   0.0       0.0       0.4605    0.0      SN15  0.56562  0.0      0.82465
FLAGS=2,2,4,1, B1=   0 B2=    2 B3=    0 B4=    0 SP55   2.5000    1.5000    0.3936    1.0000   SN55=  0.61923  0.09308  0.77960
PATCH AREA=        3.67220 CURVATURES=   0.0907   -0.2819 MEAN & GAUSS=   -0.1912   -0.0256 RADII=   11.029    -3.548

APPROXIMATE TOTAL SURFACE AREA=      10.538872
```

FIGURE 12B. Verification listing for sample surface

CRITICAL CURVATURES ON THE SURFACE

MAXIMUM SIGNED CURVATURE AND RADIUS=	0.09066	11.02867	AT PATCH NUMBER	3
MINIMUM SIGNED CURVATURE AND RADIUS=	-0.54462	-1.83616	AT PATCH NUMBER	2

ADDITIONAL MESH INFORMATION FINAL POINTS AND NORMALS

POINT	X	Y	Z	SNI	SNJ	SNK
	SPLINE NUMBER= 1					
1	0.0	0.0	0.0	-0.67532	-0.02456	0.73712
2	1.00000	0.0	1.00000	-0.32498	0.12391	0.93757
3	2.00000	0.0	1.00000	0.32202	0.16220	0.92903
4	3.00000	0.0	0.0	0.67552	0.0	0.73734
	SPLINE NUMBER= 2					
1	0.0	3.00000	0.10000	-0.56320	-0.02751	0.82585
2	1.00000	3.00000	0.60000	-0.09103	0.13048	0.95726
3	2.00000	3.00000	0.40000	0.30632	0.16320	0.93413
4	3.00000	3.00000	0.0	0.41461	0.0	0.91000

OPT IV - A-X3 XECUTION PHASE - RUN TIME= 0.42 MIN. ENTRY POINT = 0018A428

FIGURE 12C. Verification listing for sample surface in detail later. Note, however, that this display outputs the total surface area and the worst curvatures encountered on the surface.

1.4.5.2 Description of Surface Analysis and Display

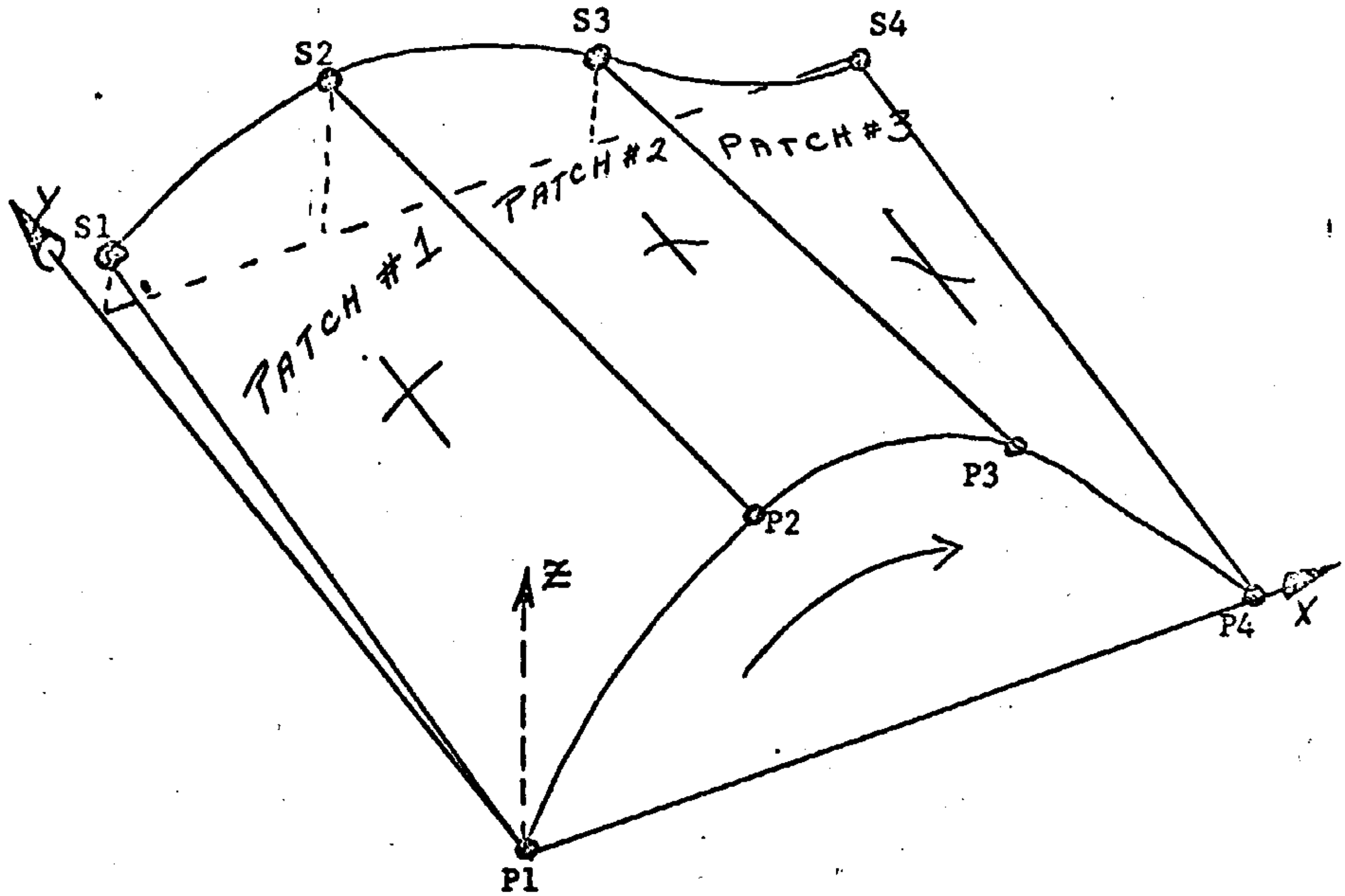
In Figures 12b and 12c of the example part program, there is a detailed analysis output for the surface. This output is admittedly somewhat voluminous -- however it serves two purposes. One purpose is that a great level of detail is required by system programmers to cross check the surfaces which are being generated by this complex experimental processor. But even more important -- certain parts of the information allow a part programmer to judge the quality of the surface without even seeing a picture of it. A description of the output on these figures will be given, including references to additional pictures and graphs as necessary.

The part defined in this sample part program is sketched in Figure 13. Here you see graphically a structure of three surface patches which is generated by a mesh input consisting of two splines of four points each.

Refer now to the analytic output displayed in Figure 12b. The header information is self explanatory. It should be noted that, whereas the total size of the structure in this sample is 115 numbers, approximately 2600 numbers may be used to define a surface in the current system. Also it should be noted that the first point of the first patch is the origin of a local coorinate system and all other points are translated accordingly.

Now continue to scan Figure 12b. Three blocks of information follow -- one for each surface patch. The points and vectors SP00, FD00, etc., are Coon's canonical form for each patch. For more detail on this symbolic notation, refer to the earlier section in this documentation which describes the 'PNTVEC' definition of a Coon's patch. In order to obtain a better intuition on the meaning of these vectors, refer to the magnified picture of patch number 1 in Figure 14. Here, some of these symbolic vectors are plotted on a sketch of patch number 1. The patch is parameterized as a (u, v) unit square with the u-value proceeding in the major spline direction from P1 (SP00) to P2 (SP10) and the v-value proceeding in the cross spline direction from P1 (SP00) to S1 (SP01). In order to gain a quick idea of the orientation of this patch in space, skip to the ninth line of that patch. Here you will find the center point of the patch (u=.5, v=.5) and its unit normal to be:

SP55 = .5, 1.5, .49



S = SSURF/MESH, XYPLAN, SPLINE, P1, P2, P3, P4, $
SPLINE, S1, S2, S3, S4

Figure 13. Sketch of Ruled Mesh Surface Shown Previous Verification Listing

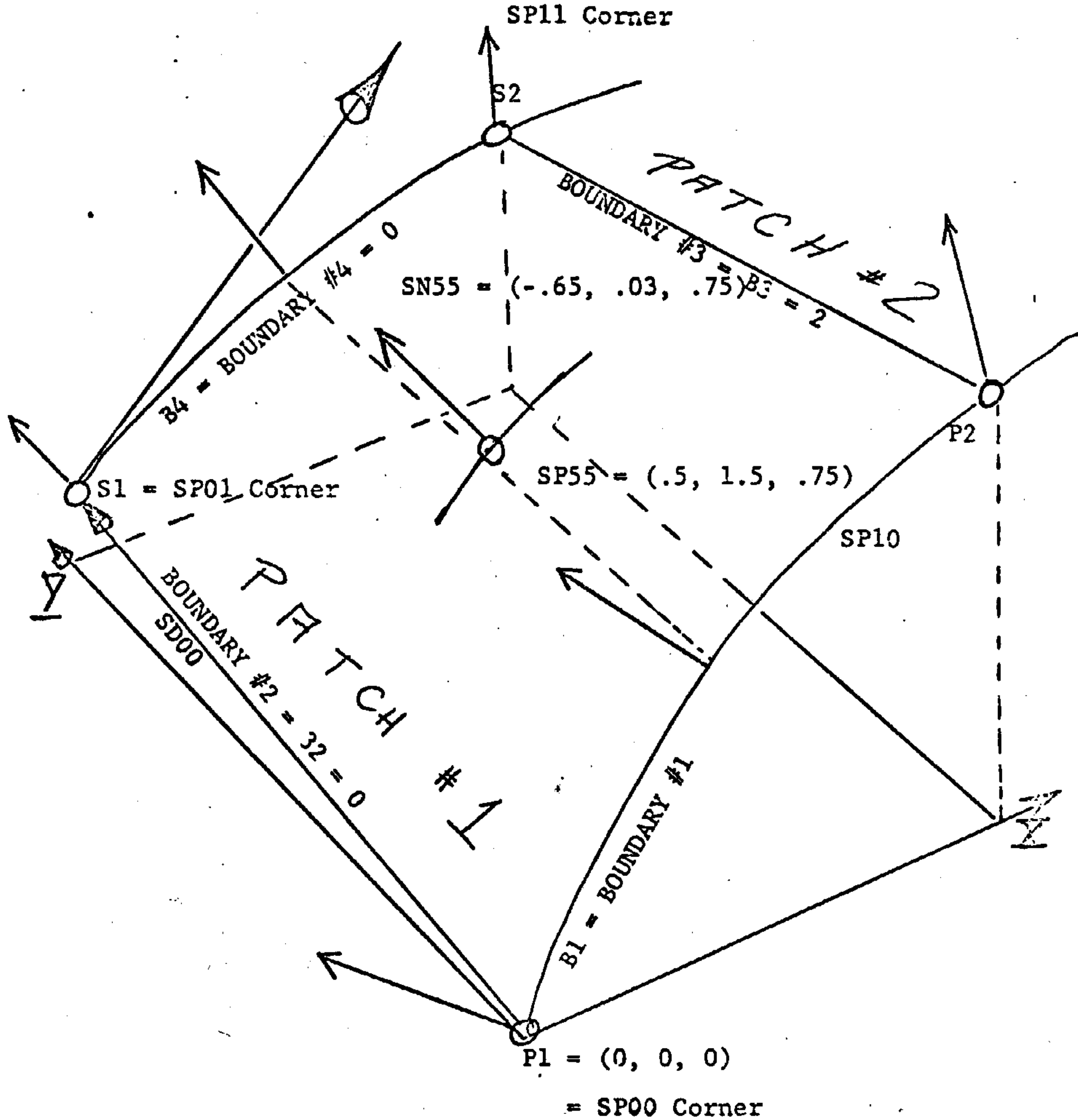


Figure 14. Magnified View of Patch #1 of Sculptured Surface Shown on Previous Figure

SN55 = -.65, .03, .75

In order to obtain the relation of this patch to others in the structure, observe the boundary information on line number 9. The boundaries B1, B2, B3, and B4 are clearly marked on the figure and it can be seen that only boundary B3 has adjacent patches, namely patch number 2. In other words, B3=2. It is of interest to examine surface normals displayed on the right-most columns of the page -- especially since this is a ruled surface. In particular, three surface normals are output along the center (u=0.5) ruling of the patch, namely SN50, SN55, and SN51. Note that these surface normal vectors are not parallel so that this patch is a twisted ruled surface as opposed to a developable one. In case that these normals were all parallel -- then and only then would it be possible to 'develop' the surface -- that is roll it without wrinkles or stretching onto a plane. Furthermore, since these normals are different -- it is not truly correct to lay a cutter along a ruling or to move across a ruling with a single cut vector. Some gouging in either case will occur. However, by examining this output, the programmer may make some judgment on the severity of this 'twisting' effect.

In addition to Coon's canonical vectors and surface normals, the last line for patch number 1 describes surface curvatures at the patch center (Figure 12b). If a patch has a generally consistent curvature, then the curvature analysis at the center of a patch is frequently of use in judging the character of the complete patch. However, there are times when curvature of the patch center is not representative of the total curvature of the patch and the use of this output is a judgment factor on the part of the programmer.

There are four curvatures and two radii output at the center of patch number 1. Curvature is nothing more than the inverse of the radius. For example, a curvature of 2 would signify a radius of 1/2. Curvature is a preferred quantity because a plane has a curvature of 0, but its radius is infinity. Surface curvature is also a signed quantity. The idea of sign is based on the 'natural' surface normal to the surface. If the surface curvature is 'cupped' toward the normal, then the curvature (and radius) is positive, while a curvature 'cupped' away from the surface normal is negative. The picture of curvature is still not complete because a surface really has many curvatures at a point. A saddle surface, for example, has both positive and negative curvatures at every point! Curvature in any direction at a point on a surface is measured by slicing the surface with a plane which contains the surface normal. As this plane is rotated around the surface normal as an axis two extreme curvatures are encountered (a largest and a smallest). These two extreme curvatures are called the principal curvatures of the surface, and they actually occur in perpendicular directions.

The first two numbers printed on the last line of the output display (Figure 12b) for patch number 1 (at 'CURVATURES AT SP55'=_, are the two principal surface curvatures. Likewise, the radii output at the end of this line ('RADII=') are the reciprocals of these numbers. Because these numbers are opposite in sign, the programmer can immediately deduce that the surface does have a gentle inflection. Also the sharper curvature (-3.294 radius), curves away from the 'natural' surface normal (SN55=(-.65, .037, 0.75)). This is expected from the definition. It is surprising that one of the curvatures is <u>not</u> zero -- because, after all, this is a ruled surface. What could be wrong! The answer is nothing! As mentioned earlier, the ruled surface is a 'twisted' ruled surface so that there is no zero curvature direction on the surface -- in a sense, surface characteristics are quite different than the character of curves which generate the surface. Thus, straight lines can generate the ruled surface, but the curvature of the surface is not necessarily zero.

Two additional curvatures are output at the center of a patch -- the so-called Mean and Gaussian curvatures. These curvatures are merely the arithmetic mean and product of the two principal curvatures as can be seen below.

Principal curvatures K1 = .04697 and K2 = -0.30358

Mean curvature = K1 + K2 = .04697 - .30358 = -.2566

Gaussian curvature = K1 * K2 = $\frac{(.04697)*(-.30358)}{2}$ = -.01426.

Gaussian curvature is a very informative quantity. A positive Gaussian curvature means <u>no</u> inflection, while a negative Gaussian curvature means the surface does have an inflection. (An inflection means that the surface cuts its own tangent plane). The only time the Gaussian curvature is zero is for developable surfaces -- i.e., one that can be rolled out on a plane without changing relative distances on a surface. There is also a very direct interpretation of the Gaussian curvature, namely the ratio of the <u>spread</u> of the surface normals over a region around a surface point to the area of the region itself. This number, better than any other, describes surface curvature at one point.

This completes the description of the analysis provided for individual patches. As mentioned already, the display is detailed, but certain key quantities can quickly be extracted which will tell the programmer a great deal about the surface of patches which he has defined.

The second item on the patch display (Figure 12c) consists of three bits of information about the complete surface -- namely the total surface area can be useful for two purposes -- it tells the programmer how much area he must clear, and it can inform him of gross errors in his input. That is, if by accident, one point is drastically misplaced in the mesh definition, and if the mesh definition processes without error, he may then notice a larger surface area than he was expecting -- leading him to further study the situation. The two worst curvatures and radii are obtained by saving the algebraically largest and the smallest curvatures computed at the patch centers. They inform the programmer, based on his judgment as mentioned before, of the most extreme curvatures encountered on the surface. The programmer may then use these curvatures as a quick measure of whether the surface inflects at all (if both worst curvatures have the same sign, there is no inflection), and of how severe a curvature is present. In addition, the programmer can use this information in judging the largest cutter radius which can be applied to the surface without gouging. Beware, that this judgment depends on the side from which the surface is to be cut. The data output in our example indicates that the 'natural' surface normal is generally plus Z oriented. In order to apply the cutter from the plus Z side of the surface, the dangerous surface radii are those which are cupped toward the cutter (in other words, positive surface radii). The largest positive radii encountered is 11.02887 at the center of patch number 3, so that, in effect, there is no significant gouge problem from the plus Z side of the surface. On the other hand, in cutting the female version of the same surface, the cutter radius should not be larger than 1.83616 inches, and this curvature occurs at the center of the second patch.

So much for the generalized surface numbers. In the special case where a surface is defined by a mesh, a simple display of the final points and normals on each spline is printed. Recall that if weight constraints are used in the spline definition, then the points displayed may be different from the input points -- otherwise, they should be identical.

SECTION 2

PART PROGRAMMER GUIDE

TO

SSIP

2. PART PROGRAMMERS GUIDE TO SSIP

2.1 General Remarks

The purpose of this section is to assist the part programmer in the use of SSIP. The programming techniques indicated have been tested but not rigorously in the sense that any guarantee could be attached to the reliability or capability of the system or its documentation. The major advantage over other systems is an improved reliability and performance efficiency for a useful set of surfaces and cutter paths.

This documentation was not designed to stand alone but rather to be used in conjunction with the System/360 APT Part Programming Manual. It is concerned only with those features which are peculiar to SSIP or which differ, in one respect or another, from what is found in IBM ARELEM. At the end of each section segment, when applicable, cross reference is made to the manual mentioned above, more specifically, the Version IV Edition, bearing the serial number GH20-0309-4.

There are some items which, conceivably, should have been included and were not, such as multiple intersections and the special feedrate option. It was felt that the differences here between SSIP and IBM ARELEM were insignificant.

2.2 CUTTER

The format of the cutter defining statement is:

CUTTER/d,r,0,0,0,b,h

where:

d = diameter at cutting end
r = corner radius
b = bevel angle in degrees
h = height

These parameters are shown in Figure 15. SSIP does not recognize the third, fourth, fifth (e,f,a) parameters of the System/360 APT CUTTER statement. These parameters must be set to zero. Any combination of d, r, b, h which produces a real shape is permissible. The parameters (d,r,h) must be positive, as in System/360 APT, but the parameters (b) may be negative and the parameter (h) may be less than the corner radius (r). Of the 4 tool elements - top, side, corner radius, end flat - any or all may reduce to zero. Two alternate forms of the CUTTER statement are acceptable:

CUTTER/d,r

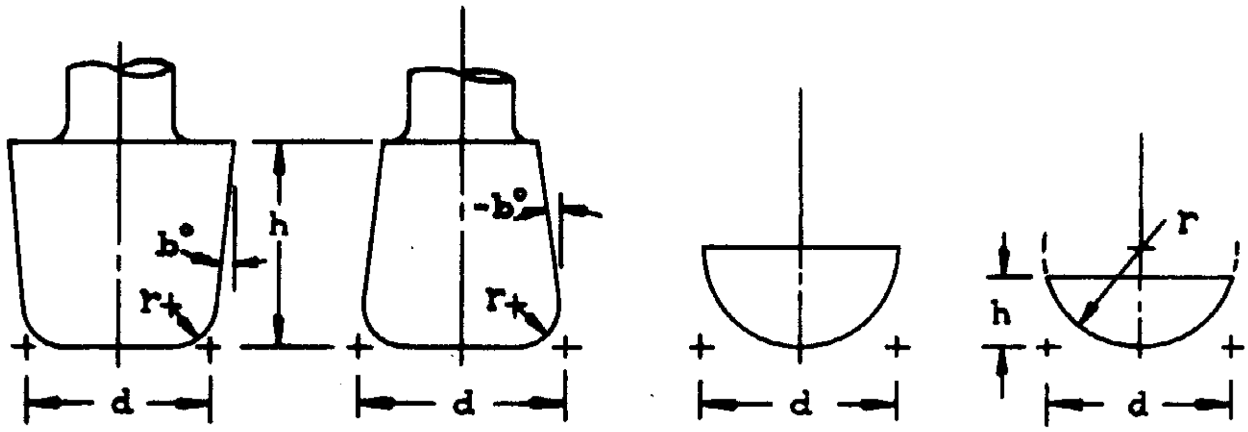


Figure 15

Undercutting:

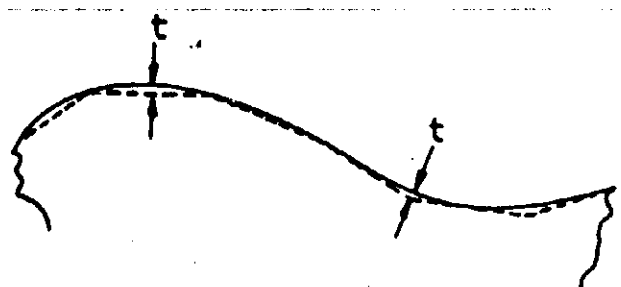


Figure 16

Overcutting:

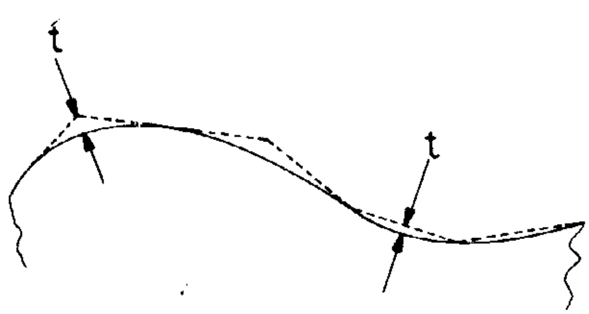


Figure 17

CUTTER/d

It should be noted that where (h) is not specified, it is equated to (r); where (r) is not specified, it is equated to (0); where the CUTTER statement is not specified, or where it is invalid, all parameters are equated to (0) and processing continues.

Diagnostics:

-617 Contact point not between tool top and bottom
1010 CUTTER/--- statement no good

Cross Reference: System/360 Part Programming Manual: 5.01.01.

2.3 Tolerance Specification

2.3.1 INTOL, OUTTOL and TOLER: are three words which relate to the tolerance control of a contoured surface. In System/360 APT, INTOL stipulates undercutting; OUTTOL and TOLER stipulate overcutting. Undercuts on a convex surface are chordal; undercuts on a concave surface are tangential. The opposite is true in overcutting. Overcuts on a convex surface are tangential; overcuts on a concave surface are chordal. Refer to Figures 16 and 17.

SSIP accepts each of these three words but does not apply them as described above. In SSIP the cut vectors, as related to the surface, are always chordal. Accordingly, there is undercutting where the surface is convex and overcutting where the surface is concave. The allowable crown height, or maximum distance from curve to chord, is the sum of INTOL and OUTTOL. Default settings are as follows: INTOL/.0005, OUTTOL/.0005 and TOLER/.001. When TOLER/n is specified, OUTTOL is set equal to TOLER and, if n is greater than .0005, INTOL is set to zero. If n is less than .0005, INTOL remains unchanged. Finally, SSIP will ignore INTOL/n, if n is less than .00001. The following examples point up usage and effect:

a. Default (See Figure 18)

```
C1 = CIRCLE/11,11,10
L1 = LINE/3,11,7,11
PT1 = POINT/XSMALL,INTOF,L1,C1
GOTO/PT1
CUT,PENDWN,INDIRV/1,1,0
GOFWD/C1,L1
PENUP,DNTCUT
```

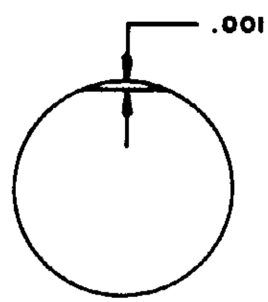


Figure 18

b. INTOL only (See Figure 19)

```
INTOL/.005
C1 = CIRCLE/11,11,10
L1 = LINE/3,11,7,11
PT1 = POINT/XSMALL,INTOF,L1,C1
GOTO/PT1
CUT,PENDWN,INDIRV/1,1,0
GOFWD/C1,L1
PENUP,DNTCUT
```

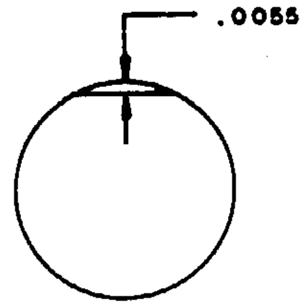


Figure 19

c. INTOL and OUTTOL (See Figure 20)

```
INTOL/.005
OUTTOL/.005
C1 = CIRCLE/11,11,10
L1 = LINE/3,11,7,11
PT1 = POINT/XSMALL,INTOF,L1,C1
GOTO/PT1
CUT,PENDWN,INDIRV/1,1,0
GOFWD/C1,L1
PENUP,DNTCUT
```

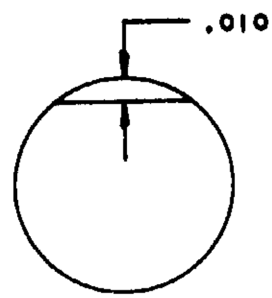


Figure 20

2.3.2 GOUGCK

Sometimes the diameter of the cutter is too large as related to the sharp curvature of the surface or, conversely, the curvature of the surface is too sharp as related to the large diameter of the cutter. Such a relationship can make for an erratic cutter path. For this reason the GOUGCK feature was added to SSIP. It applies only to the drive surface.

```
GOUGCK/ON
GOUGCK/OFF
```

Originally, it was felt that the special calculation phase occasioned by GOUGCK/ON was very time consuming, but this is not the case. Accordingly, the default is GOUGCK/ON and this is contrary to the default in System/360 APT.

2.3.3 MAXDP

Relates to tolerance, inasmuch as it controls maximum step size. In System/360 APT it applies only to steps arrived at by iterative rather than analytical methods. By way of example, it would not apply if DS, PS and CS were planes. The default setting in System/360 APT is ten (10). The default setting in SSIP is two (2), and it applies to any motion which involves DS, PS and CS.

Cross Reference: System/360 Part Programming Manual: 5.01.02, 5.05.

2.4 Cutter to Surface Relationships

2.4.1 Part Surface

Two APT words (modal) describe the relationship between cutter and part surface:

TLONPS
TLOFPS

TLONPS specifies that contact between cutter and part surface be at the tool end point. TLOFPS implies that it be at the point of tangency between cutter and surface, by specifying that the tool end point be off the part surface. If neither is specified, TLOFPS is assumed with one exception: when TLAXIS/NORMPS is in effect, the cutter to surface relationship is always TLONPS, any absence of specification or specification to the contrary notwithstanding.

2.4.2 Drive Surface

The following APT words (modal) describe the relationship between cutter and drive surface:

TLLFT
TLRGT
TLON,TLNDON

If the part programmer imagines himself as sitting on top of the cutter and looking in the direction of motion then TLLFT relates to the left side of the drive surface, TLRGT to the right side of the drive surface. TLON and TLDNON are synonymous, each specifiying that the tool end point is to be located on the drive surface. SSIP differs from System/360 APT in three respects. In SSIP, the default is TLON. Secondly, if a modifier is specified which does not agree with the current position of the cutter, the position of the cutter is changed to agree with the modifier. A -501 warning is issued -- An unprogrammed move was output by SSIP. Thirdly, when SSIP encounters a GOBACK statement it changes the TLLFT/TLRGT settings. The new setting is modal.

2.4.3 Check Surface

The following APT words (non modal) describe the relationship between cutter and check surface:

TO
ON
PAST
TANTO,DSTAN

There is no difference between SSIP and System/360 APT in the meaning or usage of TO, ON or PAST and the default in each processor is TO. In regard to the synonyms TANTO and DSTAN, SSIP will honor them only if the subject check surface is a plane, cylinder or cone. SSIP differs also from System/360 APT inasmuch as it does not recognize PSTAN.

2.4.4 Multiple Check Surfaces

The System/360 Part Programming Manual states that the maximum number of check surfaces allowed for a cut sequence is three. Not so in SSIP where the maximum is two. Another difference is this: SSIP requires each surface to be in the path of the cutter, IBM ARELEM does not. TRANTO, although not restricted to, is used in conjunction with Multiple Check Surfaces. SSIP and IBM ARELEM both require that the transfer be to a statement which will generate a PROTAP record. SSIP further requires that the transfer be forward and not backward or, in other words, to a subsequent and not a previous statement.

Diagnostics:

282 PS is parallel DS cannot proceed
329 (PERPTO) vector direction too far from forward sense
1703 Cosine argument, angle between TLAXIS and surface normal, no good
1704 TLON mode not allowed

Cross Reference: System/360 Part Programming Manual: 5.02.01, 5.02.02, 5.02.03.

2.5 Tool Axis Specification

Tool axis control is effected by means of the TLAXIS statement. SSIP allows this statement to take on many forms. Some of these forms are recognized by other processors, some are not. Turnabout, some forms which are recognized by other processors are not recognized by SSIP. All tool axis settings are modal with the exception of TLAXIS/i,j,k described in 2.5.2.

2.5.1 TLAXIS/1 (Modal)

Specifies that the current tool axis orientation is to be maintained until a new one is specified.

2.5.2 TLAXIS/i,j,k (Non Modal)

Allows for tool axis setting per vector direction (i,j,k). It is used most often for purposes of initialization, more specifically, to establish a sense of direction: which way is up or down! It will be maintained until changed in point to point programming. It will be lost when the programmed motion involves PS, DS and CS. It may be locked in by using TLAXIS/1.

2.5.3 TLAXIS/NORMPS (Modal)

Specifies that during a cut sequence the tool axis is to be always normal to the part surface. The axis will change, as required, to meet this condition. As noted in 2.4.1, when TLAXIS/NORMPS is in effect, the cutter to surface relationship is always TLONPS any absence of specification or specification to the contrary notwithstanding.

2.6 Startup Procedures

The GO/ statement is used to position the cutter with regard to one, two or three surfaces. The general format is:

```
        TO          TO          TO
     GO/ON   , DS,  ON   , PS,  ON   , CS
        PAST        PAST        PAST
```

Startup is frequently a problem to the part programmer and this is not something which is peculiar to SSIP. He will fare better, if he keeps in mind the following:

1. Theoretically, the more information given, the greater the likelihood that the cutter will be positioned properly. Therefore a three surface startup should be better than a two which, in turn, should be better than a one.

2. In spite of what was written above, it is not always advisable to use INDIRV or INDIRP. These initiate a one way search, whereas the default is a two way search. However, one or the other must be used when there are two positions which will satisfy the given conditions e.g. intersection of a plane and cylinder.

3. Set the TLAXIS perpendicular or approximately perpendicular to the PS. The TLAXIS may not be parallel to the PS or perpendicular to the DS.

4. Immediately before the GO/ command the cutter, not merely its axis, should be off all three surfaces: DS, PS and CS. Moreover, if any of the three is a closed surface avoid starting up from its centroid or axis. This applies to CIRCLE, CYLNDR, SPHERE.

Diagnostics:

1	NUMPTS exceeded without finding CS
128	Forward sense may not be PERPTO DS at cut start

198 TLAXIS may not be parallel PS at cut start
204 TLAXIS may not be parallel PS at cut start
519 Too many surfaces given in start-up
2851 Tool is on surface centroid (move it slightly)
5130 Forward sense may not be parallel TLAXIS during startup

Cross Reference: System/360 Part Programming Manual: 5.03.02, 5.03.02.01, 5.03.02.02, 5.03.02.03.

2.7 The THICK Statement

This statement (modal) specifies that a positive or negative thickness be applied to any or all of the surfaces used in a motion or startup statement. The format of this statement is:

THICK/THPS,THDS,THCS

where:

THPS is the thickness to be applied to Part Surface
THDS is the thickness to be applied to Drive Surface
THCS is the thickness to be applied to Check Surface

A positive THPS will bring the cutter up off the PS; a negative THPS will move the cutter down into the PS. THDS and THCS are dependent upon the relationship of cutter to surface. If TLRGT is in effect a positive value will move the cutter to the right, negative to the left. If TLLFT is in effect a positive value will move the cutter to the left, negative to the right. The positional modifier TLON (DS) nullifies THDS; the positional modifier ON (CS) nullifies THCS.

Cross Reference: System/360 Part Programming Manual: 14.01.05.

2.8 Unprogrammed Moves

In treating the relationship between cutter and DS, it was stated that if the tool position does not agree with the positional modifier: TLLFT, TLON, TLRGT, then SSIP will generate an unprogrammed move in favor of the modifier and issue a -501 warning: "An unprogrammed move was output by SSP." Actually, unless instructed otherwise, SSIP will issue this warning and generate an unprogrammed move whenever the entry to a motion sequence is not the logical start position. By way of example, this situation will arise where there is a mismatch of surfaces and SSIP is asked to move the cutter from one to the other. It matters not whether the surfaces are DS, as described and shown below, or PS, not shown. In Figure 21 motion is along S1 to PL1. If the next instruction

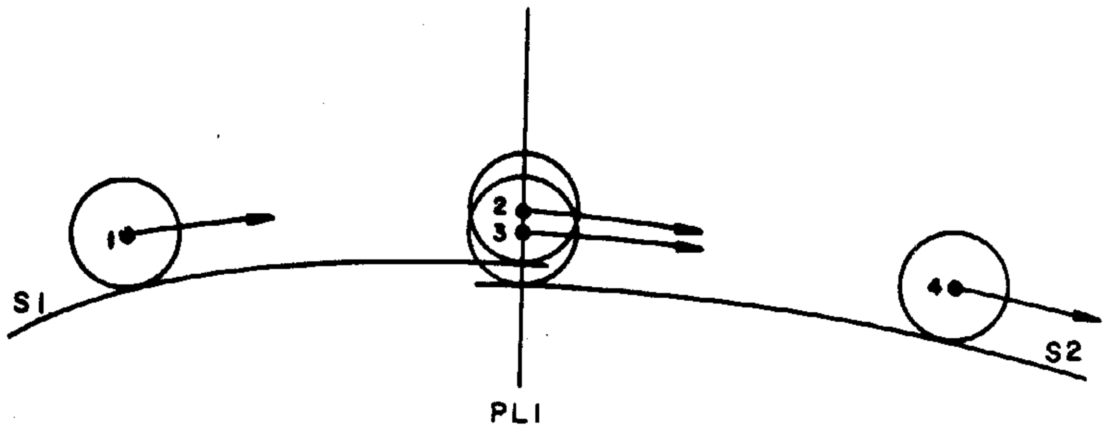


Figure 21

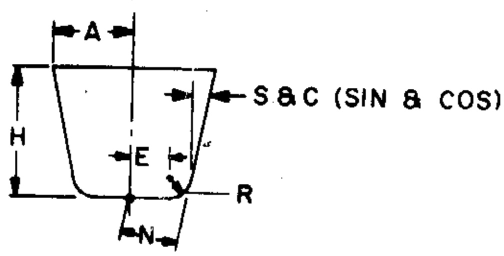


Figure 22

is a GOFWD/S2, SSIP will issue the warning, generate an unprogrammed move from position #2 to position #3 and then move on to position #4. The programmer can control the issuance of warning and move by use of the statement:

```
DELETE/ON
       OFF
```

The default is DELETE/OFF and, when in effect, the processor will perform as described above. If DELETE/ON is in effect the processor will issue neither warning nor move. In Figure 21 motion will be from position #2 to position #4. The programmer who uses DELETE/ON should exercise greater care in verifying his program.

Diagnostics:

-501 Warning - an unprogrammed move was output by SSP

2.9 Debugging Aids

2.9.1 DEBUG

This feature is activated by the statement:

```
DEBUG/SEC2,ON
```

If activated, SSIP will print an abundance of information for debugging purposes: intermediate tool positions, flag settings, direction senses and the like. It will continue to do so until it encounters the terminating member of the couplet:

```
DEBUG/SEC2,OFF
```

This feature is designed for use by the system programmer. DYNDMP is not available.

2.9.2 ERRPRT

Designed for the part programmer is an abbreviated, automatic print which is put out whenever SSIP issues an error message. This does not apply to warnings. Nor will it be put out if DEBUG/SEC2,ON is in effect. The error message and debugging information are arranged and interpreted like so:

```
$$$$$     ERROR NO. _____ $$$$$
***  DEBUG INFO FOR ISN _____ ***
  *** POSITION OF TOOL AT START OF PASS ***

TEX    TEY    TEZ    WI    WJ    WK    FVA    FVB    FVC
---    ---    ---    --    --    --    ---    ---    ---
```

(Tool End)			(Tool Axis)		(Forward Vector)	
CUTH	CUTS	CUTC	CUTR	CUTE	CUTN	CUTA
----	----	----	----	----	----	----

In Figure 22 CUTH = H
CUTS = S
CUTC = C
CUTR = R
CUTE = E
CUTN = N
CUTA = A

This is followed by the TLAXIS specifications: 1, NORMPS, etc. and the APT statement which generated the pass. Finally, when available, the first and last ten points of the pass are listed under the following format:

MENDFL X Y Z I J K FVA FVB FVC CSDIS

where:

MENDFL	= flag which indicates the position of the point in the pass: (-) Startup; (+) Endup; (0) in between
X,Y,Z	= location of tool end
I,J,K	= tool axis orientation
FVA,FVB,FVC	= forward vector
CSDIS	= true distance to CS

__Diagnostics__:

-901 Warning - tool to PS contact point not as expected (Printed only if DEBUG/SEC2,ON)
-902 Warning - tool to DS contact point not as expected (Printed only if DEBUG/SEC2,ON)
-903 Warning - tool to CS contact point not as expected (Printed only if DEBUG/SEC2,ON)

__Cross Reference__: System/360 Part Programming Manual: 15.02.01.

2.10 Defaults

__Word/Phrase__	__Reference__	__Comment__
CUTTER/0	2.2	
INTOL/.0005	2.3.1	
OUTTOL/.0005	2.3.1	
TOLER/.001	2.3.1	
GOUGCK/ON	2.3.2	
MAXDP/2	2.3.3	
TLOFPS	2.4.1	Not so, if TLAXIS/NORMPS

TLON	2.4.2
TO	2.4.3
TLAXIS/1	2.5.1
TLAXIS/0,0,1	2.5.2
DEBUG/SEC2,OFF	2.9.1
SSIP	
MULTAX/OFF	

2.11 Diagnostics

The following diagnostics are generated by SSIP. As in System/360 APT a minus (-) sign before a number indicates a warning rather than an error.

Number	Text
1	NUMPTS exceeded without finding CS.
10	System Read/Write error.
11	System Read/Write error.
41	System Read/Write error.
128	Forward sense may not be PERPTO DS at cut start.
198	TLAXIS may not be parallel PS at cut start.
204	TLAXIS may not be parallel PS at cut start.
222	OFFSET not allowed.
282	PS is parallel DS cannot proceed.
329	(PERPTO) vector direction too far from forward sense.
334	Invalid surface type (poorly defined).
343	Tool must not be on DS here.
348	Forward sense must not be PERPTO DS here.
352	Surface axis indefinite (vector too small) or TLAXIS is incorrect.
360	SECUT/--- statement no good.
462	TLAXIS may not be PERPTO DS.
-501	Warning - an unprogrammed move was output by SSP.
519	Too many surfaces given in start-up.
524	Indicated move is too small (less than .001).
542	Pass end indeterminate for this DS-CS TANTO case.
543	Pass end indeterminate for this DS-CS TANTcase.
610	Tool to TABCYL failure (check geometry).
-617	Contact point not between tool top and bottom.
-620	BEVELS/--- statement no good.
666	Tool vs. QADRIC failure (non-convergence in 10 tries).
668	Tool to TABCYL failure (check geometry).
-716	Warning - End-up not as expected. See Section II diagnostics.
740	Tool forward motion has stopped PS-DS angle no good, radius curve too small etc.
-901	Warning - tool to PS contact point not as expected. (Printed only if DEBUG/SEC2,ON).
-902	Warning - tool to DS contact point not as expected. (Printed only if DEBUG/SEC2,ON).

-903	Warning - tool to CS contact point not as expected. (Printed only if DEBUG/SEC2,ON).
1010	CUTTER/--- statement no good.
1703	Cosine argument, angle between TLAXIS and surface normal, not in range.
1704	TLON mode not allowed.
2102	25 warnings on one pass.
2851	Tool is on surface centroid (move it slightly).

2.12 DISPLY/6

Prints 'PSIS' patch boundary points in Section 3. Data is sequenced by U then V parameters. Surfaces of revolution are not supported.

EXAMPLE:

```
JOE=SSURF
PSIS/JOE
DISPLY/6
```

SECTION 3

TEST CASES

3.1 Fuselage Mounted Pylon and Nacelle

```
PARTNO  SSIP/SSX3A -TEST 1- FUSELAGE MOUNTED PYLON AND NACELLE   (CHUBB)
$$  PURPOSE -  1. DEFINE A SET OF SCULPTURED SURFACES THAT REPRESENT
$$                A TYPICAL GROUP OF AIRCRAFT LOFTED SURFACES.
$$
$$             2. DEFINE A LARGE NUMBER OF SCULPTURED SURFACES IN
$$                A SINGLE APT PART PROGRAM.
$$
$$             3. DEFINE A LARGE VARIETY OF SSX3A SCULPTURED SURFACES
$$
$$             4. PROVIDE A SET OF SURFACES THAT MAY BE USED IN
$$                SUBSEQUENT TEST CASES.
$$
$$             5. TEST REFSYS.
$$
$$             6. GENERATE MOTION ON THE SURFACES(VARIETY OF CUTS).
$$
$$             7. TEST THE GO/ DS,PS,CS UNDER A NUMBER OF DIFFERENT
$$                CONDITIONS.
$$
$$-----------------------------------------------------------------------
MULTAX
CLPRNT
TLAXIS/NORMPS
CANON/ON
MAXDP/ 12
TOLER/.0025
SSON  =SSURF/DISPLY,ON
$$-----------------------------------------------------------------------
$$
$$   SCULPTURED SUFACE -NAME-      -TYPE-
$$
$$ (FUSELAGE)            FUS     =  SSURF/MESH,XYZ
$$ (PYLON FWD)           PYLONF  =  SSURF/RULED,(SPLINE)
$$ (PYLON UPPER AFT)     PYLNUA  =  SSURF/MESH,XYZ,(2 SPLINE RULED SURF)
$$ (PYLON LOWER AFT)     PYLNLA  =  SSURF/MESH,XYZ,(2 SPLINE RULED SURF)
$$ (NACELLE FWD OUTBD)   NACFOB  =  SSURF/PATCH,PNTVEC
$$ (NACELLE FWD INBD)    NACFIB  =  SSURF/PATCH,PNTVEC
$$ (NAC. NOZZLE OUTBD)   NNOZO   =  SSURF/REVCLV,(CURSEG)
$$ (NAC. NOZZLE INBD)    NNOZI   =  SSURF/REVCLV,(CURSEG)
$$ (INLET OUTBD)         INLETO  =  SSURF/REVCLV,(SPLINE)
$$ (INLET INBD)          INLETI  =  SSURF/REVCLV,(SPLINE)
$$-----------------------------------------------------------------------
XV = VECTOR / 1,0,0
YV = VECTOR / 0,1,0
ZV = VECTOR / 0,0,1
NOR= POINT  / 0,0,0
PRINT/0
$$$$$ DEFINITION OF FUS     $$$$$
$$
RESERV/FP,64
SYN/ POINT,PT
$$
FP(  1)=PT/  52.6092, 818,  63.0258
FP(  2)=PT/  52.2075, 838,  62.7441
FP(  3)=PT/  51.6382, 858,  62.3455
```

```
FP(  4)=PT/  50.8079, 878,  61.7640
FP(  5)=PT/  49.6375, 898,  60.9446
FP(  6)=PT/  48.0820, 918,  59.8554
FP(  7)=PT/  46.1009, 938,  58.4682
FP(  8)=PT/  44.9228, 948,  57.6433
FP(  9)=PT/  57.7064, 818,  53.0969
FP( 10)=PT/  57.1455, 838,  52.8353
FP( 11)=PT/  56.4104, 858,  52.4926
FP( 12)=PT/  55.3715, 878,  52.0081
FP( 13)=PT/  53.9226, 898,  51.3325
FP( 14)=PT/  52.0075, 918,  50.4395
FP( 15)=PT/  49.5763, 938,  49.3058
FP( 16)=PT/  48.1292, 948,  48.6310
FP( 17)=PT/  60.9279, 818,  42.5136
FP( 18)=PT/  60.1991, 838,  42.3183
FP( 19)=PT/  59.2968, 858,  42.0765
FP( 20)=PT/  58.0545, 878,  41.7436
FP( 21)=PT/  56.3382, 898,  41.2838
FP( 22)=PT/  54.0822, 918,  40.6793
FP( 23)=PT/  51.2287, 938,  39.9147
FP( 24)=PT/  49.5307, 948,  39.4597
FP( 25)=PT/  62.2244, 818,  31.6319
FP( 26)=PT/  61.3397, 838,  31.5545
FP( 27)=PT/  60.2886, 958,  31.4625
FP( 28)=PT/  58.8695, 878,  31.3384
FP( 29)=PT/  56.9242, 898,  31.1682
FP( 30)=PT/  54.3755, 918,  30.9452
FP( 31)=PT/  51.1537, 938,  30.6633
FP( 32)=PT/  49.2394, 948,  30.4958
FP( 33)=PT/  61.9176, 818,  20.7709
FP( 34)=PT/  60.9249, 838,  20.8577
FP( 35)=PT/  59.7296, 858,  20.9623
FP( 36)=PT/  58.1200, 878,  21.1031
FP( 37)=PT/  55.9294, 898,  21.2948
FP( 38)=PT/  53.0792, 918,  21.5441
FP( 39)=PT/  49.5168, 938,  21.8558
FP( 40)=PT/  47.4346, 948,  22.0380
FP( 41)=PT/  60.2531, 818,  10.0432
FP( 42)=PT/  59.1625, 838,  10.3354
FP( 43)=PT/  57.8009, 858,  10.7002
FP( 44)=PT/  55.9368, 878,  11.1997
FP( 45)=PT/  53.4419, 898,  11.8682
FP( 46)=PT/  50.3017, 918,  12.7096
FP( 47)=PT/  46.5039, 938,  13.7273
FP( 48)=PT/  44.3338, 948,  14.3087
FP( 49)=PT/  57.2002, 818,   -.4849
FP( 50)=PT/  55.9726, 838,    .0875
FP( 51)=PT/  54.3162, 858,    .8598
FP( 52)=PT/  52.1634, 878,   1.8637
FP( 53)=PT/  49.4540, 898,   3.1272
FP( 54)=PT/  46.1781, 918,   4.6547
FP( 55)=PT/  42.3266, 938,   6.4507
FP( 56)=PT/  40.1671, 948,   7.4577
FP( 57)=PT/  53.5726, 818,  -8.6154
FP( 58)=PT/  52.1460, 838,  -7.6760
```

```
FP( 59)=PT/  50.3198, 858,  -6.4901
FP( 60)=PT/  48.0504, 878,  -5.0163
FP( 61)=PT/  45.2939, 898,  -3.2262
FP( 62)=PT/  42.0435, 918,  -1.1154
FP( 63)=PT/  38.2921, 938,   1.3207
FP( 64)=PT/  36.2156, 948,   2.6692
$$
NOFS =POINT/ 85, 835, 28
NPT2 =POINT/ 75, 935, 28
ZNO  =PLANE/ 0,0,1,28
XNO  =PLANE/ PERPTO,ZNO, NOFS, NPT2
YNO  =PLANE/ NOFS, PERPTO, XNO, ZNO
NS2FS= MATRIX/ (PLANE/PARLEL,XNO,XLARGE,0),(PLANE/PARLEL,YNO,YLARGE,O),$
               (PLANE/PARLEL,ZNO,ZLARGE,0)
FS2NS= MATRIX/ INVERS,NS2FS
PRINT/0
REFSYS / FS2NS
FUS=SSURF/MESH,XYZ,    $ FUSELAGE SURFACE
             SPLINE,FP(  1,THRU,  8),$
             SPLINE,FP(  9,THRU, 16),$
             SPLINE,FP( 17,THRU, 24),$
             SPLINE,FP( 25,THRU, 32),$
             SPLINE,FP( 33,THRU, 40),$
             SPLINE,FP( 41,THRU, 48),$
             SPLINE,FP( 49,THRU, 56),$
             SPLINE,FP( 57,THRU, 64)
REFSYS / NOMORE
PRINT/0
$$$$$ DEFINITION OF PYLONE  $$$$$
$$
RESERV/ PPF,23
PPF( 1)= POINT/ -18.   ,  50.    ,  -3.640  $$  M= .0535
PPF( 2)= POINT/ -18.   ,  45.    ,  -3.862
PPF( 3)= POINT/ -18.   ,  40.    ,  -4.010
PPF( 4)= POINT/ -18.   ,  35.    ,  -4.063
PPF( 5)= POINT/ -18.   ,  30.    ,  -4.025
PPF( 6)= POINT/ -18.   ,  25.    ,  -3.835
PPF( 7)= POINT/ -18.   ,  20.    ,  -3.442
PPF( 8)= POINT/ -18.   ,  15.    ,  -2.630
PPF( 9)= POINT/ -18.   ,  11.5   ,  -1.544
PPF(10)= POINT/ -18.   ,  10.5   ,   -.927
PPF(11)= POINT/ -18.   ,  10.15  ,   -.530
PPF(12)= POINT/ -18.   ,  10.00  ,   0.      $$   ZV
PPF(13)= POINT/ -18.   ,  10.15  ,    .530
PPF(14)= POINT/ -18.   ,  10.50  ,    .927
PPF(15)= POINT/ -18.   ,  11.50  ,   1.544
PPF(16)= POINT/ -18.   ,  15.    ,   2.630
PPF(17)= POINT/ -18.   ,  20.    ,   3.442
PPF(18)= POINT/ -18.   ,  25.    ,   3.835
PPF(19)= POINT/ -18.   ,  30.    ,   4.025
PPF(20)= POINT/ -18.   ,  35.    ,   4.063
PPF(21)= POINT/ -18.   ,  40.    ,   4.010
PPF(22)= POINT/ -18.   ,  45.    ,   3.862
PPF(23)= POINT/ -18.   ,  50.    ,   3.640   $$  M=-.0535
$$
```

```
PYLV=VECTOR / 0,1, .0535
PYUV=VECTOR / 0,1,-.0535
PRINT/0
PYLSPL = SCURV / SPLINE,  PPF(1),TANSPL, PYLV, PPF(2,THRU,12),TANSPL,  $
                           ZV, PPF(13,THRU,23),TANSPL,PYUV
PRINT/0
PYLONF = SSURF/ RULED, PYLSPL, XAXIS, XV
PRINT/0
$$$$$ DEFINITION OF PYLNUA  $$$$$
$$
RESERV/ PPA,12
PPA( 1)= POINT/ -28.5   ,  50.   ,   3.640  $$  M=-.0535
PPA( 2)= POINT/ -28.5   ,  60.   ,   3.050
PPA( 3)= POINT/ -28.5   ,  70.   ,   2.375
PPA( 4)= POINT/ -28.5   ,  80.   ,   1.620
PPA( 5)= POINT/ -28.5   ,  90.   ,    .844
PPA( 6)= POINT/ -28.5   , 100.   ,   0.
PPA( 7)= POINT/  -8.5   ,  50.   ,   3.640  $$  M=-.0535
PPA( 8)= POINT/  -8.5   ,  56.3  ,   3.248
PPA( 9)= POINT/  -8.5   ,  62.6  ,   2.660
PPA(10)= POINT/  -8.5   ,  68.9  ,   1.908
PPA(11)= POINT/  -8.5   ,  75.2  ,   1.008
PPA(12)= POINT/  -8.5   ,  81.5  ,   0.
$$
PRINT/0
PYLNUA= SSURF/ MESH,XYZ,  SPLINE,  PPA(1),TANSPL,PYUV, PPA(2,THRU,6),  $
                          SPLINE,  PPA(7),TANSPL,PYUV, PPA(8,THRU,12)
PRINT/0
$$$$$ DEFINITION OF PYLNLA  $$$$$
$$
REFSYS / (IVERSZ=MATRIX/ 1,0,0,0, 0,1,0,0, 0,0,-1,0 )
$$
PYLNLA= SSURF/ MESH,XYZ,  SPLINE,  PPA(1),TANSPL,PYUV, PPA(2,THRU,6),  $
                         SPLINE,   PPA(7),TANSPL,PYUV, PPA(8,THRU,12)
REFSYS / NOMORE
PRINT/0
$$$$$ DEFINITION OF NACFOB  $$$$$
$$
RESERV/ P1,16,P2,16,P3,16,P4,16,P5,16,P6,16,P7,16,P8,16,P9,16,P10,16
$$   PATCH NO. 1
$$
P1 ( 1)= POINT/   0.     ,   0.     ,   9.00        $$ SP00
P1 ( 2)= POINT/   0.     ,   2.50   ,  11.05        $$ SP10
P1 ( 3)=VECTOR/   0.     ,   0.     ,   2.35        $$ FD00
P1 ( 4)=VECTOR/   0.     ,   4.9628,    1.7370      $$ FD1C
P1 ( 5)= POINT/   9.     ,   0.     ,   0.          $$ SP01
P1 ( 6)= POINT/  11.05   ,   2.50   ,   0.          $$ SP11
P1 ( 7)=VECTOR/   2.35   ,   0.     ,   0.          $$ FD01
P1 ( 8)=VECTOR/   1.7370,    4.9628,    0.          $$ FD11
P1 ( 9)=VECTOR/  14.9117,    0.     ,   0.          $$ SD01
P1 (10)=VECTOR/  18.3082,    0.     ,   0.          $$ SD10
P1 (11)=VECTOR/   3.8935,    0.     ,   0.          $$ TW00
P1 (12)=VECTOR/   2.8778,    0.     ,   0.          $$ TW10
P1 (13)=VECTOR/   0.     ,   0.     , -14.9117      $$ SD01
P1 (14)=VECTOR/   0.     ,   0.     , -18.3082      $$ SD11
```

```
P1 (15)=VECTOR/   0.     ,   0.     ,  -3.8935        $$ TW01
P1 (16)=VECTOR/   0.     ,   0.     ,  -2.8778        $$ TW11
$$
$$   PATCH NO.2
$$
P2 ( 1)= POINT/   0.     ,   2.50   ,  11.05          $$ SP00
P2 ( 2)= POINT/   0.     ,   8.75   ,  12.25          $$ SP10
P2 ( 3)=VECTOR/   0.     ,   4.4833,    1.5692        $$ FD00
P2 ( 4)=VECTOR/   0.     ,   7.9886,     .8148        $$ FD10
P2 ( 5)= POINT/  11.05   ,   2.50   ,   0.            $$ SP01
P2 ( 6)= POINT/  12.25   ,   8.75   ,   0.            $$ SP11
P2 ( 7)=VECTOR/   1.5692,    4.4833,    0.            $$ FD01
P2 ( 8)=VECTOR/    .8148,    7.9886,    0.            $$ FD11
P2 ( 9)=VECTOR/  18.3082,    0.     ,   0.            $$ SD00
P2 (10)=VECTOR/  20.2965,    0.     ,   0.            $$ SD10
P2 (11)=VECTOR/   2.5998,    0.     ,   0.            $$ TW00
P2 (12)=VECTOR/   1.350 ,    0.     ,   0.            $$ TW10
P2 (13)=VECTOR/   0.     ,   0.     , -18.3082        $$ SD01
P2 (14)=VECTOR/   0.     ,   0.     , -20.2965        $$ SD11
P2 (15)=VECTOR/   0.     ,   0.     ,  -2.5998        $$ TW01
P2 (16)=VECTOR/   0.     ,   0.     ,  -1.350         $$ TW11
$$
$$   PATCH NO. 3
$$
P3 ( 1)= POINT/   0.     ,   8.75   ,  12.25          $$ SP00
P3 ( 2)= POINT/   0.     ,  16.00   ,  12.60          $$ SP10
P3 ( 3)=VECTOR/   0.     ,   6.7741,     .6910        $$ FD00
P3 ( 4)=VECTOR/   0.     ,   7.4300,    0.            $$ FD10
P3 ( 5)= POINT/  12.25   ,   8.75   ,   0.            $$ SP01
P3 ( 6)= POINT/  12.60   ,  16.00   ,   0.            $$ SP11
P3 ( 7)=VECTOR/    .6910,    6.7741,    0.            $$ FD01
P3 ( 8)=VECTOR/   0.     ,   7.430 ,    0.            $$ FD11
P3 ( 9)=VECTOR/  20.2965,    0.     ,   0.            $$ SD00
P3 (10)=VECTOR/  20.8764,    0.     ,   0.            $$ SD10
P3 (11)=VECTOR/   1.1448,    0.     ,   0.            $$ TW00
P3 (12)=VECTOR/   0.     ,   0.     ,   0.            $$ TW10
P3 (13)=VECTOR/   0.     ,   0.     , -20.2965        $$ SD01
P3 (14)=VECTOR/   0.     ,   0.     , -20.8764        $$ SD11
P3 (15)=VECTOR/  -1.1448,    0.     ,   0.            $$ TW01
P3 (16)=VECTOR/   0.     ,   0.     ,   0.            $$ TW11
$$
$$   PATCH NO. 4
$$
P4 ( 1)= POINT/   0.     ,  16.00   ,  12.60          $$ SP00
P4 ( 2)= POINT/   0.     ,  55.00   ,  12.60          $$ SP10
P4 ( 3)=VECTOR/   0.     ,  21.8798,    0.            $$ FD00
P4 ( 4)=VECTOR/   0.     ,  39.9812,    0.            $$ FD10
P4 ( 5)= POINT/  12.60   ,  16.00   ,   0.            $$ SP01
P4 ( 6)= POINT/  12.60   ,  55.00   ,   0.            $$ SP11
P4 ( 7)=VECTOR/   0.     ,  21.8798,    0.            $$ FD01
P4 ( 8)=VECTOR/   0.     ,  39.9812,    0.            $$ FD11
P4 ( 9)=VECTOR/  20.8764,    0.     ,   0.            $$ SD00
P4 (10)=VECTOR/  20.8764,    0.     ,   0.            $$ SD10
P4 (11)=VECTOR/   0.     ,   0.     ,   0.            $$ TW00
P4 (12)=VECTOR/   0.     ,   0.     ,   0.            $$ TW10
```

```
P4 (13)=VECTOR/   0.      ,   .0     , -20.8764      $$ SD01
P4 (14)=VECTOR/   0.      ,   .0     , -20.8764      $$ SD11
P4 (15)=VECTOR/   0.      ,   .0     ,   0.          $$ TW01
P4 (16)=VECTOR/   0.      ,   .0     ,   0.          $$ TW11
$$
$$   PATCH NO. 5
$$
P5 ( 1)= POINT/   0.      , 55.00    ,  12.60        $$ SP00
P5 ( 2)= POINT/   0.      , 67.50    ,  11.65        $$ SP10
P5 ( 3)=VECTOR/   0.      , 12.40    ,   0.          $$ FD00
P5 ( 4)=VECTOR/   0.      , 12.60    ,  -1.90        $$ FD10
P5 ( 5)= POINT/  12.60    , 55.00    ,   0.          $$ SP01
P5 ( 6)= POINT/  11.65    , 67.50    ,   0.          $$ SP11
P5 ( 7)=VECTOR/   0.      , 12.40    ,   0.          $$ FD01
P5 ( 8)=VECTOR/  -1.90    , 12.60    ,   0.          $$ FD11
P5 ( 9)=VECTOR/  20.8764,   0.       ,   0.          $$ SD00
P5 (10)=VECTOR/  19.3024,   0.       ,   0.          $$ SD10
P5 (11)=VECTOR/   0.      ,  0.      ,   0.          $$ TW00
P5 (12)=VECTOR/  -3.1479,   0.       ,   0.          $$ TW10
P5 (13)=VECTOR/   0.      ,  0.      , -20.8764      $$ SD01
P5 (14)=VECTOR/   0.      ,  0.      , -19.3024      $$ SD11
P5 (15)=VECTOR/   0.      ,  0.      ,   0.          $$ TW01
P5 (16)=VECTOR/   0.      ,  0.      ,   3.1479      $$ TW11
$$
$$   PATCH NO. 6
$$
P6 ( 1)= POINT/   9.00    ,  0.      ,   0.          $$ SP00
P6 ( 2)= POINT/  11.05    ,  2.50    ,   0.          $$ SP10
P6 ( 3)=VECTOR/   2.35    ,  0.      ,   0.          $$ FD00
P6 ( 4)=VECTOR/   1.7370,   4.9628,      0.          $$ FD10
P6 ( 5)= POINT/   0.      ,  0.      ,  -9.0         $$ SP01
P6 ( 6)= POINT/   0.      ,  2.50    , -11.68        $$ SP11
P6 ( 7)=VECTOR/   0.      ,  0.      ,  -2.75        $$ FD01
P6 ( 8)=VECTOR/   0.      ,  4.9839, -2.6041         $$ FD11
P6 ( 9)=VECTOR/   0.      ,  0.      , -14.9117      $$ SD00
P6 (10)=VECTOR/   0.      ,  0.      , -19.8560      $$ SD10
P6 (11)=VECTOR/   0.      ,  0.      ,  -3.8935      $$ TW00
P6 (12)=VECTOR/   0.      ,  0.      ,  -5.9343      $$ TW10
P6 (13)=VECTOR/ -14.9117,   0.       ,   0.          $$ SD01
P6 (14)=VECTOR/ -18.7850,   0.       ,   0.          $$ SD11
P6 (15)=VECTOR/  -4.5562,   0.       ,   0.          $$ TW01
P6 (16)=VECTOR/  -4.3972,   0.       ,   0.          $$ TW11
$$
$$   PATCH NO. 7
$$
P7 ( 1)= POINT/  11.05    ,  2.50    ,   0.          $$ SP00
P7 ( 2)= POINT/  12.25    ,  8.75    ,   0.          $$ SP10
P7 ( 3)=VECTOR/   1.5692,   4.4833,      0.          $$ FD00
P7 ( 4)=VECTOR/    .8148,   7.9886,      0.          $$ FD10
P7 ( 5)= POINT/   0.      ,  2.50    , -11.68        $$ SP01
P7 ( 6)= POINT/   0.      ,  8.75    , -13.79        $$ SP11
P7 ( 7)=VECTOR/   0.      ,  4.5024,   -2.3525       $$ FD01
P7 ( 8)=VECTOR/   0.      ,  7.9701,   -1.8411       $$ FD11
P7 ( 9)=VECTOR/   0.      ,  0.      , -19.8560      $$ SD00
P7 (10)=VECTOR/   0.      ,  0.      , -26.0355      $$ SD10
```

```
P7 (11)=VECTOR/   0.      ,   0.      ,  -5.3609      $$ TW00
P7 (12)=VECTOR/   0.      ,   0.      ,  -6.3067      $$ TW10
P7 (13)=VECTOR/ -18.7850,   0.      ,   0.          $$ SD01
P7 (14)=VECTOR/ -23.1280,   0.      ,   0.          $$ SD11
P7 (15)=VECTOR/  -3.9724,   0.      ,   0.          $$ TW01
P7 (16)=VECTOR/  -4.0157,   0.      ,   0.          $$ TW11
$$
$$   PATCH NO. 8
$$
P8 ( 1)= POINT/  12.25    ,   8.75   ,   0.          $$ SP00
P8 ( 2)= POINT/  12.60    ,  16.00   ,   0.          $$ SP10
P8 ( 3)=VECTOR/    .6910,   6.7741,   0.          $$ FD00
P8 ( 4)=VECTOR/   0.      ,   7.43   ,   0.          $$ FD10
P8 ( 5)= POINT/   0.      ,   8.75   , -13.79        $$ SP01
P8 ( 6)= POINT/   0.      ,  16.     , -15.1         $$ SP11
P8 ( 7)=VECTOR/   0.      ,   6.7584,  -1.5612      $$ FD01
P8 ( 8)=VECTOR/   0.      ,   8.0751,  -1.1023      $$ FD11
P8 ( 9)=VECTOR/   0.      ,   0.      , -26.0355     $$ SD00
P8 (10)=VECTOR/   0.      ,   0.      , -30.2        $$ SD10
P8 (11)=VECTOR/   0.      ,   0.      ,  -5.3479     $$ TW00
P8 (12)=VECTOR/   0.      ,   0.      ,  -2.8888     $$ TW10
P8 (13)=VECTOR/ -23.1280,   0.      ,   0.          $$ SD01
P8 (14)=VECTOR/ -25.2000,   0.      ,   0.          $$ SD11
P8 (15)=VECTOR/  -3.4052,   0.      ,   0.          $$ TW01
P8 (16)=VECTOR/   -.7733,   0.      ,   0.          $$ TW11
$$
$$   PATCH NO. 9
$$
P9 ( 1)= POINT/  12.60    ,  16.00   ,   0.          $$ SP00
P9 ( 2)= POINT/  12.60    ,  55.00   ,   0.          $$ SP10
P9 ( 3)=VECTOR/   0.      ,  21.8798,   0.          $$ FD00
P9 ( 4)=VECTOR/   0.      ,  39.9812,   0.          $$ FD10
P9 ( 5)= POINT/   0.      ,  16.00   , -15.1         $$ SP01
P9 ( 6)= POINT/   0.      ,  55.00   , -13.35        $$ SP11
P9 ( 7)=VECTOR/   0.      ,  23.7795,  -3.2459      $$ FD01
P9 ( 8)=VECTOR/   0.      ,  54.7655,   6.9120      $$ FD11
P9 ( 9)=VECTOR/   0.      ,   0.      , -30.20       $$ SD00
P9 (10)=VECTOR/   0.      ,   0.      , -23.2824     $$ SD10
P9 (11)=VECTOR/   0.      ,   0.      ,  -8.5068     $$ TW00
P9 (12)=VECTOR/   0.      ,   0.      ,  13.8751     $$ TW10
P9 (13)=VECTOR/ -25.2     ,   0.      ,   0.          $$ SD01
P9 (14)=VECTOR/ -21.9744,   0.      ,   0.          $$ SD11
P9 (15)=VECTOR/  -2.2771,   0.      ,   0.          $$ TW01
P9 (16)=VECTOR/   6.6244,   0.      ,   0.          $$ TW11
$$
$$   PATCH NO. 10
$$
P10( 1)= POINT/  12.60    ,  55.0    ,   0.          $$ SP00
P10( 2)= POINT/  11.65    ,  67.50   ,   0.          $$ SP10
P10( 3)=VECTOR/   0.      ,  12.40   ,   0.          $$ FD00
P10( 4)=VECTOR/  -1.90    ,  12.60   ,   0.          $$ FD10
P10( 5)= POINT/   0.      ,  55.0    , -13.35        $$ SP01
P10( 6)= POINT/   0.      ,  67.5    , -11.65        $$ SP11
P10( 7)=VECTOR/   0.      ,  16.9853,   2.1437      $$ FD01
P10( 8)=VECTOR/   0.      ,   7.9825,   1.2038      $$ FD11
```

```
P10( 9)=VECTOR/   0.     ,   0.    , -23.2824      $$ SD00
P10(10)=VECTOR/   0.     ,   0.    , -19.3024      $$ SD10
P10(11)=VECTOR/   0.     ,   0.    ,   4.3033      $$ TW00
P10(12)=VECTOR/   0.     ,   0.    ,   3.1481      $$ TW10
P10(13)=VECTOR/ -21.9744,   0.    ,   0.          $$ SD01
P10(14)=VECTOR/ -19.3024,   0.    ,   0.          $$ SD11
P10(15)=VECTOR/   2.0545,   0.    ,   0.          $$ TW01
P10(16)=VECTOR/   1.9944,   0.    ,   0.          $$ TW11
$$
PRINT/0
NACFOB= SSURF/ PATCH, PNTVEC, 6, 3,       PLUS,     $
        P1   (   ALL   ),                 PLUS,     $
        P2   (   ALL   ),                 PLUS,     $
        P3   (   ALL   ),                 PLUS,     $
        P4   (   ALL   ),                 PLUS,     $
        P5   (   ALL   ),                 PLUS,     $
        P6   (   ALL   ),                 PLUS,     $
        P7   (   ALL   ),                 PLUS,     $
        P8   (   ALL   ),                 PLUS,     $
        P9   (   ALL   ),                 PLUS,     $
        P10  (   ALL   )                    .
PRINT/0
$$$$$ DEFINITION OF NACFIB  $$$$$
$$
REFSYS / (IVERSX=MATRIX/ -1,0,0,0, 0,1,0,0, 0,0,1,0 )
$$
NACFIB= SSURF/ PATCH, PNTVEC, 6, 3,       PLUS,     $
        P1   (   ALL   ),                 PLUS,     $
        P2   (   ALL   ),                 PLUS,     $
        P3   (   ALL   ),                 PLUS,     $
        P4   (   ALL   ),                 PLUS,     $
        P5   (   ALL   ),                 PLUS,     $
        P6   (   ALL   ),                 PLUS,     $
        P7   (   ALL   ),                 PLUS,     $
        P8   (   ALL   ),                 PLUS,     $
        P9   (   ALL   ),                 PLUS,     $
        P10  (   ALL   )
REFSYS / NOMORE
PRINT/0
$$$$$ DEFINITION OF NNOZO AND NNOZI  $$$$$
$$
CPTF =POINT/ 0, 67.5, 11.65
CPTM =POINT/ 0, 75, 10.1
CPTA =POINT/ 0, 81.5, 8.35
CONVF=VECTOR/ 0, 12.6, -1.9
CONVA=VECTOR/ 0, 1, -.290
PRINT/0
NOZCON= SCURV/ CURSEG,CPTF  , TANSPL,CONVF , CPTM, CPTA, TANSPL, CONVA
PRINT/0
NNOZO =SSURF/ REVOLV,NOZCON,XAXIS,NOR,YV,CCLW,  0, 180
PRINT/0
NNOZI =SSURF/ REVOLV,NOZCON,XAXIS,NOR,YV, CLW,  0, 180
PRINT/0
$$$$$ DEFINITION OF INLETO AND INLRTI  $$$$$
$$
```

```
RESERV/ IP,9
IP(1) = POINT /   0.    ,    0.    ,   9.000
IP(2) = POINT /   0.    ,     .250 ,   8.400
IP(3) = POINT /   0.    ,     .750 ,   8.065
IP(4) = POINT /   0.    ,   1.500 ,   7.832
IP(5) = POINT /   0.    ,   2.300 ,   7.760
IP(6) = POINT /   0.    ,   5.000 ,   8.230
IP(7) = POINT /   0.    ,   7.750 ,   9.015
IP(8) = POINT /   0.    ,  11.750 ,   9.770
IP(9) = POINT /   0.    ,  17.250 ,  10.000
$$
PRINT/0
INLSPL = SCURV/ SPLINE, IP(1), TANSPL,ZV, IP(2),IP(3),IP(4),IP(5),        $
                        TANSPL,YV, IP(6),IP(7),IP(8),IP(9),TANSPL,YV
PRINT/0
INLETO=SSURF/ REVOLV,INLSPL,XAXIS,NOR,YV,CCLW,  0, 180
PRINT/0
INLETI=SSURF/ REVOLV,INLSPL,XAXIS,NOR,YV, CLW,  0, 180
PRINT/0
$$$$$    DEFINE  FUS  PATCH  BOUNDARIES
RESERV/ RADPLN,8, YFUS,8
LOOPST
      A=1
      B=1
10)   OBTAIN,POINT/FP(B), X1,Y1,Z1
      OBTAIN,POINT/FP(B+4),X2,Y2,Z2
      OBTAIN,POINT/FP(B+7),X3,Y3,Z3
REFSYS/FS2NS
      RADPLN(A)=PLANE/(X1,Y1,Z1),(X2,Y2,Z2),(X3,Y3,Z3)
REFSYS/NOMORE
      A=A+1
      B=B+8
      IF(8-A) 20,10,10
20)LOOPND
REFSYS/ FS2NS
YFUS( 1)=PLANE/ 0,1,0,818
YFUS( 2)=PLANE/ 0,1,0,838
YFUS( 3)=PLANE/ 0,1,0,858
YFUS( 4)=PLANE/ 0,1,0,878
YFUS( 5)=PLANE/ 0,1,0,898
YFUS( 6)=PLANE/ 0,1,0,918
YFUS( 7)=PLANE/ 0,1,0,938
YFUS( 8)=PLANE/ 0,1,0,948
REFSYS/NOMORE
$$ YN NACELLE PLANES
RESERV/ YN,9, ELPL,12, AEL,5
PL120=PLANE/ (0,0,0 ),(0,10,0),( 86.60254,0,-50)
PL240=PLANE/(0,0,0),(0,10,0),( 86.60254,0, 50)
YN( 1)=PLANE/0,1,0,0
YN( 2)=PLANE/0,1,0,2.5
YN( 3)=PLANE/0,1,0,8.75
YN( 4)=PLANE/0,1,0,16
YN( 5)=PLANE/0,1,0,55
YN( 6)=PLANE/0,1,0,67.5
YN( 7)=PLANE/0,1,0,81.5
```

```
YN( 8)=PLANE/0,1,0,17.25
YN( 9)=PLANE/0,1,0,50
$$
XNO = PLANE/ 1,0,0,0
YNO = PLANE/0,1,0,0
ZNO = PLANE/ 0,0,1,0
LECYL= CYLNDR / NOR,YV, 9
NZV= VECTOR/ 0,0,-1
$$$$$     DEFINE  PYLON  ELEMENT  PLANES
ELPL( 1)=PLANE/ PERPTO, XNO,  PPF( 1), PPF(23)
ELPL( 2)=PLANE/ PERPTO, XNO,  PPF( 2), PPF(22)
ELPL( 3)=PLANE/ PERPTO, XNO,  PPF( 3), PPF(21)
ELPL( 4)=PLANE/ PERPTO, XNO,  PPF( 4), PPF(20)
ELPL( 5)=PLANE/ PERPTO, XNO,  PPF( 5), PPF(19)
ELPL( 6)=PLANE/ PERPTO, XNO,  PPF( 6), PPF(18)
ELPL( 7)=PLANE/ PERPTO, XNO,  PPF( 7), PPF(17)
ELPL( 8)=PLANE/ PERPTO, XNO,  PPF( 8), PPF(16)
ELPL( 9)=PLANE/ PERPTO, XNO,  PPF( 9), PPF(15)
ELPL(10)=PLANE/ PERPTO, XNO,  PPF(10), PPF(14)
ELPL(11)=PLANE/ PERPTO, XNO,  PPF(11), PPF(13)
ELPL(12)=PLANE/ 0,0,1,0
AEL(1)=PLANE / PERPTO,  ZNO,  PPA( 2), PPA( 8)
AEL(2)=PLANE / PERPTO,  ZNO,  PPA( 3), PPA( 9)
AEL(3)=PLANE / PERPTO,  ZNO,  PPA( 4), PPA(10)
AEL(4)=PLANE / PERPTO,  ZNO,  PPA( 5), PPA(11)
AEL(5)=PLANE / PERPTO,  ZNO,  PPA( 6), PPA(12)
$$$$$     DRAW  YN  BOUNDARIES
TMARK/1
LOOPST
      A=2
      PENUP
      DNTCUT
1A)            GOTO/ 5,25,15, 1,0,1
      GO/ YN(A), NACFOB, XNO
      CUT,PENDWN,INDIRV/XV
      GOFWD/ YN(A), XNO
      PSIS/NACFIB
      GOFWD/ YN(A), XNO
      PENUP,DNTCUT
      A=A+1
      IF(6-A) 2A,1A,1A
2A)   LOOPND
      GOTO/ 5,70,15,1,0,1
      GO/ YN(7), NNOZO,XNO
      CUT,PENDWN,INDIRV/XV
      GOFWD/ YN(7),XNO
      PSIS/NNOZI
      GOFWD/YN(7),XNO
      PENUP,DNTCUT
      GOTO/ 5,15,15,1,0,1
      GO/ YN(8),INLETO,XNO
      CUT,PENDWN,INDIRV/XV
      GOFWD/YN(8),XNO
      PSIS/INLETI
      GOFWD/YN(8),XNO
```

```
       PENUP,DNTCUT
       GOTO/ 9, 0, 0, 0,-1,0
       PSIS/ YNO
       CUT,PENDWN,INDIRV/ZV
       GOFWD/LECYL,2,INTOF,ZNO
       PENUP,DNTCUT
$$$$$     DRAW  XN=0  AND  ZN=0
TMARK/2
RADCUT=MACRO/DS,1STPS=NNOZO,2NDPS=NACFOB,3RDPS=INLETO
       CUT,PENDWN,INDIRV/0,-1,0
       PSIS/1STPS
       GOFWD/DS,YN(6)
       PSIS/2NDPS
       GOFWD/DS,LECYL
       PSIS/3RDPS
       GOFWD/DS,YN(8)
       PENUP,DNTCUT
       TERMAC
       GOTO/0,81.5,8.35,ZV
       CALL/RADCUT,DS=XNO
       GOTO/8.35,81.5,0,XV
       CALL/RADCUT,DS=ZNO
       GOTO/0,81.5,-8.35, 0,0,-1
       CALL/RADCUT,DS=XNO
       GOTO/-8.35,81.5,0,-1,0,0
       CALL/RADCUT,DS=ZNO,1STPS=NNOZI, 2NDPS=NACFIB, 3RDPS=INLETI
$$$$$     DRAW  PYLON  INTERSECTION W/FUS  AND  NACELLE
TMARK/3
PYINTR= MACRO/ DS1,DS2,     PS2,STRV
       PENUP, DNTCUT, GOTO/-18,5,1,0,-1,0
       GO/ ZNO,PYLONF, DS1
       CUT,PENDWN,INDIRV/ STRV
       GOFWD/DS1, YN(9)
       PSIS/PS2
       GOFWD/DS1, YN(6)
       GOFWD/DS2, ZNO
       PENUP,DNTCUT
       TERMAC
       CALL/ PYINTR,DS1=FUS, DS2=FUS, PS2=PYLNUA , STRV= ZV
       CALL/ PYINTR,DS1=FUS, DS2=FUS, PS2=PYLNLA , STRV= NZV
       CALL/ PYINTR,DS1=NACFIB,DS2=NNOZI , PS2=PYLNUA , STRV=ZV
       CALL/ PYINTR,DS1=NACFIB,DS2=NNOZI , PS2=PYLNLA , STRV=NZV
$$$$$     DRAW  PYLON  ELEMENT  LINES  L.E.  THRU  T.E.
TMARK/4
LOOPST
       A=1
       PENUP,DNTCUT
1B)    GOTO/-15,13,5,0,-1,1
       GO/ELPL(A),PYLONF,NACFIB
       CUT,PENDWN,INDIRV/-1,0,0
       GOFWD/ELPL(A),FUS
       PENUP,DNTCUT
       A=A+1
       IF(12-A) 2B,1B,1B
2B)    A=1
```

```
3B)   GOTO/-15,13,-5,0,-1,-1
      GO/ELPL(A),PYLONF,NACFIB
      CUT,PENDWN,INDIRV/-1,0,0
      GOFWD/ELPL(A),FUS
      PENUP,DNTCUT
      A=A+1
      IF(11-A) 4B,3B,3B
4B)   A=1
5B)   GOTO/-15,60,5,0,0,1
      GO/AEL(A),PYLNUA,NACFIB
      CUT,PENDWN,INDIRV/-1,0,0
      GOFWD/AEL(A),FUS
      PENUP,DNTCUT,GODLTA/ 1,1,-1
      GO/AEL(A),PYLNLA,FUS
      CUT,PENDWN,INDIRV/XV
      GOFWD/AEL(A),NACFIB
      PENUP,DNTCUT
      A=A+1
      IF(2 -A)6B,5B,5B
6B)   GOTO/-15,78,5,0,0,1
      GO/AEL(A),PYLNUA,NNOZI
      CUT,PENDWN,INDIRV/-1,0,0
      GOFWD/AEL(A),FUS
      PENUP,DNTCUT,GODLTA/ 1,1,-1
      GO/AEL(A),PYLNLA,FUS
      CUT,PENDWN,INDIRV/XV
      GOFWD/AEL(A),NNOZI
      PENUP,DNTCUT
      A=A+1
      IF(5 -A)7B,6B,6B
7B)   LOOPND
$$$$$    DRAW  FUS  PATCH  BOUNDARIES
TMARK/5
      PSIS/FUS
LOOPST
      A=1
      B=12
       PENUP,DNTCUT
1D)   GOTO/-20,50,15,1,0,0
      GO/YFUS(A),FUS,RADPLN(1)
      CUT,PENDWN,INDIRV/0,0,-1
      GOFWD/YFUS(A),RADPLN(8)
      PENUP,DNTCUT
      A=A+1
      B=B+1
      IF(8 -A)2D,1D,1D
2D)   A=1
      B=12
3D)   GOTO/-20,0,0,1,0,0
      GO/YFUS(1),FUS,RADPLN(A)
      CUT,PENDWN,INDIRV/0,1,0
      GOFWD/RADPLN(A),YFUS(8)
      PENUP,DNTCUT
      A=A+1
      B=B+11
```

```
      IF(8-A)4D,3D,3D
4D)LOOPND
FINI
```

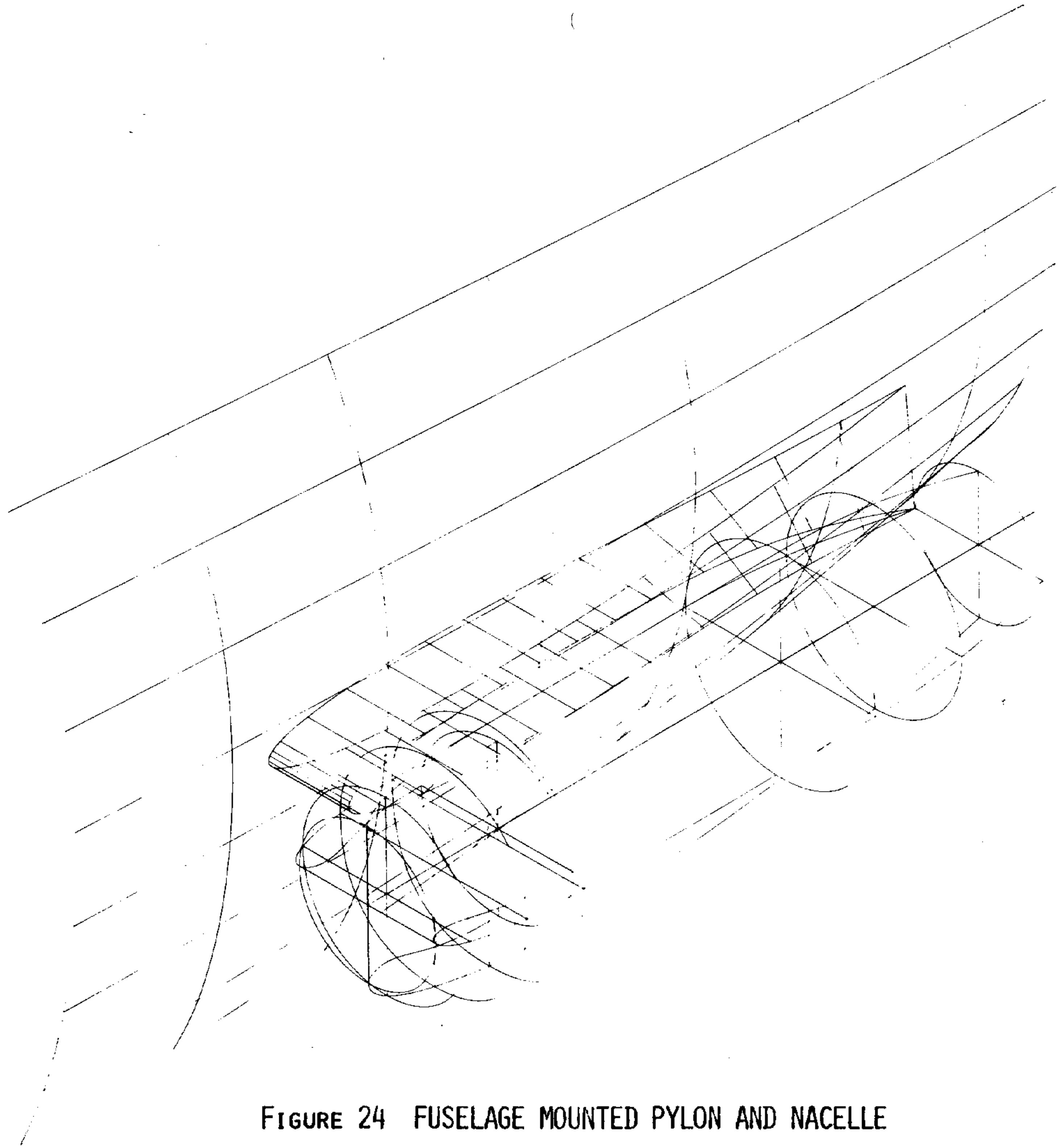


Figure 24 FUSELAGE MOUNTED PYLON AND NACELLE

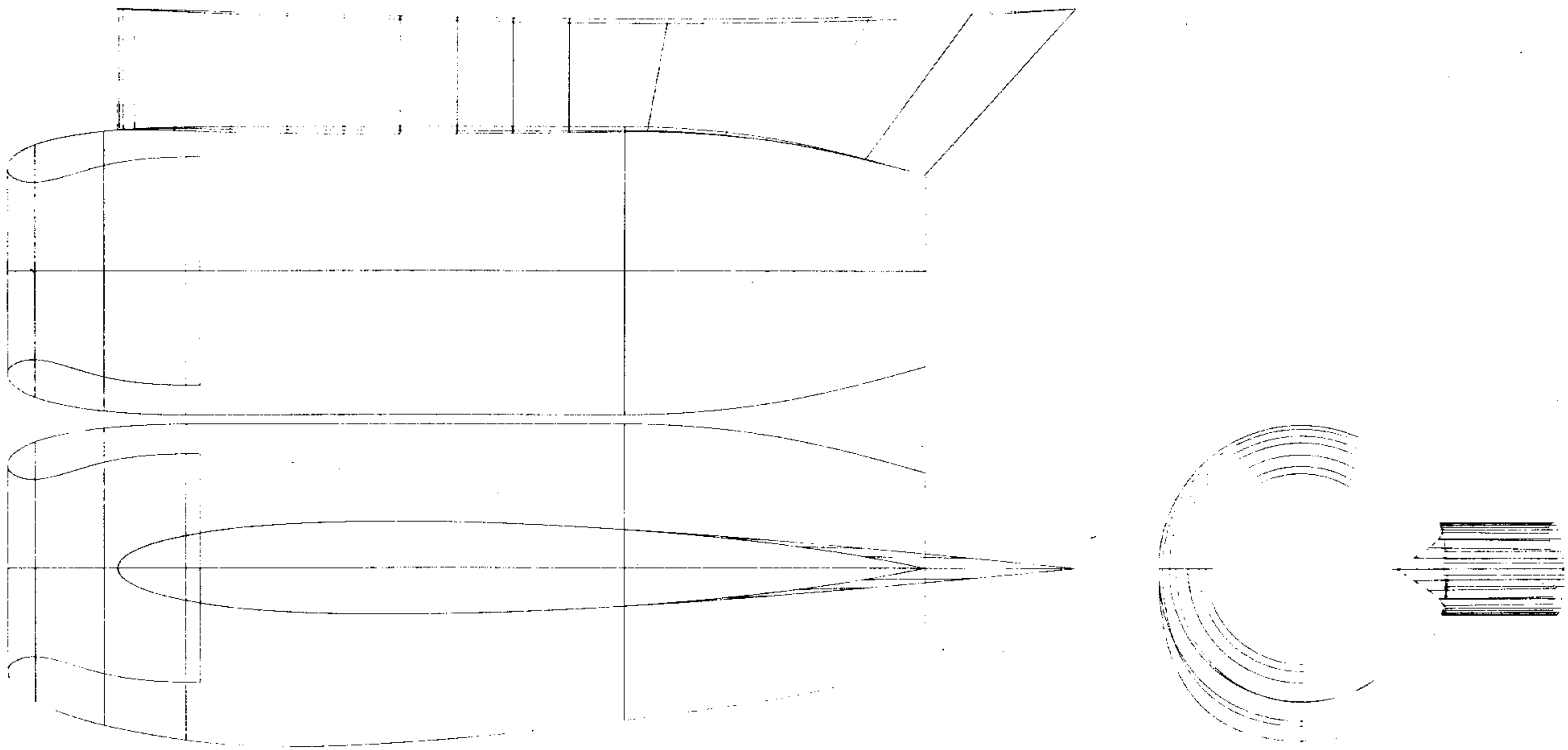


FIGURE 25 FUSELAGE MOUNTED PYLON AND NACELLE

3.2 Machine Angle Between Fus/Pylon

```
PARTNO  SSIP/SSX3A -TEST 2- MACHINE ANGLE BETWEEN FUS/PYLON      (CHUBB)
$$    PURPOSE - 1. TEST TLAXIS/1
$$                 TEST TLAXIS/NORMPS
$$                 TEST THICK/ PS,DS
$$                 TEST REFSYS
$$                 TEST FLAT END MILL CUTTER
$$                 TEST BALL END CUTTER
$$                 TEST GO/ DS,PS,CS
$$                 TEST TLLFT - TLRGT
$$
$$--------------------------------------------------------------------------
NOPUST
MULTAX
CLPRNT
CANON/ON
SSON   =SSURF/DISPLY,ON
PT1 = POINT/ -21.659, 50.000, 3.640
PT2 = POINT/ -21.852, 73.240, 1.988
ZPS = PLANE/ PT1, PERPTO, (VECTOR/1,0,1)
YPS = PLANE/ PT1, PT2, PERPTO, ZPS
XPS = PLANE/ PT1, PERPTO, ZPS, YPS
PS 2 NS = MATRIX / (PLANE/PARLEL,XPS,YLARGE,0), $
                   (PLANE/PARLEL,YPS,ZLARGE,0), $
                   (PLANE/PARLEL,ZPS,ZLARGE,0)
NS 2 PS = MATRIX / INVERS, PS 2 NS
PARK   = POINT/ 10,10,10
STRTPT= POINT/ 1,0,4
ENDPT  = POINT/ 21,0,4
CK1    = PLANE/ 1,0,0,2
CK2    = PLANE/ 1,0,0,20
$$$$$     MACHINING  MACRO     $$$$$
CUTAN =MACRO/ PS,DS,THKPS,THKDS,CUTMOD,TLSIDE
       $$ CUTMOD=TA1 OR NRMPS
       $$ TLSIDE= TLL OR TLR
       JUMPTO / CUTMOD
TA1)   TLAXIS / 1
       JUMPTO / THK
NRMPS)TLAXIS / NORMPS
THK)   THICK / THKPS, THKDS,0
       GOTO / STRTPT, 0,0,1
       GO/DS,PS,CK1
       JUMPTO/TLSIDE
TLR)   TLRGT,INDIRV/1,0,0
       JUMPTO/ GF
TLL)   TLLFT,INDIRV/1,0,0
GF)    GOFWD/DS,CK2
       GOTO/ENDPT,0,0,1
       GOTO/PARK
TERMAC
$$$$$ DEFINITION OF PYLNUA  $$$$$
$$
PYUV=VECTOR / 0,1,-.0535
RESERV/ PPA,12
PPA( 1)= POINT/ -28.5    ,  50.     ,    3.640  $$  M=-.0535
PPA( 2)= POINT/ -28.5    ,  60.     ,    3.050
```

```
PPA( 3)= POINT/ -28.5    ,  70.    ,    2.375
PPA( 4)= POINT/ -28.5    ,  80.    ,    1.620
PPA( 5)= POINT/ -28.5    ,  90.    ,     .844
PPA( 6)= POINT/ -28.5    , 100.    ,    0.
PPA( 7)= POINT/  -8.5    ,  50.    ,    3.640   $$   M=-.0535
PPA( 8)= POINT/  -8.5    ,  56.3   ,    3.248
PPA( 9)= POINT/  -8.5    ,  62.6   ,    2.660
PPA(10)= POINT/  -8.5    ,  68.9   ,    1.908
PPA(11)= POINT/  -8.5    ,  75.2   ,    1.008
PPA(12)= POINT/  -8.5    ,  81.5   ,    0.
$$
PRINT/0
REFSYS/ NS 2 PS
PYLNUA= SSURF/ MESH,XYZ,  SPLINE,  PPA(1),TANSPL,PYUV, PPA(2,THRU,6),  $
                          SPLINE,  PPA(7),TANSPL,PYUV, PPA(8,THRU,12)
REFSYS/ NOMORE
$$$$$ DEFINITION OF FUS     $$$$$
$$
RESERV/FP,64
SYN/ POINT,PT
$$
FP(  1)=PT/  52.6099, 818,  63.0258
FP(  2)=PT/  52.2075, 838,  62.7441
FP(  3)=PT/  51.6382, 858,  62.3455
FP(  4)=PT/  50.8079, 878,  61.7640
FP(  5)=PT/  49.6375, 898,  60.9446
FP(  6)=PT/  48.0820, 918,  59.8554
FP(  7)=PT/  46.1009, 938,  58.4682
FP(  8)=PT/  44.9228, 948,  57.6433
FP(  9)=PT/  57.7064, 818,  53.0969
FP( 10)=PT/  57.1455, 838,  52.8353
FP( 11)=PT/  56.4104, 858,  52.4926
FP( 12)=PT/  55.3715, 878,  52.0081
FP( 13)=PT/  53.9226, 898,  51.3325
FP( 14)=PT/  52.0075, 918,  50.4395
FP( 15)=PT/  49.5763, 938,  49.3058
FP( 16)=PT/  48.1292, 948,  48.6310
FP( 17)=PT/  60.9279, 818,  42.5136
FP( 18)=PT/  60.1991, 838,  42.3183
FP( 19)=PT/  59.2968, 858,  42.0765
FP( 20)=PT/  58.0545, 878,  41.7436
FP( 21)=PT/  56.3382, 898,  41.2838
FP( 22)=PT/  54.0822, 918,  40.6793
FP( 23)=PT/  51.2287, 938,  39.9147
FP( 24)=PT/  49.5307, 948,  39.4597
FP( 25)=PT/  62.2244, 818,  31.6319
FP( 26)=PT/  61.3397, 838,  31.5545
FP( 27)=PT/  60.2886, 858,  31.4625
FP( 28)=PT/  58.8695, 878,  31.3384
FP( 29)=PT/  56.9242, 898,  31.1682
FP( 30)=PT/  54.3755, 918,  30.9452
FP( 31)=PT/  51.1537, 938,  30.6633
FP( 32)=PT/  49.2394, 948,  30.4958
FP( 33)=PT/  61.9176, 818,  20.7709
FP( 34)=PT/  60.9249, 838,  20.8577
```

```
FP( 35)=PT/  59.7296, 858,  20.9623
FP( 36)=PT/  58.1200, 878,  21.1031
FP( 37)=PT/  55.9294, 898,  21.2948
FP( 38)=PT/  53.0792, 918,  21.5441
FP( 39)=PT/  49.5168, 938,  21.8558
FP( 40)=PT/  47.4346, 948,  22.0380
FP( 41)=PT/  60.2531, 818,  10.0432
FP( 42)=PT/  59.1625, 838,  10.3354
FP( 43)=PT/  57.8009, 858,  10.7002
FP( 44)=PT/  55.9368, 878,  11.1997
FP( 45)=PT/  53.4419, 898,  11.8682
FP( 46)=PT/  50.3017, 918,  12.7096
FP( 47)=PT/  46.5039, 938,  13.7273
FP( 48)=PT/  44.3338, 948,  14.3087
FP( 49)=PT/  57.2002, 818,   -.4849
FP( 50)=PT/  55.9726, 838,    .0875
FP( 51)=PT/  54.3162, 858,    .8598
FP( 52)=PT/  52.1634, 878,   1.8637
FP( 53)=PT/  49.4540, 898,   3.1272
FP( 54)=PT/  46.1781, 918,   4.6547
FP( 55)=PT/  42.3266, 938,   6.4507
FP( 56)=PT/  40.1671, 948,   7.4577
FP( 57)=PT/  53.5726, 818,  -8.6154
FP( 58)=PT/  52.1460, 838,  -7.6760
FP( 59)=PT/  50.3198, 858,  -6.4901
FP( 60)=PT/  48.0504, 878,  -5.0163
FP( 61)=PT/  45.2939, 898,  -3.2262
FP( 62)=PT/  42.0435, 918,  -1.1154
FP( 63)=PT/  38.2921, 938,   1.3207
FP( 64)=PT/  36.2156, 948,   2.6692
$$
NOFS =POINT/ 85, 835, 28
NPT2 =POINT/ 75, 935, 28
ZNO  =PLANE/ 0,0,1,28
XNO  =PLANE/ PERPTO,ZNO, NOFS, NPT2
YNO  =PLANE/ NOFS, PERPTO, XNO, ZNO
NS2FS= MATRIX/ (PLANE/PARLEL,XNO,XLARGE,0),(PLANE/PARLEL,YNO,YLARGE,0),$
                (PLANE/PARLEL,ZNO,ZLARGE,0)
FS2NS= MATRIX/ INVERS,NS2FS
PRINT/0
REFSYS/(FUS 2 PS=MATRIX/ NS 2 PS,  FS 2 NS )
FUS=SSURF/MESH,XYZ,    $ FUSELAGE SURFACE
             SPLINE,FP(  1,THRU,  8),$
             SPLINE,FP(  9,THRU, 16),$
             SPLINE,FP( 17,THRU, 24),$
             SPLINE,FP( 25,THRU, 32),$
             SPLINE,FP( 33,THRU, 40),$
             SPLINE,FP( 41,THRU, 48),$
             SPLINE,FP( 49,THRU, 56),$
             SPLINE,FP( 57,THRU, 64)
REFSYS / NOMORE
PRINT/0
$$$$$    CALL  MACRO    $$$$$
      CUTTER/ 2
PPRINT PASS NO. 1
```

```
      CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS=-.10, THKDS=2, $
                   CUTMOD=NRMPS, TLSIDE=TLR
PPRINT PASS NO. 2
      CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS=-.10, THKDS=1, $
                   CUTMOD=NRMPS, TLSIDE=TLR
PPRINT PASS NO. 3
      CALL/CUTAN , PS=FUS, DS=PYLNUA, THKPS=-.20, THKDS=2, $
                   CUTMOD=NRMPS, TLSIDE=TLL
PPRINT PASS NO. 4
      CALL/CUTAN , PS=FUS, DS=PYLNUA, THKPS=-.20, THKDS=1, $
                   CUTMOD=NRMPS, TLSIDE=TLL
      CUTTER/1,.5
PPRINT CLEAN OUT CORNER WITH BALL ENDMILL
LOOPST
      A= .9
C1)   CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS=-.10, THKDS=A, $
                   CUTMOD=TA1,   TLSIDE=TLR
      A=A-.1
      IF(-.2-A) C1,C1,C2
C2)   A=0
C3)   CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS= A,  THKDS=-.2,$
                   CUTMOD=TA1,   TLSIDE=TLR
      A=A+.1
      IF(.9-A) C4,C4,C3
C4)LOOPND
FINI
```

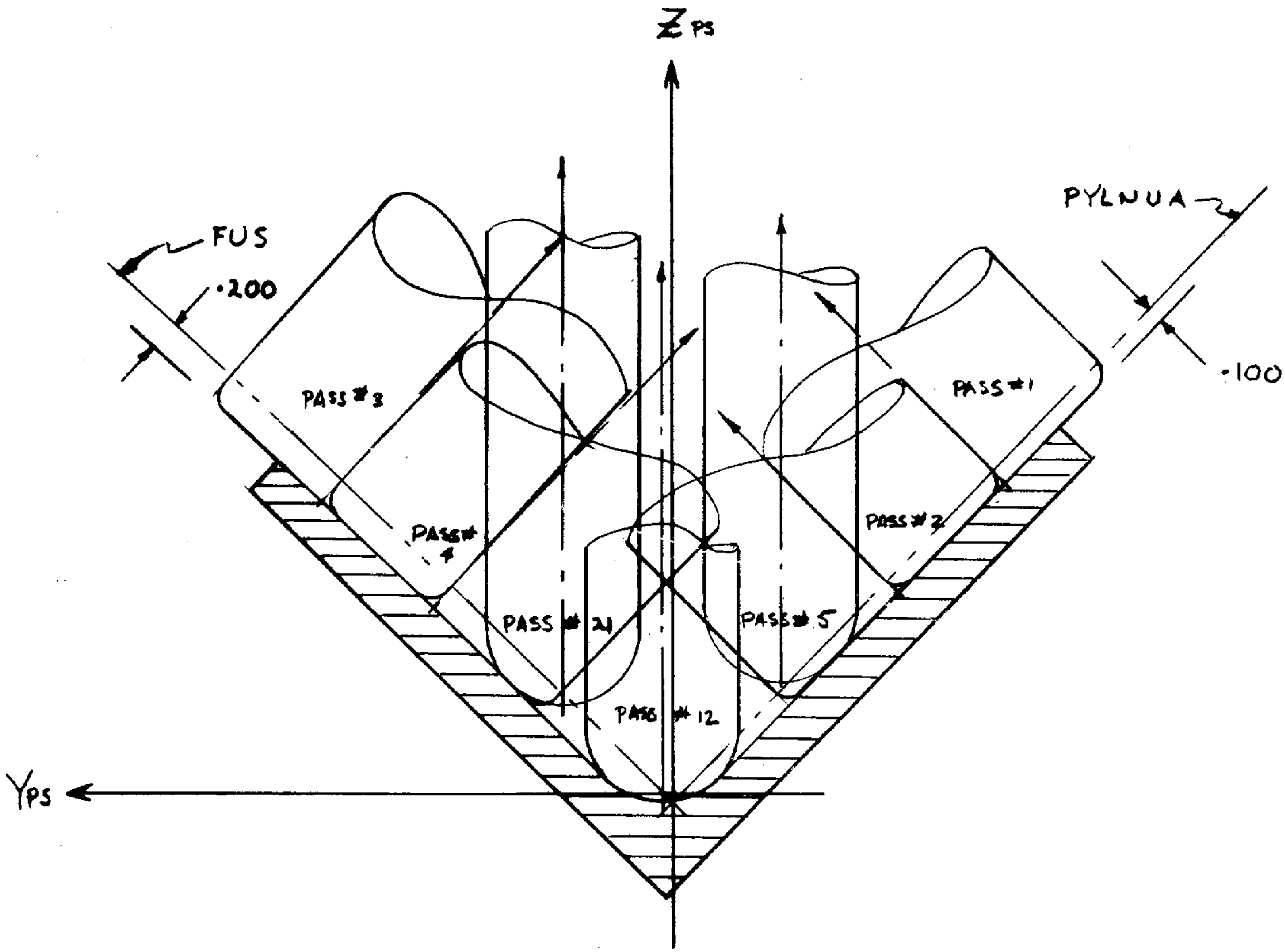


FIGURE 26 MACHINE ANGLE BETWEEN FUS/PYLON

3.2a Same as 3.2 (Alternate REFSYS Method)

```
PARTNO  SSIP/SSX3A -TEST 2A- MACHINE ANGLE BETWEEN FUS/PYLON     (CHUBB)
$$    PURPOSE - 1. SAME AS TEST 2 EXCEPT FOR REFSYS TECHNIQUE
$$              2. THIS PROGRAM SHOULD RUN ON APT 4 (SSX3A) WITH ONLY
$$                 MINOR MODIFICATIONS.
$$
$$              NOTE - THIS WAS NEVER ACCOMPLISHED
$$
$$-----------------------------------------------------------------------
NOPOST
MULTAX
CLPRNT
   SSON= SSURF / DISPLY , ON   $$ APT 3  (SSIP)
$$ PRINT/TABPRT , ON           $$ APT 4  (SSX3A)
PT1 = POINT/ -21.659, 50.000, 3.640
PT2 = POINT/ -21.852, 73.240, 1.988
ZPS = PLANE/ PT1, PERPTO, (VECTOR/1,0,1)
YPS = PLANE/ PT1, PT2, PERPTO, ZPS
XPS = PLANE/ PT1, PERPTO, ZPS, YPS
PS 2 NS = MATRIX / (PLANE/PARLEL,XPS,YLARGE,0), $
                   (PLANE/PARLEL,YPS,ZLARGE,0), $
                   (PLANE/PARLEL,ZPS,ZLARGE,0)
NS 2 PS = MATRIX / INVERS, PS 2 NS
PARK   = POINT/ 10,10,10
STRTPT= POINT/ 1,0,4
ENDPT  = POINT/ 21,0,4
CK1    = PLANE/ 1,0,0,2
CK2    = PLANE/ 1,0,0,20
ZV=VECTOR/0,0,1
$$$$$   MACHINING  MACRO      $$$$$
CUTAN =MACRO/ PS,DS,THKPS,THKDS,CUTMOD,TLSIDE
      $$ CUTMOD=TA1 OR NRMPS
      $$ TLSIDE= TLL OR TLR
      JUMPTO / CUTMOD
TA1)  TLAXIS / 1
      JUMPTO / THK
NRMPS)TLAXIS / NORMPS
THK)  THICK / THKPS, THKDS,0
      GOTO/ STRTPT,ZV
      GO/DS,PS,CK1
      JUMPTO/TLSIDE
TLR)  TLRGT,INDIRV/1,0,0
      JUMPTO/ GF
TLL)  TLLFT,INDIRV/1,0,0
GF)   GOFWD/DS,CK2
      GOTO/ ENDPT,ZV
      GOTO/PARK
TERMAC
$$$$$ DEFINITION OF PYLNUA  $$$$$
$$
RESERV/ PPA,12
$$   REFSYS POINTS TO PART SYSTEM
REFSYS/ NS 2 PS
PYUV=VECTOR / 0,1,-.0535
PPA( 1)= POINT/ -28.5    ,  50.    ,    3.640  $$  M=-.0535
PPA( 2)= POINT/ -28.5    ,  60.    ,    3.050
```

```
PPA( 3)= POINT/ -28.5    ,  70.     ,    2.375
PPA( 4)= POINT/ -28.5    ,  80.     ,    1.620
PPA( 5)= POINT/ -28.5    ,  90.     ,     .844
PPA( 6)= POINT/ -28.5    , 100.     ,    0.
PPA( 7)= POINT/  -8.5    ,  50.     ,    3.640   $$  M=-.0535
PPA( 8)= POINT/  -8.5    ,  56.3    ,    3.248
PPA( 9)= POINT/  -8.5    ,  62.6    ,    2.660
PPA(10)= POINT/  -8.5    ,  68.9    ,    1.908
PPA(11)= POINT/  -8.5    ,  75.2    ,    1.008
PPA(12)= POINT/  -8.5    ,  81.5    ,    0.
REFSYS/ NOMORE
$$
PYLNUA= SSURF/ MESH,XYZ,  SPLINE,  PPA(1),TANSPL,PYUV, PPA(2,THRU,6),  $
                          SPLINE,  PPA(7),TANSPL,PYUV, PPA(8,THRU,12)
$$$$$ DEFINITION OF FUS      $$$$$
$$
NOFS =POINT/ 85, 835, 28
NPT2 =POINT/ 75, 935, 28
ZNO  =PLANE/ 0,0,1,28
XNO  =PLANE/ PERPTO,ZNO, NOFS, NPT2
YNO  =PLANE/ NOFS, PERPTO, XNO, ZNO
NS2FS= MATRIX/ (PLANE/PARLEL,XNO,XLARGE,0),(PLANE/PARLEL,YNO,YLARGE,0),$
               (PLANE/PARLEL,ZNO,ZLARGE,0)
FS2NS= MATRIX/ INVERS,NS2FS
RESERV/FP,64
SYN/PT,POINT
$$   REFSYS POINTS TO PART SYSTEM
   REFSYS/(FUS 2 PS=MATRIX/ NS 2 PS,  FS 2 NS )  $$ APT 3  (SSIP)
$$ REFSYS/(FUS 2 PS=MATRIX/ FS 2 NS,  NS 2 PS )  $$ APT 4  (SSX3A)
$$
FP(  1)=PT/  52.6099, 818,  63.0258
FP(  2)=PT/  52.2075, 838,  62.7441
FP(  3)=PT/  51.6382, 858,  62.3455
FP(  4)=PT/  50.8079, 878,  61.7640
FP(  5)=PT/  49.6375, 898,  60.9446
FP(  6)=PT/  48.0820, 918,  59.8554
FP(  7)=PT/  46.1009, 938,  58.4682
FP(  8)=PT/  44.9228, 948,  57.6433
FP(  9)=PT/  57.7064, 818,  53.0969
FP( 10)=PT/  57.1455, 838,  52.8353
FP( 11)=PT/  56.4104, 858,  52.4926
FP( 12)=PT/  55.3715, 878,  52.0081
FP( 13)=PT/  53.9226, 898,  51.3325
FP( 14)=PT/  52.0075, 918,  50.4395
FP( 15)=PT/  49.5763, 938,  49.3058
FP( 16)=PT/  48.1292, 948,  48.6310
FP( 17)=PT/  60.9279, 818,  42.5136
FP( 18)=PT/  60.1991, 838,  42.3183
FP( 19)=PT/  59.2968, 858,  42.0765
FP( 20)=PT/  58.0545, 878,  41.7436
FP( 21)=PT/  56.3382, 898,  41.2838
FP( 22)=PT/  54.0822, 918,  40.6793
FP( 23)=PT/  51.2287, 938,  39.9147
FP( 24)=PT/  49.5307, 948,  39.4597
FP( 25)=PT/  62.2244, 818,  31.6319
```

```
FP( 26)=PT/  61.3397, 838,  31.5545
FP( 27)=PT/  60.2886, 858,  31.4625
FP( 28)=PT/  58.8695, 878,  31.3384
FP( 29)=PT/  56.9242, 898,  31.1682
FP( 30)=PT/  54.3755, 918,  30.9452
FP( 31)=PT/  51.1537, 938,  30.6633
FP( 32)=PT/  49.2394, 948,  30.4958
FP( 33)=PT/  61.9176, 818,  20.7709
FP( 34)=PT/  60.9249, 838,  20.8577
FP( 35)=PT/  59.7296, 858,  20.9623
FP( 36)=PT/  58.1200, 878,  21.1031
FP( 37)=PT/  55.9294, 898,  21.2948
FP( 38)=PT/  53.0792, 918,  21.5441
FP( 39)=PT/  49.5168, 938,  21.8558
FP( 40)=PT/  47.4346, 948,  22.0380
FP( 41)=PT/  60.2531, 818,  10.0432
FP( 42)=PT/  59.1625, 838,  10.3354
FP( 43)=PT/  57.8009, 858,  10.7002
FP( 44)=PT/  55.9368, 878,  11.1997
FP( 45)=PT/  53.4419, 898,  11.8682
FP( 46)=PT/  50.3017, 918,  12.7096
FP( 47)=PT/  46.5039, 938,  13.7273
FP( 48)=PT/  44.3338, 948,  14.3087
FP( 49)=PT/  57.2002, 818,   -.4849
FP( 50)=PT/  55.9726, 838,    .0875
FP( 51)=PT/  54.3162, 858,    .8598
FP( 52)=PT/  52.1634, 878,   1.8637
FP( 53)=PT/  49.4540, 898,   3.1272
FP( 54)=PT/  46.1781, 918,   4.6547
FP( 55)=PT/  42.3266, 938,   6.4507
FP( 56)=PT/  40.1671, 948,   7.4577
FP( 57)=PT/  53.5726, 818,  -8.6154
FP( 58)=PT/  52.1460, 838,  -7.6760
FP( 59)=PT/  50.3198, 858,  -6.4901
FP( 60)=PT/  48.0504, 878,  -5.0163
FP( 61)=PT/  45.2939, 898,  -3.2262
FP( 62)=PT/  42.0435, 918,  -1.1154
FP( 63)=PT/  38.2921, 938,   1.3207
FP( 64)=PT/  36.2156, 948,   2.6692
REFSYS / NOMORE
$$
FUS=SSURF/MESH,XYZ,    $ FUSELAGE SURFACE
                 SPLINE,FP(  1,THRU,  8),$
                 SPLINE,FP(  9,THRU, 16),$
                 SPLINE,FP( 17,THRU, 24),$
                 SPLINE,FP( 25,THRU, 32),$
                 SPLINE,FP( 33,THRU, 40),$
                 SPLINE,FP( 41,THRU, 48),$
                 SPLINE,FP( 49,THRU, 56),$
                 SPLINE,FP( 57,THRU, 64)
$$$$$     CALL  MACRO    $$$$$
    CUTTER/ 2
PPRINT PASS NO. 1
    CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS=-.10, THKDS=2, $
                 CUTMOD=NRMPS, TLSIDE=TLR
```

```
PPRINT PASS NO. 2
       CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS=-.10, THKDS=1, $
                    CUTMOD=NRMPS, TLSIDE=TLR
PPRINT PASS NO. 3
       CALL/CUTAN , PS=FUS, DS=PYLNUA, THKPS=-.20, THKDS=2, $
                    CUTMOD=NRMPS, TLSIDE=TLL
PPRINT PASS NO. 4
       CALL/CUTAN , PS=FUS, DS=PYLNUA, THKPS=-.20, THKDS=1, $
                    CUTMOD=NRMPS, TLSIDE=TLL
       CUTTER/1,.5
PPRINT CLEAN OUT CORNER WITH BALL ENDMILL
LOOPST
       A= .9
C1)    CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS=-.10, THKDS=A, $
                    CUTMOD=TA1,    TLSIDE=TLR
       A=A-.1
       IF(-.2-A) C1,C1,C2
C2)    A=0
C3)    CALL/CUTAN , PS=PYLNUA, DS=FUS, THKPS= A,  THKDS=-.2,$
                    CUTMOD=TA1,    TLSIDE=TLR
       A=A+.1
       IF(.9-A) C4,C4,C3
C4)LOOPND
FINI
```

3.3 Machine Outboard Side of Nacelle 16 x 60

```
PARTNO  SSIP/SSX3A -TEST 3- MACHINE OUTBD SIDE OF NACELLE 16X60  (CHUBB)
$$   PURPOSE - 1. DEMONSTRATE SPEED
$$             2. DEMONSTRATE SIMPLICITY OF MACHINING LARGE AREAS
$$             3. DEMONSTRATE -501 WARNING USEAGE
$$             4. DEMONSTRATE DISPLY/6
$$
$$------------------------------------------------------------------------
NOPOST
MULTAX
CLPRNT
MAXDP/ 10
TOLER/ .0025
CANON/ON
SSOFF =SSURF / DISPLY, OFF
$$$$$ DEFINITION OF NACFOB  $$$$$
$$
RESERV/ P1,16,P2,16,P3,16,P4,16,P5,16,P6,16,P7,16,P8,16,P9,16,P10,16
$$   PATCH NO. 1
$$
P1 ( 1)= POINT/   0.     ,  0.     ,   9.00      $$ SP00
P1 ( 2)= POINT/   0.     ,  2.50   ,  11.05      $$ SP10
P1 ( 3)=VECTOR/   0.     ,  0.     ,   2.35      $$ FD0C
P1 ( 4)=VECTOR/   0.     ,  4.9628,    1.7370    $$ FD10
P1 ( 5)= POINT/   9.     ,  0.     ,   0.        $$ SP01
P1 ( 6)= POINT/  11.05   ,  2.50   ,   0.        $$ SP11
P1 ( 7)=VECTOR/   2.35   ,  0.     ,   0.        $$ FD01
P1 ( 8)=VECTOR/   1.7370,   4.9628,    0.        $$ FD11
P1 ( 9)=VECTOR/  14.9117,   0.     ,   0.        $$ SD01
P1 (10)=VECTOR/  18.3082,   0.     ,   0.        $$ SD10
P1 (11)=VECTOR/   3.8935,   0.     ,   0.        $$ TW00
P1 (12)=VECTOR/   2.8778,   0.     ,   0.        $$ TW10
P1 (13)=VECTOR/   0.     ,  0.     , -14.9117    $$ SD01
P1 (14)=VECTOR/   0.     ,  0.     , -18.3082    $$ SD11
P1 (15)=VECTOR/   0.     ,  0.     ,  -3.8935    $$ TW01
P1 (16)=VECTOR/   0.     ,  0.     ,  -2.8778    $$ TW11
$$
$$   PATCH NO.2
$$
P2 ( 1)= POINT/   0.     ,  2.50   ,  11.05      $$ SP00
P2 ( 2)= POINT/   0.     ,  8.75   ,  12.25      $$ SP10
P2 ( 3)=VECTOR/   0.     ,  4.4833,    1.5692    $$ FD0C
P2 ( 4)=VECTOR/   0.     ,  7.9886,     .8148    $$ FD10
P2 ( 5)= POINT/  11.05   ,  2.50   ,   0.        $$ SP01
P2 ( 6)= POINT/  12.25   ,  8.75   ,   0.        $$ SP11
P2 ( 7)=VECTOR/   1.5692,   4.4833,    0.        $$ FD01
P2 ( 8)=VECTOR/    .8148,   7.9886,    0.        $$ FD11
P2 ( 9)=VECTOR/  18.3082,   0.     ,   0.        $$ SD00
P2 (10)=VECTOR/  20.2965,   0.     ,   0.        $$ SD10
P2 (11)=VECTOR/   2.5998,   0.     ,   0.        $$ TW00
P2 (12)=VECTOR/   1.350 ,   0.     ,   0.        $$ TW10
P2 (13)=VECTOR/   0.     ,  0.     , -18.3082    $$ SD01
P2 (14)=VECTOR/   0.     ,  0.     , -20.2965    $$ SD11
P2 (15)=VECTOR/   0.     ,  0.     ,  -2.5998    $$ TW01
P2 (16)=VECTOR/   0.     ,  0.     ,  -1.350     $$ TW11
$$
```

```
$$   PATCH NO. 3
$$
P3 ( 1)= POINT/   0.      ,  8.75    ,  12.25        $$ SP00
P3 ( 2)= POINT/   0.      ,  16.00   ,  12.60        $$ SP1C
P3 ( 3)=VECTOR/   0.      ,  6.7741,     .6910       $$ FD00
P3 ( 4)=VECTOR/   0.      ,  7.4300,    0.           $$ FD10
P3 ( 5)= POINT/  12.25    ,  8.75    ,  0.           $$ SP01
P3 ( 6)= POINT/  12.60    ,  16.00   ,  0.           $$ SP11
P3 ( 7)=VECTOR/    .6910,    6.7741,    0.           $$ FD01
P3 ( 8)=VECTOR/   0.      ,  7.430   ,  0.           $$ FD11
P3 ( 9)=VECTOR/  20.2965,    0.      ,  0.           $$ SD00
P3 (10)=VECTOR/  20.8764,    0.      ,  0.           $$ SD10
P3 (11)=VECTOR/   1.1448,    0.      ,  0.           $$ TW00
P3 (12)=VECTOR/   0.      ,  0.      ,  0.           $$ TW1C
P3 (13)=VECTOR/   0.      ,  0.      , -20.2965      $$ SD01
P3 (14)=VECTOR/   0.      ,  0.      , -20.8764      $$ SD11
P3 (15)=VECTOR/  -1.1448,    0.      ,  0.           $$ TW01
P3 (16)=VECTOR/   0.      ,  0.      ,  0.           $$ TW11
$$
$$   PATCH NO. 4
$$
P4 ( 1)= POINT/   0.      ,  16.00   ,  12.60        $$ SP00
P4 ( 2)= POINT/   0.      ,  55.00   ,  12.60        $$ SP10
P4 ( 3)=VECTOR/   0.      ,  21.8798,   0.           $$ FD00
P4 ( 4)=VECTOR/   0.      ,  39.9812,   0.           $$ FD1C
P4 ( 5)= POINT/  12.60    ,  16.00   ,  0.           $$ SP01
P4 ( 6)= POINT/  12.60    ,  55.00   ,  0.           $$ SP11
P4 ( 7)=VECTOR/   0.      ,  21.8798,   0.           $$ FD01
P4 ( 8)=VECTOR/   0.      ,  39.9812,   0.           $$ FD11
P4 ( 9)=VECTOR/  20.8764,    0.      ,  0.           $$ SD00
P4 (10)=VECTOR/  20.8764,    0.      ,  0.           $$ SD1C
P4 (11)=VECTOR/   0.      ,  0.      ,  0.           $$ TW00
P4 (12)=VECTOR/   0.      ,  0.      ,  0.           $$ TW10
P4 (13)=VECTOR/   0.      ,   .0     , -20.8764      $$ SD01
P4 (14)=VECTOR/   0.      ,   .0     , -20.8764      $$ SD11
P4 (15)=VECTOR/   0.      ,   .0     ,  0.           $$ TW01
P4 (16)=VECTOR/   0.      ,   .0     ,  0.           $$ TW11
$$
$$   PATCH NO. 5
$$
P5 ( 1)= POINT/   0.      ,  55.00   ,  12.60        $$ SP00
P5 ( 2)= POINT/   0.      ,  67.50   ,  11.65        $$ SP10
P5 ( 3)=VECTOR/   0.      ,  12.40   ,  0.           $$ FD00
P5 ( 4)=VECTOR/   0.      ,  12.60   , -1.90         $$ FD10
P5 ( 5)= POINT/  12.60    ,  55.00   ,  0.           $$ SP01
P5 ( 6)= POINT/  11.65    ,  67.50   ,  0.           $$ SP11
P5 ( 7)=VECTOR/   0.      ,  12.40   ,  0.           $$ FD01
P5 ( 8)=VECTOR/  -1.90    ,  12.60   ,  0.           $$ FD11
P5 ( 9)=VECTOR/  20.8764,    0.      ,  0.           $$ SD00
P5 (10)=VECTOR/  19.3024,    0.      ,  0.           $$ SD10
P5 (11)=VECTOR/   0.      ,  0.      ,  0.           $$ TW00
P5 (12)=VECTOR/  -3.1479,    0.      ,  0.           $$ TW10
P5 (13)=VECTOR/   0.      ,  0.      , -20.8764      $$ SD01
P5 (14)=VECTOR/   0.      ,  0.      , -19.3024      $$ SD11
P5 (15)=VECTOR/   0.      ,  0.      ,  0.           $$ TW01
```

```
P5 (16)=VECTOR/   0.      ,  0.      ,   3.1479    $$ TW11
$$
$$   PATCH NO. 6
$$
P6 ( 1)= POINT/   9.00    ,  0.      ,   0.        $$ SP00
P6 ( 2)= POINT/  11.05    ,  2.50    ,   0.        $$ SP10
P6 ( 3)=VECTOR/   2.35    ,  0.      ,   0.        $$ FD00
P6 ( 4)=VECTOR/   1.7370,    4.9628,     0.        $$ FD10
P6 ( 5)= POINT/   0.      ,  0.      ,  -9.0       $$ SP01
P6 ( 6)= POINT/   0.      ,  2.50    , -11.68      $$ SP11
P6 ( 7)=VECTOR/   0.      ,  0.      ,  -2.75      $$ FD01
P6 ( 8)=VECTOR/   0.      ,  4.9839,    -2.6041    $$ FD11
P6 ( 9)=VECTOR/   0.      ,  0.      , -14.9117    $$ SD00
P6 (10)=VECTOR/   0.      ,  0.      , -19.8560    $$ SD10
P6 (11)=VECTOR/   0.      ,  0.      ,  -3.8935    $$ TW00
P6 (12)=VECTOR/   0.      ,  0.      ,  -5.9343    $$ TW10
P6 (13)=VECTOR/ -14.9117,    0.      ,   0.        $$ SD01
P6 (14)=VECTOR/ -18.7850,    0.      ,   0.        $$ SD11
P6 (15)=VECTOR/  -4.5562,    0.      ,   0.        $$ TW01
P6 (16)=VECTOR/  -4.3972,    0.      ,   0.        $$ TW11
$$
$$   PATCH NO. 7
$$
P7 ( 1)= POINT/  11.05    ,  2.50    ,   0.        $$ SP00
P7 ( 2)= POINT/  12.25    ,  8.75    ,   0.        $$ SP10
P7 ( 3)=VECTOR/   1.5692,    4.4833,     0.        $$ FD00
P7 ( 4)=VECTOR/    .8148,    7.9886,     0.        $$ FD10
P7 ( 5)= POINT/   0.      ,  2.50    , -11.68      $$ SP01
P7 ( 6)= POINT/   0.      ,  8.75    , -13.79      $$ SP11
P7 ( 7)=VECTOR/   0.      ,  4.5024,    -2.3525    $$ FD01
P7 ( 8)=VECTOR/   0.      ,  7.9701,    -1.8411    $$ FD11
P7 ( 9)=VECTOR/   0.      ,  0.      , -19.8560    $$ SD00
P7 (10)=VECTOR/   0.      ,  0.      , -26.0355    $$ SD10
P7 (11)=VECTOR/   0.      ,  0.      ,  -5.3609    $$ TW00
P7 (12)=VECTOR/   0.      ,  0.      ,  -6.3067    $$ TW10
P7 (13)=VECTOR/ -18.7850,    0.      ,   0.        $$ SD01
P7 (14)=VECTOR/ -23.1280,    0.      ,   0.        $$ SD11
P7 (15)=VECTOR/  -3.9724,    0.      ,   0.        $$ TW01
P7 (16)=VECTOR/  -4.0157,    0.      ,   0.        $$ TW11
$$
$$   PATCH NO. 8
$$
P8 ( 1)= POINT/  12.25    ,  8.75    ,   0.        $$ SP00
P8 ( 2)= POINT/  12.60    , 16.00    ,   0.        $$ SP10
P8 ( 3)=VECTOR/    .6910,    6.7741,     0.        $$ FD00
P8 ( 4)=VECTOR/   0.      ,  7.43    ,   0.        $$ FD10
P8 ( 5)= POINT/   0.      ,  8.75    , -13.79      $$ SP01
P8 ( 6)= POINT/   0.      , 16.      , -15.1       $$ SP11
P8 ( 7)=VECTOR/   0.      ,  6.7584,    -1.5612    $$ FD01
P8 ( 8)=VECTOR/   0.      ,  8.0751,    -1.1023    $$ FD11
P8 ( 9)=VECTOR/   0.      ,  0.      , -26.0355    $$ SD00
P8 (10)=VECTOR/   0.      ,  0.      , -30.2       $$ SD10
P8 (11)=VECTOR/   0.      ,  0.      ,  -5.3479    $$ TW00
P8 (12)=VECTOR/   0.      ,  0.      ,  -2.8888    $$ TW10
P8 (13)=VECTOR/ -23.1280,    0.      ,   0.        $$ SD01
```

```
P8 (14)=VECTOR/ -25.2000,   0.     ,   0.         $$ SD11
P8 (15)=VECTOR/  -3.4052,   0.     ,   0.         $$ TW01
P8 (16)=VECTOR/   -.7733,   0.     ,   0.         $$ TW11
$$
$$    PATCH NO. 9
$$
P9 ( 1)= POINT/  12.60   ,  16.00  ,   0.         $$ SP00
P9 ( 2)= POINT/  12.60   ,  55.00  ,   0.         $$ SP10
P9 ( 3)=VECTOR/   0.     ,  21.8798,   0.         $$ FD00
P9 ( 4)=VECTOR/   0.     ,  39.9812,   0.         $$ FD10
P9 ( 5)= POINT/   0.     ,  16.00  , -15.1        $$ SP01
P9 ( 6)= POINT/   0.     ,  55.00  , -13.35       $$ SP11
P9 ( 7)=VECTOR/   0.     ,  23.7795,  -3.2459     $$ FD01
P9 ( 8)=VECTOR/   0.     ,  54.7655,   6.9120     $$ FD11
P9 ( 9)=VECTOR/   0.     ,   0.    , -30.20       $$ SD00
P9 (10)=VECTOR/   0.     ,   0.    , -23.2824     $$ SD10
P9 (11)=VECTOR/   0.     ,   0.    ,  -8.5068     $$ TW00
P9 (12)=VECTOR/   0.     ,   0.    ,  13.8751     $$ TW10
P9 (13)=VECTOR/ -25.2    ,   0.    ,   0.         $$ SD01
P9 (14)=VECTOR/ -21.9744,    0.    ,   0.         $$ SD11
P9 (15)=VECTOR/  -2.2771,    0.    ,   0.         $$ TW01
P9 (16)=VECTOR/   6.6244,    0.    ,   0.         $$ TW11
$$
$$    PATCH NO. 10
$$
P10( 1)= POINT/  12.60   ,  55.0   ,   0.         $$ SP00
P10( 2)= POINT/  11.65   ,  67.50  ,   0.         $$ SP10
P10( 3)=VECTOR/   0.     ,  12.40  ,   0.         $$ FD00
P10( 4)=VECTOR/  -1.90   ,  12.60  ,   0.         $$ FD10
P10( 5)= POINT/   0.     ,  55.0   , -13.35       $$ SP01
P10( 6)= POINT/   0.     ,  67.5   , -11.65       $$ SP11
P10( 7)=VECTOR/   0.     ,  16.9853,   2.1437     $$ FD01
P10( 8)=VECTOR/   0.     ,   7.9825,   1.2038     $$ FD11
P10( 9)=VECTOR/   0.     ,   0.    , -23.2824     $$ SD0C
P10(10)=VECTOR/   0.     ,   0.    , -19.3024     $$ SD10
P10(11)=VECTOR/   0.     ,   0.    ,   4.3033     $$ TW00
P10(12)=VECTOR/   0.     ,   0.    ,   3.1481     $$ TW1C
P10(13)=VECTOR/ -21.9744,    0.    ,   0.         $$ SD01
P10(14)=VECTOR/ -19.3024,    0.    ,   0.         $$ SD11
P10(15)=VECTOR/   2.0545,    0.    ,   0.         $$ TW01
P10(16)=VECTOR/   1.9944,    0.    ,   0.         $$ TW11
$$
PRINT/0
NACFOB= SSURF/ PATCH, PNTVEC, 6, 3,        PLUS,    $
          P1  (  ALL  ),                   PLUS,    $
          P2  (  ALL  ),                   PLUS,    $
          P3  (  ALL  ),                   PLUS,    $
          P4  (  ALL  ),                   PLUS,    $
          P5  (  ALL  ),                   PLUS,    $
          P6  (  ALL  ),                   PLUS,    $
          P7  (  ALL  ),                   PLUS,    $
          P8  (  ALL  ),                   PLUS,    $
          P9  (  ALL  ),                   PLUS,    $
          P10 (  ALL  )
PRINT/0
```

```
$$$$$ DEFINITION OF NACFIB  $$$$$
$$
REFSYS / (IVERSX=MATRIX/ -1,0,0,0, 0,1,0,0, 0,0,1,0 )
$$
NACFIB= SSURF/ PATCH, PNTVEC, 6, 3,      PLUS,     $
        P1  (  ALL  ),                   PLUS,     $
        P2  (  ALL  ),                   PLUS,     $
        P3  (  ALL  ),                   PLUS,     $
        P4  (  ALL  ),                   PLUS,     $
        P5  (  ALL  ),                   PLUS,     $
        P6  (  ALL  ),                   PLUS,     $
        P7  (  ALL  ),                   PLUS,     $
        P8  (  ALL  ),                   PLUS,     $
        P9  (  ALL  ),                   PLUS,     $
        P10 (  ALL  )
REFSYS / NOMORE
PRINT/0
     CK1  = PLANE/0,1,0,3
     CK2  = PLANE/0,1,0,67.
     CUTTER/ 2,1
$$$$$     MACHINING  LOOP    $$$$$
LOOPST
      TLAXIS/ 1
      A=-8
      GOTO/ 12,1,-10, 1,0,0
      PSIS/NACFOB
A1)   INDIRV/0,1,0
      GOFWD/(PLANE/0,0,1,A),CK2  $$  -501 WARNING WILL ACCRUE HERE
      A=A+.15
      IF(8-A) A3, A2, A2
A2)   INDIRV/0,-1,0
      GOFWD/(PLANE/0,0,1,A),CK1  $$  -501 WARNING WILL ACCRUE HERE
      A=A+.15
      IF(8-A) A3, A1, A1
A3)   GODLTA/ 5
      LOOPND
$$$$$       DISPLY / 6      $$$$$
      PSIS/ NACFOB
      DISPLY / 6
      FINI
```

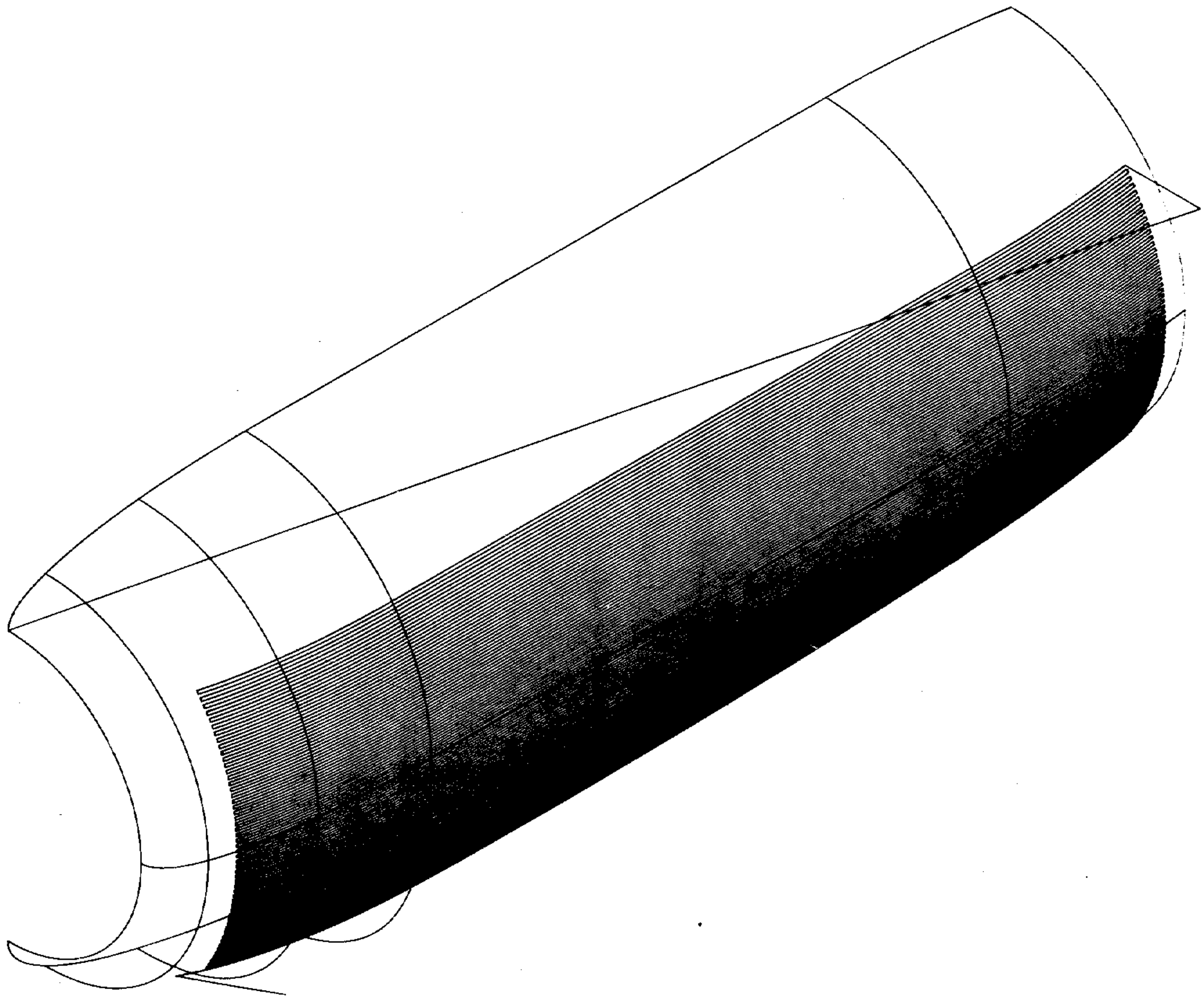


FIGURE 27 MACHINE NACELLE OUTBOARD SIDE 16 x 60

3.4 Tool Failure Crossing Boundaries

```
PARTNO  SSIP/SSX3A -TEST 4- TOOL FAILURE CROSSING BOUNDARIES     (CHUBB)
$$    PURPOSE - 1. DEMONSTRATE SSURF DIAGNOSTIC PRINT ON AND OFF
$$              2. DEMONSTRATE -501 WARNING
$$              3. DEMONSTRATE -2856 WARNING
$$              4. DEMONSTRATE DELETE/ ON AND OFF
$$              5. TEST TOOL RECOVERY AT BOUNDARIES
$$              6. TEST BOUNDARY MISMATCH
$$-----------------------------------------------------------------------
NOPOST
MULTAX/ON
CLPRNT/ON
CANON/ON
DS1   = PLANE / 1,0,0,0                                $$   90 DEG
DS2   = PLANE / .92397953, 0, -.38268343, 0            $$ 67.5 DEG
DS3   = PLANE / .70710678, 0, -.70710678, 0            $$   45 DEG
Y64   = PLANE / 0,1,0,64
Y72   = PLANE / 0,1,0,72
Y675  = PLANE / 0,1,0,67.5
AFTPT= POINT / 3,75,15
FWDPT= POINT / 3,62,15
PYV   = VECTOR/ 0,1,0
MYV   = VECTOR/ 0,-1,0
PRINT/0
$$$$$ DEFINITION OF NNOZO AND NNOZI   $$$$$
$$
CPTF =POINT/ 0, 67.5, 11.65
CPTM =POINT/ 0, 75, 10.1
CPTA =POINT/ 0, 81.5, 8.35
CONVF=VECTOR/ 0, 12.6, -1.9
CONVA=VECTOR/ 0, 1, -.290
YV = VECTOR / 0,1,0
NOR= POINT  / 0,0,0
PRINT/0
NOZCON= SCURV/ CURSEG,CPTF  , TANSPL,CONVF , CPTM, CPTA, TANSPL, CONVA
PRINT/0
NNOZO =SSURF/ REVOLV,NOZCON,XAXIS,NOR,YV,CCLW,  0, 180
PRINT/0
NNOZI =SSURF/ REVOLV,NOZCON,XAXIS,NOR,YV, CLW,  0, 180
PRINT/0
$$$$$ DEFINITION OF NACFOB  $$$$$
$$
RESERV/ P1,16,P2,16,P3,16,P4,16,P5,16,P6,16,P7,16,P8,16,P9,16,P10,16
$$   PATCH NO. 1
$$
P1 ( 1)= POINT/   0.      ,   0.      ,   9.00        $$ SP00
P1 ( 2)= POINT/   0.      ,   2.50    ,  11.05        $$ SP10
P1 ( 3)=VECTOR/   0.      ,   0.      ,   2.35        $$ FD00
P1 ( 4)=VECTOR/   0.      ,   4.9628,     1.7370      $$ FD10
P1 ( 5)= POINT/   9.      ,   0.      ,   0.          $$ SP01
P1 ( 6)= POINT/  11.05    ,   2.50    ,   0.          $$ SP11
P1 ( 7)=VECTOR/   2.35    ,   0.      ,   0.          $$ FD01
P1 ( 8)=VECTOR/   1.7370,     4.9628,     0.          $$ FD11
P1 ( 9)=VECTOR/  14.9117,     0.      ,   0.          $$ SD01
P1 (10)=VECTOR/  18.3082,     0.      ,   0.          $$ SD10
P1 (11)=VECTOR/   3.8935,     0.      ,   0.          $$ TW00
```

```
P1 (12)=VECTOR/   2.8778,   0.     ,   0.        $$ TW10
P1 (13)=VECTOR/   0.     ,   0.     , -14.9117     $$ SD01
P1 (14)=VECTOR/   0.     ,   0.     , -18.3082     $$ SD11
P1 (15)=VECTOR/   0.     ,   0.     ,  -3.8935     $$ TW01
P1 (16)=VECTOR/   0.     ,   0.     ,  -2.8778     $$ TW11
$$
$$    PATCH NO.2
$$
P2 ( 1)= POINT/   0.     ,   2.50   ,  11.05       $$ SP00
P2 ( 2)= POINT/   0.     ,   8.75   ,  12.25       $$ SP10
P2 ( 3)=VECTOR/   0.     ,   4.4833,   1.5692      $$ FD00
P2 ( 4)=VECTOR/   0.     ,   7.9886,    .8148      $$ FD10
P2 ( 5)= POINT/  11.05   ,   2.50   ,   0.         $$ SP01
P2 ( 6)= POINT/  12.25   ,   8.75   ,   0.         $$ SP11
P2 ( 7)=VECTOR/   1.5692,   4.4833,   0.         $$ FD01
P2 ( 8)=VECTOR/    .8148,   7.9886,   0.         $$ FD11
P2 ( 9)=VECTOR/  18.3082,   0.     ,   0.         $$ SD00
P2 (10)=VECTOR/  20.2965,   0.     ,   0.         $$ SD10
P2 (11)=VECTOR/   2.5998,   0.     ,   0.         $$ TW00
P2 (12)=VECTOR/   1.350 ,   0.     ,   0.         $$ TW10
P2 (13)=VECTOR/   0.     ,   0.     , -18.3082     $$ SD01
P2 (14)=VECTOR/   0.     ,   0.     , -20.2965     $$ SD11
P2 (15)=VECTOR/   0.     ,   0.     ,  -2.5998     $$ TW01
P2 (16)=VECTOR/   0.     ,   0.     ,  -1.350      $$ TW11
$$
$$    PATCH NO. 3
$$
P3 ( 1)= POINT/   0.     ,   8.75   ,  12.25       $$ SP00
P3 ( 2)= POINT/   0.     ,  16.00   ,  12.60       $$ SP10
P3 ( 3)=VECTOR/   0.     ,   6.7741,    .6910      $$ FD00
P3 ( 4)=VECTOR/   0.     ,   7.4300,   0.         $$ FD10
P3 ( 5)= POINT/  12.25   ,   8.75   ,   0.         $$ SP01
P3 ( 6)= POINT/  12.60   ,  16.00   ,   0.         $$ SP11
P3 ( 7)=VECTOR/    .6910,   6.7741,   0.         $$ FD01
P3 ( 8)=VECTOR/   0.     ,   7.430 ,   0.         $$ FD11
P3 ( 9)=VECTOR/  20.2965,   0.     ,   0.         $$ SD00
P3 (10)=VECTOR/  20.8764,   0.     ,   0.         $$ SD10
P3 (11)=VECTOR/   1.1448,   0.     ,   0.         $$ TW00
P3 (12)=VECTOR/   0.     ,   0.     ,   0.         $$ TW10
P3 (13)=VECTOR/   0.     ,   0.     , -20.2965     $$ SD01
P3 (14)=VECTOR/   0.     ,   0.     , -20.8764     $$ SD11
P3 (15)=VECTOR/  -1.1448,   0.     ,   0.         $$ TW01
P3 (16)=VECTOR/   0.     ,   0.     ,   0.         $$ TW11
$$
$$    PATCH NO. 4
$$
P4 ( 1)= POINT/   0.     ,  16.00   ,  12.60       $$ SP00
P4 ( 2)= POINT/   0.     ,  55.00   ,  12.60       $$ SP10
P4 ( 3)=VECTOR/   0.     ,  21.8798,   0.         $$ FD00
P4 ( 4)=VECTOR/   0.     ,  39.9812,   0.         $$ FD10
P4 ( 5)= POINT/  12.60   ,  16.00   ,   0.         $$ SP01
P4 ( 6)= POINT/  12.60   ,  55.00   ,   0.         $$ SP11
P4 ( 7)=VECTOR/   0.     ,  21.8798,   0.         $$ FD01
P4 ( 8)=VECTOR/   0.     ,  39.9812,   0.         $$ FD11
P4 ( 9)=VECTOR/  20.8764,   0.     ,   0.         $$ SD00
```

```
P4 (10)=VECTOR/  20.8764,    0.     ,    0.         $$ SD10
P4 (11)=VECTOR/   0.     ,    0.     ,    0.         $$ TW00
P4 (12)=VECTOR/   0.     ,    0.     ,    0.         $$ TW10
P4 (13)=VECTOR/   0.     ,     .0    , -20.8764      $$ SD01
P4 (14)=VECTOR/   0.     ,     .0    , -20.8764      $$ SD11
P4 (15)=VECTOR/   0.     ,     .0    ,    0.         $$ TW01
P4 (16)=VECTOR/   0.     ,     .0    ,    0.         $$ TW11
$$
$$   PATCH NO. 5
$$
P5 ( 1)= POINT/   0.     ,  55.00    ,  12.60        $$ SP00
P5 ( 2)= POINT/   0.     ,  67.50    ,  11.65        $$ SP10
P5 ( 3)=VECTOR/   0.     ,  12.40    ,   0.          $$ FD00
P5 ( 4)=VECTOR/   0.     ,  12.60    ,  -1.90        $$ FD10
P5 ( 5)= POINT/  12.60   ,  55.00    ,   0.          $$ SP01
P5 ( 6)= POINT/  11.65   ,  67.50    ,   0.          $$ SP11
P5 ( 7)=VECTOR/   0.     ,  12.40    ,   0.          $$ FD01
P5 ( 8)=VECTOR/  -1.90   ,  12.60    ,   0.          $$ FD11
P5 ( 9)=VECTOR/  20.8764,    0.     ,    0.          $$ SD00
P5 (10)=VECTOR/  19.3024,    0.     ,    0.          $$ SD10
P5 (11)=VECTOR/   0.     ,    0.     ,    0.          $$ TW00
P5 (12)=VECTOR/  -3.1479,    0.     ,    0.          $$ TW10
P5 (13)=VECTOR/   0.     ,    0.     , -20.8764      $$ SD01
P5 (14)=VECTOR/   0.     ,    0.     , -19.3024      $$ SD11
P5 (15)=VECTOR/   0.     ,    0.     ,    0.         $$ TW01
P5 (16)=VECTOR/   0.     ,    0.     ,    3.1479     $$ TW11
$$
$$   PATCH NO. 6
$$
P6 ( 1)= POINT/   9.00   ,    0.     ,    0.         $$ SP00
P6 ( 2)= POINT/  11.05   ,    2.50   ,    0.         $$ SP10
P6 ( 3)=VECTOR/   2.35   ,    0.     ,    0.         $$ FD00
P6 ( 4)=VECTOR/   1.7370,    4.9628,    0.          $$ FD10
P6 ( 5)= POINT/   0.     ,    0.     ,   -9.0        $$ SP01
P6 ( 6)= POINT/   0.     ,    2.50   , -11.68        $$ SP11
P6 ( 7)=VECTOR/   0.     ,    0.     ,   -2.75       $$ FD01
P6 ( 8)=VECTOR/   0.     ,    4.9839,   -2.6041      $$ FD11
P6 ( 9)=VECTOR/   0.     ,    0.     , -14.9117      $$ SD00
P6 (10)=VECTOR/   0.     ,    0.     , -19.8560      $$ SD10
P6 (11)=VECTOR/   0.     ,    0.     ,   -3.8935     $$ TW00
P6 (12)=VECTOR/   0.     ,    0.     ,   -5.9343     $$ TW10
P6 (13)=VECTOR/ -14.9117,    0.     ,    0.          $$ SD01
P6 (14)=VECTOR/ -18.7850,    0.     ,    0.          $$ SD11
P6 (15)=VECTOR/  -4.5562,    0.     ,    0.          $$ TW01
P6 (16)=VECTOR/  -4.3972,    0.     ,    0.          $$ TW11
$$
$$   PATCH NO. 7
$$
P7 ( 1)= POINT/  11.05   ,    2.50   ,    0.         $$ SP00
P7 ( 2)= POINT/  12.25   ,    8.75   ,    0.         $$ SP10
P7 ( 3)=VECTOR/   1.5692,    4.4833,    0.          $$ FD00
P7 ( 4)=VECTOR/    .8148,    7.9886,    0.          $$ FD10
P7 ( 5)= POINT/   0.     ,    2.50   , -11.68        $$ SP01
P7 ( 6)= POINT/   0.     ,    8.75   , -13.79        $$ SP11
P7 ( 7)=VECTOR/   0.     ,    4.5024,   -2.3525      $$ FD01
```

```
P7 ( 8)=VECTOR/   0.     ,   7.9701,  -1.8411      $$ FD11
P7 ( 9)=VECTOR/   0.     ,   0.    , -19.8560      $$ SD00
P7 (10)=VECTOR/   0.     ,   0.    , -26.0355      $$ SD10
P7 (11)=VECTOR/   0.     ,   0.    ,  -5.3609      $$ TW00
P7 (12)=VECTOR/   0.     ,   0.    ,  -6.3067      $$ TW10
P7 (13)=VECTOR/ -18.7850,    0.    ,   0.          $$ SD01
P7 (14)=VECTOR/ -23.1280,    0.    ,   0.          $$ SD11
P7 (15)=VECTOR/  -3.9724,    0.    ,   0.          $$ TW01
P7 (16)=VECTOR/  -4.0157,    0.    ,   0.          $$ TW11
$$
$$   PATCH NO. 8
$$
P8 ( 1)= POINT/  12.25   ,   8.75  ,   0.          $$ SP00
P8 ( 2)= POINT/  12.60   ,  16.00  ,   0.          $$ SP10
P8 ( 3)=VECTOR/    .6910,    6.7741,   0.          $$ FD00
P8 ( 4)=VECTOR/   0.     ,   7.43  ,   0.          $$ FD10
P8 ( 5)= POINT/   0.     ,   8.75  , -13.79        $$ SP01
P8 ( 6)= POINT/   0.     ,  16.    , -15.1         $$ SP11
P8 ( 7)=VECTOR/   0.     ,   6.7584,  -1.5612      $$ FD01
P8 ( 8)=VECTOR/   0.     ,   8.0751,  -1.1023      $$ FD11
P8 ( 9)=VECTOR/   0.     ,   0.    , -26.0355      $$ SD00
P8 (10)=VECTOR/   0.     ,   0.    , -30.2         $$ SD10
P8 (11)=VECTOR/   0.     ,   0.    ,  -5.3479      $$ TW00
P8 (12)=VECTOR/   0.     ,   0.    ,  -2.8888      $$ TW10
P8 (13)=VECTOR/ -23.1280,    0.    ,   0.          $$ SD01
P8 (14)=VECTOR/ -25.2000,    0.    ,   0.          $$ SD11
P8 (15)=VECTOR/  -3.4052,    0.    ,   0.          $$ TW01
P8 (16)=VECTOR/   -.7733,    0.    ,   0.          $$ TW11
$$
$$   PATCH NO. 9
$$
P9 ( 1)= POINT/  12.60   ,  16.00  ,   0.          $$ SP00
P9 ( 2)= POINT/  12.60   ,  55.00  ,   0.          $$ SP10
P9 ( 3)=VECTOR/   0.     ,  21.8798,   0.          $$ FD00
P9 ( 4)=VECTOR/   0.     ,  39.9812,   0.          $$ FD10
P9 ( 5)= POINT/   0.     ,  16.00  , -15.1         $$ SP01
P9 ( 6)= POINT/   0.     ,  55.00  , -13.35        $$ SP11
P9 ( 7)=VECTOR/   0.     ,  23.7795,  -3.2459      $$ FD01
P9 ( 8)=VECTOR/   0.     ,  54.7655,   6.9120      $$ FD11
P9 ( 9)=VECTOR/   0.     ,   0.    , -30.20        $$ SD00
P9 (10)=VECTOR/   0.     ,   0.    , -23.2824      $$ SD10
P9 (11)=VECTOR/   0.     ,   0.    ,  -8.5068      $$ TW00
P9 (12)=VECTOR/   0.     ,   0.    ,  13.8751      $$ TW10
P9 (13)=VECTOR/ -25.2    ,   0.    ,   0.          $$ SD01
P9 (14)=VECTOR/ -21.9744,    0.    ,   0.          $$ SD11
P9 (15)=VECTOR/  -2.2771,    0.    ,   0.          $$ TW01
P9 (16)=VECTOR/   6.6244,    0.    ,   0.          $$ TW11
$$
$$   PATCH NO. 10
$$
P10( 1)= POINT/  12.60   ,  55.0   ,   0.          $$ SP00
P10( 2)= POINT/  11.65   ,  67.50  ,   0.          $$ SP10
P10( 3)=VECTOR/   0.     ,  12.40  ,   0.          $$ FD00
P10( 4)=VECTOR/  -1.90   ,  12.60  ,   0.          $$ FD10
P10( 5)= POINT/   0.     ,  55.0   , -13.35        $$ SP01
```

```
P10( 6)= POINT/    0.     ,  67.5    , -11.65        $$ SP11
P10( 7)=VECTOR/    0.     ,  16.9853,    2.1437      $$ FD01
P10( 8)=VECTOR/    0.     ,   7.9825,    1.2038      $$ FD11
P10( 9)=VECTOR/    0.     ,   0.     , -23.2824      $$ SD00
P10(10)=VECTOR/    0.     ,   0.     , -19.3024      $$ SD10
P10(11)=VECTOR/    0.     ,   0.     ,   4.3033      $$ TW00
P10(12)=VECTOR/    0.     ,   0.     ,   3.1481      $$ TW10
P10(13)=VECTOR/ -21.9744,    0.     ,   0.           $$ SD01
P10(14)=VECTOR/ -19.3024,    0.     ,   0.           $$ SD11
P10(15)=VECTOR/    2.0545,   0.     ,   0.           $$ TW01
P10(16)=VECTOR/    1.9944,   0.     ,   0.           $$ TW11
$$
PRINT/0
SSON   =SSURF/DISPLY,ON    $$ TURNS DIAGNOSTIC PRINT ON
NACFOB= SSURF/ PATCH, PNTVEC, 6, 3,       PLUS,     $
        P1   (   ALL   ),                 PLUS,     $
        P2   (   ALL   ),                 PLUS,     $
        P3   (   ALL   ),                 PLUS,     $
        P4   (   ALL   ),                 PLUS,     $
        P5   (   ALL   ),                 PLUS,     $
        P6   (   ALL   ),                 PLUS,     $
        P7   (   ALL   ),                 PLUS,     $
        P8   (   ALL   ),                 PLUS,     $
        P9   (   ALL   ),                 PLUS,     $
        P10  (   ALL   )
PRINT/0
SSOFF  =SSURF/DISPLY,OFF   $$ TURNS DIAGNOSTIC PRINT OFF
$$$$$ DEFINITION OF NACFIB   $$$$$
$$
REFSYS / (IVERSX=MATRIX/ -1,0,0,0, 0,1,0,0, 0,0,1,0 )
$$
NACFIB= SSURF/ PATCH, PNTVEC, 6, 3,       PLUS,     $
        P1   (   ALL   ),                 PLUS,     $
        P2   (   ALL   ),                 PLUS,     $
        P3   (   ALL   ),                 PLUS,     $
        P4   (   ALL   ),                 PLUS,     $
        P5   (   ALL   ),                 PLUS,     $
        P6   (   ALL   ),                 PLUS,     $
        P7   (   ALL   ),                 PLUS,     $
        P8   (   ALL   ),                 PLUS,     $
        P9   (   ALL   ),                 PLUS,     $
        P10  (   ALL   )
REFSYS / NOMORE
PRINT/0
$$$$$     BOUNDARY  CHECK  MACRO     $$$$$
BCK   =MACRO /CUTMOD=1,D=2,R=0,PS1,PS2,TLCK=ON,CKST,CKND,STVEC,STPT, $
              DELET= NO, DS
      TLAXIS/CUTMOD
      CUTTER/ D,R
      FROM/STPT,0,0,1
      GO/ON,DS,PS1,ON,CKST
      INDIRV/ STVEC
      GOFWD/ DS, TLCK, Y675
      PSIS / PS2
      JUMPTO / DELET
```

```
YES)    DELETF/ ON
NO)     GOFWD/ DS,ON,CKND
        DELETF/ OFF
        TERMAC
$$
PPRINT  TEST 1 2IN CUTTER  FWD TO AFT  TLAXIS NORMPS
$$
CALL / BCK, CUTMOD=NORMPS, PS1=NACFCB,PS2=NNOZO, CKST=Y64, CKND=Y72,  $
        STVEC=PYV, STPT=FWDPT, DS=DS1
CALL / BCK, CUTMOD=NORMPS, PS1=NACFCB,PS2=NNOZO, CKST=Y64, CKND=Y72,  $
        STVEC=PYV, STPT=FWDPT, DS=DS2
CALL / BCK, CUTMOD=NORMPS, PS1=NACFCB,PS2=NNOZO, CKST=Y64, CKND=Y72,  $
        STVEC=PYV, STPT=FWDPT, DS=DS3
$$
PPRINT  TEST 2 10 IN BALL CUTTER FWD TO AFT  TLAXIS 1
$$
CALL / BCK,D=10,R=5,PS1=NACFOB,PS2=NNOZO,CKST=Y64,CKND=Y72,STVEC=PYV, $
        STPT=FWDPT, DS=DS1
CALL / BCK,D=10,R=5,PS1=NACFOB,PS2=NNOZO,CKST=Y64,CKND=Y72,STVEC=PYV, $
        STPT=FWDPT, DS=DS2
CALL / BCK,D=10,R=5,PS1=NACFOB,PS2=NNOZO,CKST=Y64,CKND=Y72,STVEC=PYV, $
        STPT=FWDPT, DS=DS3
$$
PPRINT  SAME PASS     WITH DELETE ON
$$
CALL / BCK, D=10,R=5,PS1=NACFOB,PS2=NNOZO,CKST=Y64,CKND=Y72,STVEC=PYV,$
        STPT=FWDPT, DS=DS3, DELET=YES
$$
PPRINT  SAME PASS  AFT TO FWD   W/ DELET OFF
$$
CALL / BCK,D=10,R=5,PS1=NNOZO,PS2=NACFOB,CKST=Y72,CKND=Y64,STVEC=MYV, $
        STPT= AFTPT, DS=DS3
FINI
```

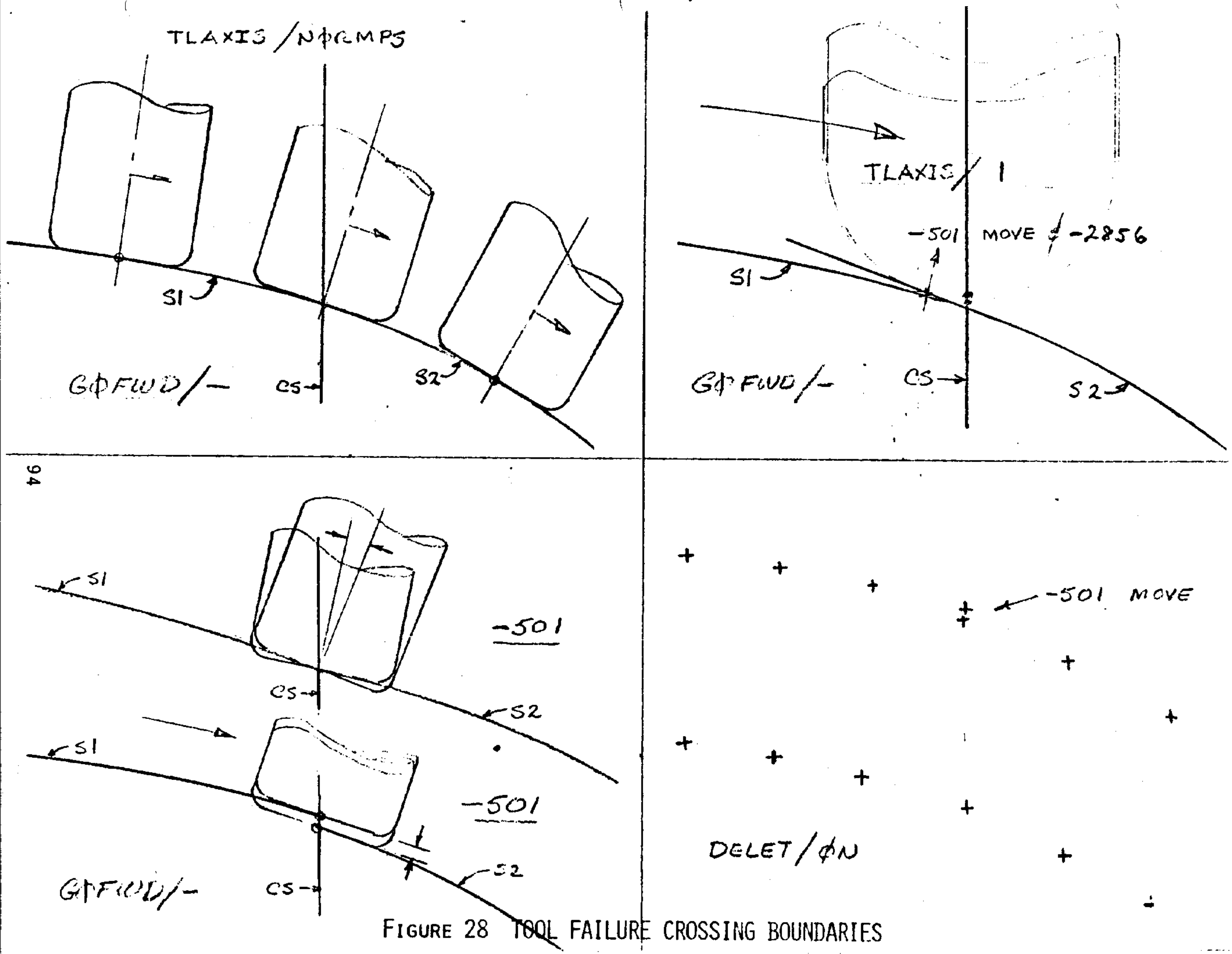


FIGURE 28 TOOL FAILURE CROSSING BOUNDARIES

3.5 HELIX Surface

```
PARTNO  SSIP/SSX3A -TEST 5-  HELIX SURFACE                                      (CHUBB)
$$    PURPOSE - 1. DEFINE HELIX S.S. SURFACE
$$              2. DEMONSTRATE  DISPLY/6
$$              3. DEMONSTRATE  SECTN3/ LARGE OR SMALL
$$              4. TEST HELIX AS PART SURFACE
$$              5. TEST HELIX AS DRIVE SURFACE
$$              6. TEST GO STARTUP
$$----------------------------------------------------------------------------------
PPUNCH/3
MACHIN/GERDAC,21050,OFF
MULTAX/ON
CLPRNT
TLAXIS/NORMPS
MAXDP/ 12
TOLER/.0025
CANON/ON
SSPRNT=SSURF/DISPLY,ON
PT0=POINT/0,0,0
ZVEC=VECTOR/0,0,1
$$$$$$  HELIX MACRO VARIABLES  $$$$$
$$
$$     P= PITCH PER REVOLUTION  NOTE- POZ P IS CCLW UP AND NEG P CCLW DWN
$$     R= HELIX RADIUS
$$     NREV= NUMBER OF REVOLUTIONS NOTE- MAX NUMBER IS 4
$$
$$     OUTPUT IS  PCC(N)   ( PARAMETRIC  CUBIC COEFFICIENT )
$$
$$
RESERV/ PCC,256      $$ MAX NUMBER REVOLUTIONS 4
DEFHLX = MACRO/ P, R, NREV
      A=0
      B=0
      C=1
      PP=0
      QP=P/4
      H=QP
      SD1B=4*R*.41421356
      SD1H=(P/(2*R*3.141592654))*SD1B
1A)   PCC(B+1)=POINT/0,0,(PP+(H-QP))
      PCC(B+2)=POINT/ (R*COSF(A)),(R*SINF(A)),(PP+(H-QP))
      PCC(B+3)=VECTOR/(R*COSF(A)),(R*SINF(A)),0
      PCC(B+4)=VECTOR/(R*COSF(A)),(R*SINF(A)),0
      PCC(B+5)=POINT/0,0,(PP+H)
      PCC(B+6)=POINT/ (R*COSF(A+90)),(R*SINF(A+90)),(PP+H)
      PCC(B+7)=VECTOR/(R*COSF(A+90)),(R*SINF(A+90)),0
      PCC(B+8)=VECTOR/(R*COSF(A+90)),(R*SINF(A+90)),0
      PCC(B+9 )=VECTOR/0,0,QP
      PCC(B+10)=VECTOR/(SD1B*COSF(A+90)),(SD1B*SINF(A+90)),SD1H
      PCC(B+11)=VECTOR/(SD1B*COSF(A+90)),(SD1B*SINF(A+90)),(SD1H-QP)
      PCC(B+12)=VECTOR/(SD1B*COSF(A+90)),(SD1B*SINF(A+90)),(SD1H-QP)
      PCC(B+13)=VECTOR/0,0,QP
      PCC(B+14)=VECTOR/(SD1B*COSF(A+180)),(SD1B*SINF(A+180)), SD1H
      PCC(B+15)=VECTOR/(SD1B*COSF(A+180)),(SD1B*SINF(A+180)),(SD1H-QP)
      PCC(B+16)=VECTOR/(SD1B*COSF(A+180)),(SD1B*SINF(A+180)),(SD1H-QP)
      A=A+90
```

```
        B=B+16
        H=H+QP
        IF(360-A)2A,2A,1A
2A)     A=0
        H=QP
        C=C+1
        PP=PP+P
        IF(NREV-C)3A,1A,1A
3A)     TERMAC
        CALL/DEFHLX, P=4, R=10,NREV=2
HELIX=SSURF/PATCH,PNTVEC,2, 9,        $
         PLUS,   PCC(  1, THRU,16 ), $     90 DEG
         PLUS,   PCC( 17, THRU,32 ), $    180 DEG
         PLUS,   PCC( 33, THRU,48 ), $    270 DEG
         PLUS,   PCC( 49, THRU,64 ), $    360 DEG
         PLUS,   PCC( 65, THRU,80 ), $    450 DEG
         PLUS,   PCC( 81, THRU,96 ), $    540 DEG
         PLUS,   PCC( 97, THRU,112), $    630 DEG
         PLUS,   PCC(113, THRU,128) $$    720 DEG
PENUP,DNTCUT
LOOPST
      A=1
1A)   GOTO/ A, 0, 0, 0,-.5,1
      INDIRV/0,1,0
      CUT,PENDWN
      PSIS/HELIX
      GOFWD/(CYLNDR/PTO,ZVEC,A), (PLANE/0,0,1,8)
      PENUP,DNTCUT
      A=A+1
      IF( 9-A) 2A,1A,1A
2A)   LOOPND
PSIS/HELIX
DISPLY/6
      CUT,PENDWN
      GOTO/ 11, -2, 2, 1,0,0
      GO/ (PLANE/ 0,0,1,0),(CONE/(POINT/0,0,10),(VECTOR/0,0,-1),40),HELIX
      INDIRV / 0,1,0
SECTN3/LARGE  $$ SEC 3 WILL OUTPUT LARGE PRINT FORMAT
      GOFWD/ HELIX,(PLANE /0,0,1,8)
SECTN3/SMALL  $$ SEC3 WILL OUTPUT SMALL PRINT FORMAT
      GOBACK/ HELIX,(PLANE/0,0,1,0)
      PENUP
PRINT/3,ALL
      FINI
```

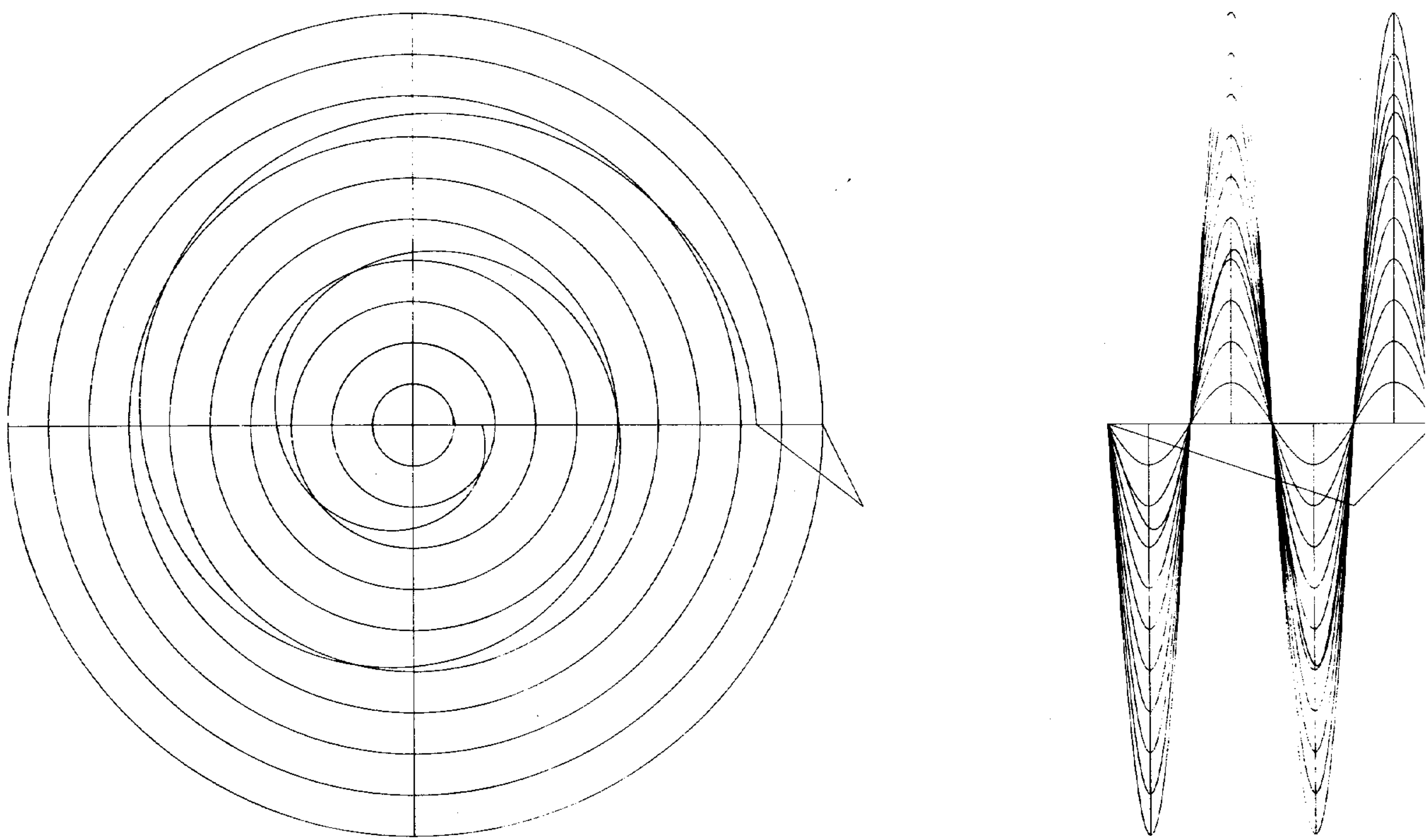


Figure 29 HELIX Surface

3.6 SSURF/REVOLV CLOCK Test

```
PARTNO  SSIP/SSX3A -TEST 6-  SSURF / REVOLV  CLOCK TEST          (CHUBB)
$$  PURPOSE -  1. TEST CLOCK DIRECTION ,
$$             2. TEST ANGULAR SEGMENT
$$
$$--------------------------------------------------------------------------
SSON  =SSURF/DISPLY,ON
P1 = POINT/ 0,2,0
P2 = POINT/ 4,4,0
P3 = POINT/ 8,3,0
TC = SCURV/ CURSEG, P1, P2, P3
V1 = VECTOR/ 1,0,0
P4 = POINT/ 0,0,0
P5 = POINT/ 8,0,0
S1A= SSURF/ REVOLV, TC, XAXIS, P4,V1, CLW,  0, 70
S1 = SSURF/REVOLV, TC, XAXIS, P4,P5, CLW,  0, 70 $$ SAME AS S1A
S2 = SSURF/REVOLV, TC, XAXIS, P4,P5, CLW, 80,250
S3 = SSURF/REVOLV, TC, XAXIS, P4,P5, CLW,350,10
S4 = SSURF/REVOLV, TC, XAXIS, P4,P5, CLW,-10,10
S5 = SSURF/REVOLV, TC, XAXIS, P4,P5,CCLW,170,0
S6 = SSURF/REVOLV, TC, XAXIS, P4,P5,CCLW,250,80
S7 = SSURF/REVOLV, TC, XAXIS, P4,P5,CCLW, 10,350
S8 = SSURF/REVOLV, TC, XAXIS, P4,P5,CCLW, 10,-10
S9 = SSURF/REVOLV, TC, XAXIS, P4,P5,CCLW,  0,190
FINI
```

3.7 SCURV/SPLINE Weight and Limits

```
PARTNO  SSIP/SSX3A -TEST 7-  SCURV/SPLINE  WEIGHT AND LIMITS     (CHUBB)
$$  PURPOSE -  1. TEST WEIGHT
$$             2. TEST LIMITS
$$
$$-----------------------------------------------------------------------
PARTNO        TEST  WEIGHT 2                                        CHUBB
DUM=SSURF/DISPLY,ON
P0 = POINT/ 0,0,0
P1 = POINT/   0,    1    ,  0
P2 = POINT/   4,    2.45 ,  0
P3 = POINT/   8,    3.3  ,  0
P4 = POINT/  12,    3.75 ,  0
P5 = POINT/  16,    4    ,  0
P6 = POINT/  20,    4.15 ,  0
P7 = POINT/  24,    4.5  ,  0
P8 = POINT/  28,    5.1  ,  0
V1 = VECTOR/ 1,0,0
TC1= SCURV/ SPLINE, WEIGHT, 0, LIMIT, .05, P1, WEIGHT, 1,  $
             P2, P3, P4, P5, P6, P7, P8, WEIGHT,1
TC2= SCURV/ SPLINE,  P1, P2,WEIGHT,0,LIMIT,.05,  $
                         P3,WEIGHT,0,LIMIT,.05,  $
                         P4,WEIGHT,0,LIMIT,.05,  $
                         P5,WEIGHT,0,LIMIT,.05,  $
                         P6,WEIGHT,0,LIMIT,.05,  $
                         P7,WEIGHT,0,LIMIT,.05,  $
                         P8
TS1= SSURF/ REVOLV , TC1, XAXIS, P0, V1, CLW,  0,70
TS2= SSURF/ REVOLV , TC2, XAXIS, P0, V1, CLW,  0,70
FINI
```

3.8 SSURF/MESH,XYZ Weight and Limits

```
PARTNO  SSIP/SSX3A -TEST 8-  SSURF/MESH,XYZ  WEIGHT AND LIMITS   (CHUBB)
$$  PURPOSE -  1. TEST WEIGHT
$$             2. TEST LIMITS
$$
$$--------------------------------------------------------------------
SSON  =SSURF/DISPLY,ON
P1 = POINT/   0.   ,    0.   ,    0.
P2 = POINT/   4.   ,    0.   ,    1.45
P3 = POINT/   8.   ,    0.   ,    2.3
P4 = POINT/  12.   ,    0.   ,    2.75
P5 = POINT/  16.   ,    0.   ,    3.
P6 = POINT/   0.   ,    4.   ,    1.45
P7 = POINT/   4.   ,    4.   ,    3.
P8 = POINT/   8.   ,    4.   ,    3.8
P9 = POINT/  12.   ,    4.   ,    4.15
P10= POINT/  16.   ,    4.   ,    4.2
P11= POINT/   0.   ,    8.   ,    1.
P12= POINT/   4.   ,    8.   ,    3.15
P13= POINT/   8.   ,    8.   ,    4.3
P14= POINT/  12.   ,    8.   ,    4.8
P15= POINT/  16.   ,    8.   ,    5.
TS =SSURF/ MESH, XYZ   ,  $
      SPLINE,  P1,  P2,  P3, WEIGHT, 0, P4, P5,          $
      SPLINE,  P6,  P7, WEIGHT, .05, P8, P9, P10,        $
      SPLINE,  P11, P12, P13,WEIGHT,0, LIMIT,.005,P14,P15
FINI
```