NASA Contractor Rep:-rt 158896

PLOTIT-METHOD OF INTERACTIVELY PLOTTING INPUT DATA FOR THE VORLAX COMPUTER PROGRAM

(NASA C -158896) PLOTIT-METHOD OF INTERACTIVELY PLOTTING INPUT DATA FOR THE VORLAX COMPUTER PROGRAM (Vought Corp., Hampton, Va.) 44 p HC A03/MF A01 CSCL 09B

N78-28830

Unclas

G3/61 25903

Frederick M. Denn

VOUGHT CORPORATION
Hampton Technical Center
3221 North Armistead Avenue • Hampton, Virginia 23666
an LTV Company

NASA CONTRACT NAS1-13500
JUNE 1978

NASA
National Aeronautics and
Space Administration
Langley Research Center
Hampton, Virginia 23665

SUMMARY

A method of plotting the geometric input to the VORLAX computer program by means of an interactive remote computer terminal is described. The software consists of a procedure file and two programs and was developed for use with the Langley Research Center computer system. The programs and procedure file are described and a sample execution is presented.

INTRODUCTION

The VORLAX computer program uses a sparse set of geometric input data to describe the aircraft configuration being analyzed. The geometry of the configuration can at times become very complex, and it is necessary to plot the configuration resulting from the data in order to ascertain its accuracy. The procedure file and two computer programs described herein provide a method for plotting this data at an interactive graphics terminal. A sample execution of the procedure is presented.

Use of this method allows the configuration to be plotted with any combination of roll, pitch, or yaw angles. Three independent forms of data display are available, and these may be specified in any combination. These are: (1) configurations with or without camber, (2) configurations showing only major panels or only minor panels, and (3) configurations with or without control points plotted. Any section of the plot may be enlarged for examination in greater detail.

The procedure file and computer programs to plot the VORLAX input data have been written to be used in the Langley Research Center computer system which provides a Network Operating System (NOS) and a Tektronix Plot 10 package. Langley Research Center users will find the procedure file and computer programs in mass storage as public files in the catalog of user number 214737C.

DISCUSSION

DESCRIPTION OF PROCEDURE FILES AND PROGRAMS

A method has been developed for plotting the VORLAX input data which consists of a procedure file and two computer programs. The procedure file PLOTIT and the computer programs READS and PLOTS are described in the following. A sample execution is provided which includes illustrations of the displays.

Procedure File PLOTIT

Procedure file PLOTIT (Appendix A) is used to simplify the plotting procedure. PLOTIT first gets the desired data file and renames it TAPE1. It then gets the binary form of program READS, which reads the input data and prepares files suitable for the subsequent operations. PLOTIT then gets and executes the binary form of PLOTS, which performs the aircraft geometry plotting.

Program READS

Program READS (Appendix B) reads the VORLAX input data from a disc file named TAPE1 and determines the necessary scaling factor in order for the aircraft drawing to fit on the screen.

The data for each of the panels are then read, and the coordinates of a set of points that describe the panels are calculated. These coordinates are stored on three disc files named TAPE3, TAPE4, and TAPE5. These data files will be read by program PLOTS.

Program READS provides the user with two methods of representing the aircraft. The first method displays the aircraft with camber as shown in figure 1(a). The second method shows the aircraft without camber. Figure 1(b) is the aircraft of figure 1(a) without camber.

Program PLOTS

Program PLOTS (Appendix C) reads the data produced by program READS. This program offers the user several variations in the display which are described in the following sections.

Rotation of the Aircraft. - The initial position of the aircraft is a side view with the nose to the left. From this position, it is possible to rotate the aircraft first in roll, then in pitch, and finally in yaw. Positive angles are defined as follows: roll, right wing down; pitch, nose up; and yaw, nose right.

Additional Independent Variations. - PLOTS offers two additional independent variations in the display. These variations are: (1) with or without subpaneling and (2) with or without control points. Examples of these variations are shown in figures 1(c) through 1(f). Figure 1(c) shows the aircraft without subpaneling or control points; figure 1(d) has subpaneling added; in figure 1(e) control points only have been added; and in figure 1(f) control points and subpaneling have been added. All of the plots are constructed using orthographic projection.

Recovery from Input Errors. - If an error has been made in specifying the input parameters, the execution of PLOTS can be stopped. There are three ways to stop PLOTS while it is executing. These are: (1) enter a value greater than 360 for roll angle, (2) answer any of the questions with STOP, or (3) stop the program while it is plotting by interrupting it with the break key and then entering an S.

Program PLOTS can be restarted at the beginning by sending the command PLOTB to the computer. PLOTB will also restart the program if it is stopped for any other reason.

SAMPLE EXECUTION

Figure 2 shows a sample execution of PLOTIT at a remote terminal. The first command gets the procedure file PLOTIT.

GET,PLOTIT/UN=214737C

The second command initiates the execution of the procedure file.

CALL,PLOTIT(T=VORLAXX)

VORLAXX is the name of the file on which the VORLAX data deck has been stored for this example. The file name in the calling statement can be any name which corresponds to a file on which VORLAX data is stored.

The first question asked by the computer deals with camber in the panels.

DO YOU WANT CAMBER IN THE PANELS ?
TRUE OR FALSE

If camber is desired in the panels, type in TRUE, otherwise type in FALSE and the camber will be set equal to zero.

The next three questions asked by the computer are concerned with the desired roll, pitch, and yaw angles of the configuration, and are as follows.

INPUT THE ROLL ANGLE FOR THE AIRCRAFT
(DEG), > 360 TO STOP.
PITCH ANGLE
YAW ANGLE

The angles desired in degrees, are typed in after the questions. If termination of the program is desired, a value greater than 360 may be typed in for the roll angle.

The next two questions are concerned with the desirability of displaying paneling and control points. The two questions are:

SUBPANELING ? TRUE OR FALSE
CONTROL POINTS ? TRUE OR FALSE

If these questions are answered TRUE, the subpaneling and control points are incorporated into the plots. If they are answered FALSE, then these quantities are deleted. The plot resulting from the input in figure 2 is presented in figure 3.

If a certain section of the plot needs to be enlarged in order to examine the plotted data more closely, this may be accomplished at the terminal. When the computer has finished plotting, it will print the following statement:

FOR ENLARGEMENT INPUT YES

At this point a hard copy can be made if desired. If any reply except YES is typed in, the computer will ask for a new set of angles. If YES is typed in, the graphics cursor (cross hairs) will appear. The cursor should then be located at the lower left corner [fig. 4(a)] of a rectangular region to be enlarged. A non-control keyboard character should be pressed. This will cause the cursor to disappear. The carriage return is then pressed. This sequence sends the coordinates of the first corner to the computer. The graphics cursor will reappear and should be relocated to the upper right corner of the desired rectangular region [fig. 4(b)]. A non-control keyboard character and the carriage return are then pressed as for the first corner. An enlargement of the region defined by these positions of the graphics cursor is shown in figure 4(c).

When the plot is finished, the computer will again print

FOR ENLARGEMENT INPUT YES

This allows a further enlargement of a section of the plot if desired.

CONCLUDING REMARKS

A plotting routine, PLOTIT, has been developed for plotting the input data for the VORLAX computer program. This program allows the user to plot geometric input data interactively at a remote graphics terminal and thereby ascertain very rapidly whether or not the data is correct.

The routine consists of two programs and a procedure file. These have been designed for use on the Control Data Corporation computer system with a Network Operating System (NOS) and a Tektronix Plot 10 graphics package at the NASA Langley Research Center.

REFERENCES

1. Miranda, Luis R.; Elliott, Robert D.; and Baker, William M.: A Generalized Vortex Lattice Method for Subsonic and Supersonic Flow Applications. NASA CR-2865, December 1977.

APPENDIX A

PROCEDURE FILE PLOTIT

This procedure file gets the binary form of READS(READB) and executes it, then gets the binary form of PLOTS (PLOTB) and executes it.

```
PLOTIT.
RETURN,TAPE3,TAPE4,TAPE5.
GET,TAPE1=T.
GET,READB/UN=214737C.
READB.
RETURN,READB.
GET,PLOTB/UN=214737C.
PLOTB.
EXIT.
```

APPENDIX B

SOURCE LISTING OF PROGRAM READS

```
      PROGRAM READS(INPUT,OUTPUT,TAPE1,TAPE4,TAPE2=OUTPUT,TAPE3,TAPE5)
C           THIS PROGRAM READS THE INPUT DATA FOR THE VORLAX PROGRAM
C           SO IT CAN BE PLOTED.
      DIMENSION TITLE(8)
      COMMON/BLOCK/XOFFSET
      LOGICAL CAMBER
      REAL LENGTH
      READ(1,100)TITLE
      READ(1,103)LAX,LAY
      READ(1,101)
      READ(1,102)NPAN,WSPAND
      CALL SIZES(NPAN,LENGTH,XOFFSET,WSPAN)
      IF(WSPAND.GT.WSPAN)WSPAN=WSPAND
      TESTRTO=LENGTH/WSPAN
      IF(LENGTH.GT.WSPAN)WSPAN=LENGTH
C        REALY IS THE WIDTH OF THE PLOTTING SURFACE.
      WRITE(3)NPAN,TITLE,TESTRTO
      REALY=10.0
      REALY=REALY-.20
      SCALE=REALY/WSPAN
      XOFFSET=-(XOFFSET+WSPAN*.5)
      WRITE(2,104)
      READ 105,CAMBER
      DO 200 I=1,NPAN
  200 CALL PANLRED(SCALE,LAX,LAY,CAMBER)
  100 FORMAT(8A10)
  101 FORMAT(//)
  102 FORMAT(I2,48X,F10.0)
  103 FORMAT(11X,I1,9X,I1)
  104 FORMAT("DO YOU WANT CAMBER IN THE PANELS ?  ",/,"TRUE OR FALSE")
  105 FORMAT(L7)
      STOP
      END
```

```
      SUBROUTINE PANLRED(SCALE,LAX,LAY,CAMBER)
      COMMON/BLOCK/XOFFSET
      COMMON/BLOKO/NAP,XAF(50),ZC(2,50),CORD1,CORD2,CD1SIN,CD2SIN
      COMMON/CANDS/SIN1,SIN2,COS1,COS2,DELTAY
      COMMON/ROJ1/ROJ,CROSSIZ
      LOGICAL CAMBER
      INTEGER PRD,RNCV
C
C       THIS SUBROUTINE READS THE VORLAX DATA FOR A PANEL,EACH TIME IT
C       IS CALLED.  THE COORDINATES OF A SET OF POINTS THAT DESCRIBE
C       THE PANEL ARE CALCULATED AND STORED ON DISC.
C
      DIMENSION X(2),Y(2),Z(2),CORD(2),RLE(2)
      COMMON/PHIRO/ PHI(100),RO(100),SINE(100),COSINE(100)
      COMMON VORS(3,500)
      COMMON/PI/PIE
      COMMON/TWIST/AINC1,DAINC
      REAL K
      DATA PIE/3.14159/,CROSSIZ/.02/
C
C       THIS SECTION READS THE VORLAX DATA CARDS FOR A PANEL.
C
      DO 200 I=1,2
  200 READ(1,100)X(I),Y(I),Z(I),CORD(I)
      DO 250 I=1,2
      X(I)=(X(I)+XOFFSET)*SCALE
      Y(I)=Y(I)*SCALE
      Z(I)=Z(I)*SCALE
  250 CORD(I)=CORD(I)*SCALE
      READ(1,101)TVOR,TNCV,PDL
      RNCV=INT(TNCV)
      NVOR=INT(TVOR)
      NVOR1=NVOR+1
      IF(PDL.LE.360.)GO TO 1
      READ(1,102)(PHI(N),RO(N),N=1,NVOR1)
      DO 251 N=1,NVOR1
      PHI(N)=PHI(N)*3.14159/180.
  251 RO(N)=RO(N)*SCALE
```

```
    1 READ(1,103)AINC1,AINC2,ITS,NAP,IQUANT,ISYNT,NPP
      AINC1=ATAN(AINC1)
      AINC2=ATAN(AINC2)
      DAINC=AINC2-AINC1
      IF(IQUANT.EQ.0)IQUANT=2
      IF(ISYNT.NE.0)READ(1,104)
      IF(NAP.LE.2)GO TO 2
      READ(1,102)(XAF(I),I=1,NAP)
      DO 88 I=1,NAP
   88 XAF(I)=XAF(I)*.01
      IF(ITS.EQ.0.OR.PDL.GE.360.)GO TO 3
      READ(1,105)RLE(1)
      RLE(1)=RLE(1)*SCALE
    3 READ(1,102)(ZC(1,I),I=1,NAP)
      IF(ITS.EQ.0.OR.PDL.GT.360.)GO TO 4
      READ(1,105)RLE(2)
      RLE(2)=RLE(2)*SCALE
    4 READ(1,102)(ZC(2,I),I=1,NAP)
      IF(.NOT.CAMBER.A.NAP.GT.2)CALL  ZEROZC1(PDL)
    2 CONTINUE
  100 FORMAT(4F10.0)
  101 FORMAT(2F10.0,10X,F10.0)
  102 FORMAT(8F10.0)
  103 FORMAT(2F10.0,I2,8X,I2,9X,I1,9X,I1,9X,I1)
  104 FORMAT(/)
  105 FORMAT(F10.0)
C
C        THIS SECTION CALCULATES THE LOCATION OF A SET OF POINTS
C        THAT DEFINE A FLAT PANEL.
C
      NVRN=(RNCV)*(NVOR+1)+1
      NVRNPRD=(RNCV+2)*(NVOR+1)
      N4= 4.0*RNCV
      N2=2.0*RNCV
      X1=X(1)
      CORD1=CORD(1)
C*****  IF THE PANEL IS CURVED GO TO 5
      IF(PDL.GT.360.)GO TO 5
      X2=X(2)
```

```
      Y1=Y(1)                                                    READS 111
      CORD2=CORD(2)                                              READS 112
      SIN1=SIN(AINC1)                                            READS 113
      SIN2=SIN(AINC2)                                            READS 114
      COS1=COS(AINC1)                                            READS 115
      COS2=COS(AINC2)                                            READS 116
      CD1SIN=CORD1*SIN1                                          READS 117
      CD1COS=CORD1*COS1                                          READS 118
      CD2SIN=CORD2*SIN2                                          READS 119
      CD2COS=CORD2*COS2                                          READS 120
      Z1=Z(1)                                                    READS 121
      Z2=Z(2)                                                    READS 122
      DELTAY=(Y(2)-Y1)                                           READS 123
      DELTAX=(X2-X1)                                             READS 124
      DELTAZ=(Z2-Z1)                                             READS 125
      DELTACO=(CD2COS-CD1COS)                                    READS 126
      YRAT=SQRT(DELTAZ*DELTAZ+DELTAY*DELTAY)                     READS 127
C                                                                READS 128
      ISETS=ISETSF(NVRNPRD)                                      READS 129
      CALL  SAVE(IFUG,3,IDUM,1)                                  READS 130
      WRITE(3)NVOR,RNCV,PDL,IQUANT,NVRNPRD,ISETS                 READS 131
      CALL  INTERP(0.0,ZZO,ZZ,XOO,XERP)                          READS 132
      DO 310 I=1,NVOR1                                           READS 133
      IF(LAY.EQ.1)RATIO=(I-1.)/NVOR                              READS 134
      IF(LAY.EQ.0)RATIO=.5*(1.-COS(PIE*(I-1.)/NVOR))             READS 135
      VORS(1,I)=X1+XOO+RATIO*(DELTAX+XERP)                       READS 136
      VORS(2,I)=Y1+DELTAY*RATIO                                  READS 137
  310 VORS(3,I)=ZZO+RATIO*(ZZ+DELTAZ)+Z1                         READS 138
C                                                                READS 139
      DO 501  I=NVOR1,NVRN,NVOR1                                 READS 140
      IF(LAX.EQ.1)PCORD=(I/NVOR1*4.0-3.0)/N4                     READS 141
      IF(LAX.EQ.0)PCORD=.5*(1.-COS((2.*I/NVOR1-1.)*PIE/N2))      READS 142
      CALL  INTERP(PCORD,ZZO,ZZ,XOO,XERP)                        READS 143
      DO 501  J=1,NVOR1                                          READS 144
      I2=I+J+IFUG                                                READS 145
      IF(LAY.EQ.1)RATIO=(J-1.)/NVOR                              READS 146
      IF(LAY.EQ.0)RATIO=.5*(1.-COS(PIE*(J-1.)/NVOR))             READS 147
```

```
      VORS(2,I2)=Y1+DELTAY*RATIO                                        READS
      VORS(3,I2)=Z1+DELTAZ*RATIO                                        READS
      XCORD=CD1COS+(DELTACO)*RATIO                                      READS
      XX=X1+RATIO*(DELTAX)                                              READS
      VORS(3,I2)=VORS(3,I2)+ZZO+ZZ*RATIO                                READS
      VORS(1,I2)=XX+PCORD*XCORD+X00+RATIO*XERP                          READS
  501 IF(I2.EQ.500)CALL SAVE(IFUG,3,IDUM,2)                             READS
      CALL INTERP(1.0,ZZ01,ZZ1,X001,XERP1)                              READS
      DO 300 I=1,NVOR1                                                  READS
      I2=NVRN+NVOR1+I-1+IFUG                                            READS
      IF(LAY.EQ.1)RATIO=(I-1.)/NVOR                                     READS
      IF(LAY.EQ.0)RATIO=.5*(1.-COS(PIE*(I-1.)/NVOR))                    READS
      VORS(2,I2)=Y1+DELTAY*RATIO                                        READS
      VORS(3,I2)=ZZ01+RATIO*(ZZ1+DELTAZ)+Z1                             READS
      VORS(1,I2)=(DELTAX+DELTACO+XERP1)*RATIO+X1+CD1COS+X001            READS
  300 IF(I2.EQ.500)CALL SAVE(IFUG,3,IDUM,2)                             READS
      CALL SAVE(IFUG,3,NVRNPRD,3)                                       READS
C                                                                       READS
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * READS
C                                                                       READS
C          THIS SECTION OF THE PROGRAM CALCULATES THE LOCATION OF THE   READS
C          CONTROL POINTS FOR A FLAT PANEL SUCH AS A WING.  IT THEN     READS
C          PLACES AN "X" ON EACH POINT.                                 READS
      PRD=NVOR*RNCV                                                     READS
      NNO=(RNCV-1)*NVOR+1                                               READS
      NNN=PRD*4                                                         READS
      ISETS=ISETSF(NNN)                                                 READS
      WRITE(4)NVOR,RNCV,PRD,NNN,NNN,NNO,ISETS                           READS
      CALL SAVE(IFUG,4,IDUM,1)                                          READS
      DO 1001 I=1,NNO,NVOR                                              READS
      K=(I-1)/NVOR+1                                                    READS
      IF(LAX.EQ.0)PCORD=.5*(1.-COS(K*PIE/RNCV))                         READS
      IF(LAX.EQ.1)PCORD=(4.*K-1.)/N4                                    READS
      CALL INTERP(PCORD,ZZO,ZZ,X00,XERP)                                READS
      YRAT=SQRT(DELTAY*DELTAY+(DELTAZ+ZZ)**2.)                          READS
      V1=CROSSIZ*DELTAY/YRAT                                            READS
      V2=CROSSIZ*(DELTAZ+ZZ)/YRAT                                       READS
      DO 1001 J=1,NVOR                                                  READS
```

```
      J2=(I+J-2)*4+IFUG                                                   READS
      J21=J2+1                                                            READS
      J22=J2+2                                                            READS
      J23=J2+3                                                            READS
      J24=J2+4                                                            READS
      IF(LAY.EQ.1)RATIO=(J-.5)/NVOR                                       READS
      IF(LAY.EQ.0)RATIO=.25*(2.-COS(PIE*J/NVOR)-COS(PIE*(J-1.)/NVOR))     READS
      CALL ANGLE(RATIO,WS,WC)                                             READS
      XCORD=CD1COS+DELTACO*RATIO                                          READS
      TEMX=X1+RATIO*(DELTAX+XERP)+PCORD*XCORD+X00                         READS
      VORS(1,J21)=TEMX-CROSSIZ*WC                                         READS
      VORS(1,J22)=TEMX+CROSSIZ*WC                                         READS
      VORS(1,J23)=VORS(1,J24)=TEMX                                        READS
      TEMY=Y1+DELTAY*RATIO                                                READS
      VORS(2,J21)=VORS(2,J22)=TEMY                                        READS
      TEMZ=Z1+DELTAZ*RATIO                                                READS
      TEMZ=TEMZ+ZZ0+ZZ*RATIO                                              READS
      VORS(3,J21)=TEMZ-CROSSIZ*WS                                         READS
      VORS(3,J22)=TEMZ+CROSSIZ*WS                                         READS
      VORS(3,J23)=TEMZ-V2                                                 READS
      VORS(3,J24)=TEMZ+V2                                                 READS
      VORS(2,J23)=TEMY-V1                                                 READS
      VORS(2,J24)=TEMY+V1                                                 READS
      IF(J24.EQ.500)CALL SAVE(IFUG,4,IDUM,2)                              READS
 1001 CONTINUE                                                            READS
      CALL SAVE(IFUG,4,NNN,3)                                             READS
 1010 CONTINUE                                                            READS
      RETURN                                                              READS
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *   READS
C       THIS SECTION OF THE PROGRAM IS FOR CURVED MAJOR PANELS.            READS
C                                                                         READS
    5 YNOT=Y(1)-RO(1)*COS(PHI(1))                                         READS
      ZNOT=Z(1)-RO(1)*SIN(PHI(1))                                         READS
      TEMP2=X1+CORD1                                                      READS
C        THIS CALCULATES THE SCALING FACTORS FOR THE RADIUS VECTORS       READS
C        FROM AREA RATIOS.                                                READS
      IF(NAP.LE.2)GO TO 90                                                READS
      DO 89 J=1,NAP                                                       READS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
   89 ZC(2,J)=SQRT(ZC(2,J)*.01)                                        READS    224
C                                                                      READS    225
C        THIS SECTION CALCULATES THE LOCATION OF A SET OF POINTS THAT  READS    226
C        DESCRIBE A CURVED MAJOR PANEL SUCH AS A FUSELAGE OR A NACELLE READS    227
C        WITH SUBPANELING.                                             READS    228
C                                                                      READS    229
   90 CALL INTERP2(0.0,XSHFT1,SKAL1,DM,DM,ZSHFT1,DM,9H NO POINT)       READS    230
      CALL SAVE(IFUG,5,IDUM,1)                                         READS    231
      ISETS=ISETSF(NVRNPRD)                                            READS    232
      WRITE(5)NVOR,RNCV,NVRNPRD,ISETS                                  READS    233
      DO 1333 I=1,NVOR1                                                READS    234
      R=RO(I)                                                          READS    235
      FI=PHI(I)                                                        READS    236
      VORS(1,I)=X1                                                     READS    237
      VORS(2,I)=R*COS(FI)*SKAL1+YNOT                                   READS    238
 1333 VORS(3,I)=R*SIN(FI)*SKAL1+ZNOT+ZSHFT1                            READS    239
C                                                                      READS    240
      DO 1501 I=NVOR1,NVRN,NVOR1                                       READS    241
      IF(LAX.EQ.1)PCORD=(I/NVOR1*4.-3.)/N4                             READS    242
      IF(LAX.EQ.0)PCORD=.5*(1.-COS((2.*I/NVOR1-1.)*PIE/N2))            READS    243
      CALL INTERP2(PCORD,XSHIFT,SKAL,DM,DM,ZSHIFT,DM,9H NO POINT)      READS    244
      XTEMP=X1+XSHIFT                                                  READS    245
      DO 1501 J=1,NVOR1                                                READS    246
      I2=I+J+IFUG                                                      READS    247
      R=RO(J)*SKAL                                                     READS    248
      FI=PHI(J)                                                        READS    249
      VORS(1,I2)=XTEMP                                                 READS    250
      VORS(2,I2)=R*COS(FI)+YNOT                                        READS    251
      VORS(3,I2)=R*SIN(FI)+ZNOT+ZSHIFT                                 READS    252
 1501 IF(I2.EQ.500)CALL SAVE(IFUG,5,IDUM,2)                            READS    253
      CALL INTERP2(1.0,XSHFT2,SKAL2,DM,DM,ZSHFT2,DM,9H NO POINT)       READS    254
      DO 1300 I=1,NVOR1                                                READS    255
      I2=NVRN+NVOR1+I-1+IFUG                                           READS    256
      R=RO(I)                                                          READS    257
      FI=PHI(I)                                                        READS    258
      VORS(1,I2)=TEMP2                                                 READS    259
      VORS(2,I2)=R*SKAL2*COS(FI)+YNOT                                  READS    260
      VORS(3,I2)=R*SKAL2*SIN(FI)+ZNOT+ZSHFT2                           READS    261
```

```
 1300 IF(I2.EQ.500)CALL SAVE(IFUG,5,IDUM,2)                            READS
      CALL SAVE(IFUG,5,NVRNPRD,3)                                       READS
C                                                                       READS
C        THIS SECTION CALCULATES THE LOCATION OF A SET OF POINTS THAT   READS
C         DESCRIBE A CURVED MAJOR PANEL, WITHOUT SUBPANELING.           READS
      NVRN=(NAP)*(NVOR+1)                                               READS
      NVRNPRD=(NAP)*(NVOR+1)                                            READS
      ISETS=ISETSF(NVRNPRD)                                             READS
      WRITE(3)NVOR ,NAP ,PDL,IQUANT,NVRNPRD,ISETS                       READS
      CALL SAVE(IFUG,3,NVRNPRD,1)                                       READS
      IF(NAP.LE.2)GO TO 556                                             READS
      DO 555 I=1,NVRN,NVOR1                                             READS
      II=(I)/NVOR1+1                                                    READS
      XTEMP=X1+CORD1*XAF(II)                                            READS
      SKAL=ZC(2,II)                                                     READS
      DO 555 J=1,NVOR1                                                  READS
      R=RO(J)*SKAL                                                      READS
      FI=PHI(J)                                                         READS
      I2=I+J-1+IFUG                                                     READS
      VORS(1,I2)=XTEMP                                                  READS
      VORS(2,I2)=R*COS(FI)+YNOT                                         READS
      VORS(3,I2)=R*SIN(FI)+ZNOT+CORD1*.01*ZC(1,II)                      READS
  555 IF(I2.EQ.500)CALL SAVE(IFUG,3,IDUM,2)                             READS
      IF(NAP.GT.2)GO TO 5555                                            READS
  556 NAP=2                                                             READS
      NVRNPRD=NAP*NVOR1                                                 READS
      DO 5554 J=1,NVOR1                                                 READS
      J2=J+NVOR1                                                        READS
      VORS(1,J)=X1                                                      READS
      VORS(1,J2)=X1+CORD1                                               READS
      VORS(2,J)=VORS(2,J2)=YNOT+RO(J)*COS(PHI(J))                       READS
 5554 VORS(3,J)=VORS(3,J2)=ZNOT+RO(J)*SIN(PHI(J))                       READS
 5555 CALL SAVE(IFUG,3,NVRNPRD,3)                                       READS
C                                                                       READS
C        THIS SECTION CALCULATES THE LOCATION OF THE CONTROL POINTS FOR READS
C         A CURVED MAJOR PANEL.  IT THEN PLACES AN "X" ON EACH POINT.   READS
      ROJ=RO(1)                                                         READS
      DO 2010 N=1,NVOR                                                  READS
```

```
      AZO=RO(N)*SIN(PHI(N))                                    READS    300
      AZ1=RO(N+1)*SIN(PHI(N+1))                                READS    301
      AYO=RO(N)*COS(PHI(N))                                    READS    302
      AY1=RO(N+1)*COS(PHI(N+1))                                READS    303
      DAZ=AZO-AZ1                                              READS    304
      DAY=AYO-AY1                                              READS    305
      H=SQRT(DAZ*DAZ+DAY*DAY)                                  READS    306
      SINE(N)=DAZ/H                                            READS    307
      COSINE(N)=DAY/H                                          READS    308
      R1=RO(N)                                                 READS    309
      R2=RO(N+1)                                               READS    310
      F1=PHI(N)                                                READS    311
      F2=PHI(N+1)                                              READS    312
      RO(N)=  (R1*SIN(F1)+R2*SIN(F2))*.5                       READS    313
 2010 PHI(N)=(R1*COS(F1)+R2*COS(F2))*.5                        READS    314
      PRD=NVOR*RNCV                                            READS    315
      NNO=(RNCV-1)*NVOR+1                                      READS    316
      NNN=PRD*4                                                READS    317
      ISETS=ISETSF(NNN)                                        READS    318
      WRITE(4)NVOR,RNCV,PRD,NNN,NNN,NNO,ISETS                  READS    319
      CALL  SAVE(IFUG,4,IDUM,1)                                READS    320
      DO 2001 I=1,NNO,NVOR                                     READS    321
      K=(I-1)/NVOR+1                                           READS    322
      IF(LAX.EQ.0)PCORD=.5*(1.-COS(K*PIE/RNCV))                READS    323
      IF(LAX.EQ.1)PCORD=(4.*K-1.)/N4                           READS    324
      CALL  INTERP2(PCORD,TEMX,SKAL,DSKL,CROSX,CAMB,DZ,6HPOINTS) READS  325
      TEMX=TEMX+X1                                             READS    326
      DO 2001  J=1,NVOR                                        READS    327
      J2=(I+J-2)*4+IFUG                                        READS    328
      J21=J2+1                                                 READS    329
      J22=J2+2                                                 READS    330
      J23=J2+3                                                 READS    331
      J24=J2+4                                                 READS    332
      VS=SINE(J)*CROSSIZ                                       READS    333
      HS=COSINE(J)*CROSSIZ                                     READS    334
      TEMY=PHI(J)*SKAL+YNOT                                    READS    335
      TY2=DSKL*PHI(J)                                          READS    336
      TEMZ=RO(J)*SKAL+ZNOT+CAMB                                READS    337
```

```
      TZ2=RO(J)*DSKL+DZ                                       READS   338
      VORS(1,J21)=TEMX-CROSX                                  READS   339
      VORS(1,J22)=TEMX+CROSX                                  READS   340
      VORS(1,J23)=TEMX                                        READS   341
      VORS(1,J24)=TEMX                                        READS   342
      VORS(2,J21)=TEMY-TY2                                    READS   343
      VORS(2,J22)=TEMY+TY2                                    READS   344
      VORS(2,J23)=TEMY-HS                                     READS   345
      VORS(2,J24)=TEMY+HS                                     READS   346
      VORS(3,J21)=TEMZ-TZ2                                    READS   347
      VORS(3,J22)=TEMZ+TZ2                                    READS   348
      VORS(3,J23)=TEMZ-VS                                     READS   349
      VORS(3,J24)=TEMZ+VS                                     READS   350
      IF(J24.EQ.500)CALL SAVE(IFUG,4,IDUM,2)                  READS   351
 2001 CONTINUE                                                READS   352
      CALL SAVE(IFUG,4,NNN,3)                                 READS   353
      RETURN                                                  READS   354
      END                                                     READS   355


      SUBROUTINE INTERP(P,ZA,Z,XA,X)                          READS   356
C       THIS SUBROUTINE DOES THE INTERPOLATION FOR CAMBER OF A FLAT READS 357
C       MAJOR PANEL.                                          READS   358
      COMMON/BLOKO/NAP,XAF(50),ZC(2,50),CORD1,CORD2,CD1SIN,CD2SIN READS 359
      COMMON/CANDS/SIN1,SIN2,COS1,COS2,DELTAY                 READS   360
      COMMON/TWIST/AINC1,DAINC                                READS   361
      COMMON/SLOPE/ANGL,DANG                                  READS   362
      IF(NAP.LE.2.OR.DELTAY.EQ.0.0)GO TO 3                    READS   363
      DO 1 I=2,NAP                                            READS   364
    1 IF(P.LE.XAF(I))GO TO 2                                  READS   365
    2 DAF=XAF(I)-XAF(I-1)                                     READS   366
      DZC1=ZC(1,I)-ZC(1,I-1)                                  READS   367
      DZC2=ZC(2,I)-ZC(2,I-1)                                  READS   368
      A1=ATAN(DZC1/DAF*.01)                                   READS   369
      A2=ATAN(DZC2/DAF*.01)                                   READS   370
      DAA=A2-A1                                               READS   371
      PC=(P-XAF(I-1))/DAF                                     READS   372
      CAMB1=(DZC1*PC+ZC(1,I-1))*CORD1*.01                     READS   373
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CAMB2=(DZC2*PC+ZC(2,I-1))*CORD2*.01                               READS    374
      GO TO 4                                                           READS    375
    3 CAMB1=CAMB2=A1=DAA=0.0                                            READS    376
    4 DANG=DAA+DAINC                                                    READS    377
      ANGL=AINC1+A1                                                     READS    378
      ZA=CAMB1*COS1+P*CD1SIN                                            READS    379
      ZB=CAMB2*COS2+P*CD2SIN                                            READS    380
      Z=ZB-ZA                                                           READS    381
      XA=-CAMB1*SIN1                                                    READS    382
      XB=-CAMB2*SIN2                                                    READS    383
      X=XB-XA                                                           READS    384
      RETURN                                                            READS    385
      END                                                               READS    386
C                                                                       READS    387

      SUBROUTINE INTERP2(P,X,S,DS,CROS,Z,DZ,TEST)                       READS    388
C       THIS SUBROUTINE DOES THE INTERPOLATION FOR THE RADIUS VECTORS   READS    389
C       AND THE CAMBER OF A CURVED MAJOR PANEL.                         READS    390
      COMMON/BLOKO/NAP,XAF(50),ZC(2,50),CORD1,CORD2,CD1SIN,CD2SIN       READS    391
      COMMON/ROJ1/ROJ,CROSSIZ                                           READS    392
      X=P*CORD1                                                         READS    393
      IF(NAP.LE.2)GO TO 3                                               READS    394
      DO 1 I=2,NAP                                                      READS    395
    1 IF(P.LE.XAF(I))GO TO 2                                            READS    396
    2 DAF=XAF(I)-XAF(I-1)                                               READS    397
      DZC1=ZC(1,I)-ZC(1,I-1)                                            READS    398
      DZC2=(ZC(2,I)-ZC(2,I-1))                                          READS    399
      PC=(P-XAF(I-1))/DAF                                               READS    400
      S=ZC(2,I-1)+DZC2*PC                                               READS    401
      Z=ZC(1,I-1)+DZC1*PC                                               READS    402
      Z=Z*.01*CORD1                                                     READS    403
      IF(TEST.NE.6HPOINTS)RETURN                                        READS    404
      DRAD=ROJ*DZC2                                                     READS    405
      DELX=DAF*CORD1                                                    READS    406
      COSINE=DELX/SQRT(DRAD*DRAD+DELX*DELX)                             READS    407
      CROS=COSINE*CROSSIZ                                               READS    408
      P2=CROS/CORD1                                                     READS    409
```

```
      PC2=(P+P2-XAF(I-1))/DAF                                          READS     410
      S2=ZC(2,I-1)+DZC2*PC2                                            READS     411
      Z2=ZC(1,I-1)+DZC1*PC2                                            READS     412
      Z2=Z2*CORD1*.01                                                  READS     413
      DZ=Z2-Z                                                          READS     414
      DS=S2-S                                                          READS     415
      RETURN                                                           READS     416
    3 S=S2=1.0                                                         READS     417
      Z=DS=DZ=0.0                                                      READS     418
      CROS=CROSSIZ                                                     READS     419
      RETURN                                                           READS     420
      END                                                              READS     421

      SUBROUTINE ANGLE(R,S,C)                                          READS     422
C      THIS SUBROUTINE IS USED FOR PLACING X'S ON CONTROL POINTS,      READS     423
C      ON FLAT MAJOR PANELS.                                           READS     424
      COMMON/SLOPE/ANGL,DANG                                           READS     425
      ANG=ANGL+R*DANG                                                  READS     426
      S=SIN(ANG)                                                       READS     427
      C=COS(ANG)                                                       READS     428
      RETURN                                                           READS     429
      END                                                              READS     430
C                                                                      READS     431

      SUBROUTINE ZEROZC1(PDL)                                          READS     432
C      THIS SUBROUTINE REMOVES CAMBER FROM THE MAJOR PANELS.           READS     433
      COMMON/BLOKO/NAP,XAF(50),ZC(2,50),CORD1,CORD2,CD1SIN,CD2SIN      READS     434
      DO 1 I=1,2                                                       READS     435
      IF(I.EQ.2.A.PDL.GE.360.)RETURN                                   READS     436
      DO 1 J=1,NAP                                                     READS     437
    1 ZC(I,J)=0.0                                                      READS     438
      RETURN                                                           READS     439
      END                                                              READS     440
C                                                                      READS     441
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      FUNCTION ISETSF(N)                                                READS    442
C      THIS FUNCTION DETERMINES THE NUMBER OF DATA SETS NEEDED TO       READS    443
C      TC DESCRIBE A MAJOR PANNEL.                                      READS    444
      R=1.0/500.*N                                                      READS    445
      ISETSF=INT(R)                                                     READS    446
      IF(ISETSF.LT.R)ISETSF=ISETSF+1                                    READS    447
      RETURN                                                            READS    448
      END                                                               READS    449
C                                                                       READS    450
      SUBROUTINE SAVE(FUG,FILE,PROD,TEST)                               READS    451
C      THIS SUBROUTINE MAKES IT POSSIBLE TO REDUCE CORE REQUIREMENTS    READS    452
C      BY REDUCING ARRAY SIZE.                                          READS    453
      INTEGER FUG,FILE,PNTS,TEST,PROD                                   READS    454
      COMMON VORS(3,500)                                                READS    455
      IF(TEST.EQ.2)GO TO 2                                              READS    456
      IF(TEST.EQ.3)GO TO 3                                              READS    457
      FUG=0                                                             READS    458
      RETURN                                                            READS    459
    2 FUG=FUG-500                                                       READS    460
      WRITE(FILE)((VORS(I,J),I=1,3),J=1,500)                            READS    461
      RETURN                                                            READS    462
    3 PNTS=MOD(PROD,500)                                                READS    463
      IF(PNTS.EQ.0)RETURN                                               READS    464
      WRITE(FILE)((VORS(I,J),I=1,3),J=1,PNTS)                           READS    465
      RETURN                                                            READS    466
      END                                                               READS    467
C                                                                       READS    468
      SUBROUTINE SIZES(NPAN,LENGTH,XOFFSET,WIDTH)                       READS    469
C **  DETERMINES THE LENGTH, WIDTH, STARTING POINT OF THE               READS    470
C **  AIRCRAFT.                                                         READS    471
      REAL LENGTH                                                       READS    472
      NPAN2=NPAN-1                                                      READS    473
      CALL FIND(LENGTH,XOFFSET,WIDTH)                                   READS    474
      IF(NPAN.EQ.1)GO TO 8                                              READS    475
```

```
      DO 7 I=1,NPAN2                                                    READS   476
      CALL FIND(BIG,SMALL,WIDE)                                         READS   477
      IF(SMALL.LT.XOFFSET)XOFFSET=SMALL                                 READS   478
      IF(WIDE.GT.WIDTH)WIDTH=WIDE                                       READS   479
    7 IF(BIG.GT.LENGTH)LENGTH=BIG                                       READS   480
      WIDTH=WIDTH*2.0                                                   READS   481
      LENGTH=LENGTH-XOFFSET                                             READS   482
    8 REWIND 1                                                          READS   483
      READ(1,5)                                                         READS   484
    5 FORMAT(/////)                                                     READS   485
      RETURN                                                            READS   486
      END                                                               READS   487
C                                                                       READS   488

      SUBROUTINE FIND(BIG,SMALL,WIDE)                                   READS   489
C **  DETERMINES THE STARTING POINT, END POINT, AND                     READS   490
C **  DISTANCE FROM THE AXIS FOR A PANEL.                               READS   491
      COMMON/BLOKO/NAP,XAF(50),ZC(2,50),CORD1,CORD2,CD1SIN,CD2SIN       READS   492
      COMMON/PHIRO/ PHI(100),RO(100),SINE(100),COSINE(100)              READS   493
      READ(1,100)X1,Y1,Z1,CORD1                                         READS   494
      READ(1,100)X2,Y2,Z2,CORD2                                         READS   495
      Y1=ABS(Y1)                                                        READS   496
      Y2=ABS(Y2)                                                        READS   497
      READ(1,101)TVOR,TNCV,PDL                                          READS   498
      NVOR1=INT(TVOR)+1                                                 READS   499
      IF(PDL.LE.360.)GO TO 1                                            READS   500
      READ(1,102)(PHI(N),RO(N),N=1,NVOR1)                               READS   501
    1 READ(1,103)AINC1,AINC2,ITS,NAP,IQUANT,ISYNT,NPP                   READS   502
      IF(ISYNT.NE.0)READ(1,104)                                         READS   503
      IF(NAP.LE.2)GO TO 2                                               READS   504
      READ(1,102)(XAF(I),I=1,NAP)                                       READS   505
      IF(ITS.EQ.0.OR.PDL.GE.360.)GO TO 3                                READS   506
      READ(1,105)R                                                      READS   507
    3 READ(1,102)(ZC(1,I),I=1,NAP)                                      READS   508
      IF(ITS.EQ.0.OR.PDL.GT.360.)GO TO 4                                READS   509
      READ(1,105)R                                                      READS   510
```

APPENDIX B. - Concluded

```
    4 READ(1,102)(ZC(2,I),I=1,NAP)                          READS
    2 CONTINUE                                              READS
  100 FORMAT(4F10.0)                                        READS
  101 FORMAT(2F10.0,10X,F10.0)                              READS
  102 FORMAT(8F10.0)                                        READS
  103 FORMAT(2F10.0,I2,8X,I2,9X,I1,9X,I1,9X,I1)             READS
  104 FORMAT(/)                                             READS
  105 FORMAT(F10.0)                                         READS
C                                                           READS
      SMALL=X2                                              READS
      IF(X1.LT.SMALL)SMALL=X1                               READS
   11 X1=X1+CORD1                                           READS
      X2=X2+CORD2                                           READS
      BIG=X2                                                READS
      IF(X1.GT.BIG)BIG=X1                                   READS
      WIDE=Y1                                               READS
      IF(Y2.GT.WIDE)WIDE=Y2                                 READS
      RETURN                                                READS
      END                                                   READS
```

APPENDIX C

SOURCE LISTING OF PROGRAM PLOTS

```
      PROGRAM PLOTS(INPUT,OUTPUT,TAPE2=OUTPUT,TAPE3,TAPE4,TAPE5)
C       THIS INITIALIZES THE PLOTTING ROUTINES AND CALLS PLOTPAN.
      CALL INITT(120)
      CALL TERM(3,4096)
      CALL CHRSIZ(4)
      CALL PLOTPAN
      CALL FINITT(0,0)
      STCP
      END
C


      SUBROUTINE PLOTPAN
      COMMON VORS(3,500)
      COMMON /PRAMS/NVOR,NVRNPRD,RNCV,NVOR1,NVRN
      COMMON /TITL/ TITLE(6)
      COMMON /ROT/KARRAY2(7),AMYTRIX(3,3)
      INTEGER RNCV
C
C        THIS PROGRAM READS A SET OF 3 ANGLES AND PLOTS THE
C        CONFIGURATION AFTER ROTATING IT THROUGH THE INDICATED ANGLES.
C        FIRST ONE SIDE IS PLOTTED THEN IT IS REFLECTED THROUGH
C        ITS X-Z PLANE AND THE OTHER SIDE IS PLOTTED.
C        THE ROTATIONS ARE CARIED OUT BY MATRIX MULTPICATION.
C        IT IS ALSO DETERMINED IF THE SUBPANELING AND CONTRUL
C        POINTS ARE TO BE SHOWN.
C
      LOGICAL  SUBLINE,CPONTZ
      DATA (KARRAY2(M),M=1,7)/20,3, 3,500,3,3,3/
      DATA PIE/3.14159/
    9 REWIND 3
      REWIND 4
      REWIND 5
```

```
      READ(3)NPAN,TITLE,TESTRTO
C
C       HERE THE ANGLES OF ROTATION ARE READ; ROLL, PITCH, AND YAW,
C       ARE READ.
C
      CALL ERASE
      WRITE(2,104)
      READ *,ROLL
      IF(ROLL.GT.360.)RETURN
      WRITE(2,105)
      READ *, PITCH
      WRITE(2,106)
      READ *,YAW
      WRITE(2,107)
      READ 112,SUBLINE
      WRITE(2,108)
      READ 112,CPONTZ
  104 FORMAT(" INPUT THE ROLL ANGLE FOR THE AIRCRAFT",/,
     2" (DEG), >360 TO STOP.")
  105 FORMAT("PITCH ANGLE")
  106 FORMAT("YAW ANGLE")
  107 FORMAT("SUBPANELING ?  TRUE OR FALSE")
  108 FORMAT("CONTROL POINTS ? TRUE OR FALSE")
  112 FORMAT(L7)
C
      IF(TESTRTO.LT.1.00)GO TO 500
      IF(TESTRTO.GT.1.25)TESTRTO=1.25
      RYY=5.0/TESTRTO
      IXX2=INT(TESTRTO*3200)
      IXX1=INT((4096-IXX2)*.5)
      CALL SWINDO(IXX1,IXX2,1,3200)
      CALL DWINDO(-5.0,5.0,-RYY,RYY)
      GO TO 501
  500 CALL SWINDO(450,3200,1,3200)
      CALL DWINDO(-5.0,5.0,-5.0,5.0)
  501 CALL TURNIT(ROLL,PITCH,YAW)
    3 CALL MOVABS(100,3500)
      CALL ANMODE
```

```
      WRITE(2,788)TITLE,ROLL,PITCH,YAW                                  PLOTS  70
  788 FORMAT(8A10,/," THE ROLL ANGLE IS  ",F7.1,                        PLOTS  71
     1/," THE PITCH ANGLE IS ",F7.1,                                    PLOTS  72
     2/," THE YAW ANGLE IS   ",F7.1)                                    PLOTS  73
      DO 300 NPANS=1,NPAN                                               PLOTS  74
      DO 299 L=1,2                                                      PLOTS  75
      READ(3)NVOR ,RNCV ,PDL,IQUANT,NVRNPRD,ISETS                       PLOTS  76
      ISET=ISETS                                                        PLOTS  77
      NVOR1=NVOR+1                                                      PLOTS  78
      NVRN=NVRNPRD-NVOR1+1                                              PLOTS  79
      IF(SUBLINE.A.PDL.LT.360)CALL VORSUB(5HLINES,ISETS,3,L)            PLOTS  80
      IF(.NOT.SUBLINE.A.PDL.GT.360)CALL VORSUB(5HLINES,ISETS,3,L)       PLOTS  81
      IF(SUBLINE.A.PDL.GT.360)CALL RNDLINE(L)                           PLOTS  82
      IF(.NOT.SUBLINE.A.PDL.LT.360)CALL VORSUB(5HEDGES,ISETS,3,L)       PLOTS  83
      IF(PDL.GE.0.0.A.CPONTZ)CALL CPOINTS(L)                            PLOTS  84
      IF(SUBLINE.A.PDL.GT.360.)ISET=0                                   PLOTS  85
      IF(L.EQ.2.OR.IQUANT.EQ.1)GO TO 300                                PLOTS  86
      CALL BACKUP(3,ISET+1)                                             PLOTS  87
  299 CONTINUE                                                          PLOTS  88
  300 IF(L.EQ.2.A.ISET.EQ.0.OR.IQUANT.EQ.1.A.ISET.EQ.0)CALL SKPFILE(3,  PLOTS  89
     1ISETS)                                                            PLOTS  90
      CALL MOVABS(100,3500)                                             PLOTS  91
      CALL BIGER(II)                                                    PLOTS  92
      IF(II-1)9,9,3                                                     PLOTS  93
      END                                                               PLOTS  94
C                                                                       PLOTS  95


      SUBROUTINE VORSUB(WHICH,ISETS,IFILE,L)                            PLOTS  96
      COMMON /PRAMS/NVOR,NVRNPRD,RNCV,NVOR1,NVRN                        PLOTS  97
      COMMON VORS(3,500)                                                PLOTS  98
C                                                                       PLOTS  99
C        THIS SUBROUTINE PLOTS A FLAT MAJOR PANEL WITH                  PLOTS 100
C        OR WITHOUT SUBPANELING, OR A CURVED PANEL WITHOUT SUBPANELING. PLOTS 101
C                                                                       PLOTS 102
      CALL SAVE(IFUG,IFILE,IDUM,1,L,ISETS)                              PLOTS 103
      IF(WHICH.EQ.5HLINES)GO TO 1                                       PLOTS 104
      N1=NVRN-1                                                         PLOTS 105
```

```
      N2=NVOR                                                           PLOTS
      GO TO 2                                                           PLOTS
    1 N1=NVOR1                                                          PLOTS
      N2=1                                                              PLOTS
    2 CALL SAVE(IFUG,IFILE,NVRNPRD,2,L,ISETS)                           PLOTS
      DO 257 I=1,NVRN,N1                                                PLOTS
      DO 257 J=1,NVOR1                                                  PLOTS
      I2=I+J-1+IFUG                                                     PLOTS
      IF(I2.LE.500)GO TO 3                                              PLOTS
   10 I2=I2-500                                                         PLOTS
      CALL SAVE(IFUG,IFILE,NVRNPRD,2,L,ISETS)                           PLOTS
      IF(I2.GT.500)GO TO 10                                             PLOTS
    3 IF(J.EQ.1)CALL MOVEA(VORS(1,I2 ),VORS(3,I2 ))                     PLOTS
  257 CALL DRAWA(VORS(1,I2 ),VORS(3,I2 ))                               PLOTS
      DO 258 J=1,NVOR1,N2                                               PLOTS
      IF(ISETS.EQ.1)GO TO 4                                             PLOTS
      CALL BACKUP(IFILE,ISETS)                                          PLOTS
      CALL SAVE(IFUG,IFILE,NVRNPRD,1,L,ISETS)                           PLOTS
      CALL SAVE(IFUG,IFILE,NVRNPRD,2,L,ISETS)                           PLOTS
    4 DO 258 I=1,NVRN,NVOR1                                             PLOTS
      I2=I+J-1+IFUG                                                     PLOTS
      IF(I2.LE.500)GO TO 5                                              PLOTS
      CALL SAVE(IFUG,IFILE,NVRNPRD,2,L,ISETS)                           PLOTS
      I2=I2-500                                                         PLOTS
    5 IF(I.EQ.1)CALL MOVEA(VORS(1,I2 ),VORS(3,I2 ))                     PLOTS
  258 CALL DRAWA(VORS(1,I2 ),VORS(3,I2 ))                               PLOTS
      RETURN                                                            PLOTS
      END                                                               PLOTS
C                                                                       PLOTS


      SUBROUTINE REFLECT(N)                                             PLOTS
      COMMON VORS(3,500)                                                PLOTS
C      THIS SUBROUTINE REFLECTS THE CONFIGURATION ACCROSS ITS X-Z PLANE.PLOTS
      DO 1 J=1,N                                                        PLOTS
    1 VORS(2,J)=-VORS(2,J)                                              PLOTS
      RETURN                                                            PLOTS
      END                                                               PLOTS
```

```
C
      SUBROUTINE BIGER(I)
C       THIS SUBROUTINE CAUSES A PORTION OF THE PLOT DEFINED WITH
C       THE GRAPHICS CURSER, TO BE ENLARGED.
C
      CALL ANMODE
      WRITE(2,103)
      READ 102, ENLARGE
      IF(ENLARGE.EQ.3HYES)GO TO 1
      I=1
      RETURN
    1 CALL VCURSR(ICHAR,X1,Y1)
      CALL VCURSR(ICHAR,X2,Y2)
      CALL ERASE
      RATIO=(X2-X1)/(Y2-Y1)
      IF(RATIO.LT.1.25)GO TO 10
      IX1=48
      IX2=4000
      IY2=INT(4000/RATIO)
      IY1=INT((3200-IY2)*.5)
      GO TO 20
   10 IY1=1
      IY2=3200
      IX2=INT(3200*RATIO)
      IX1=INT((4098-IX2)*.5)
   20 CALL SWINDO(IX1,IX2,IY1,IY2)
      CALL DWINDO(X1,X2,Y1,Y2)
      REWIND 3
      REWIND 4
      REWIND 5
      READ(3)
      I=2
      RETURN
  102 FORMAT(1A3)
  103 FORMAT(/////," FOR ENLARGEMENT INPUT YES")
      END
```

```
C                                                                  PLOTS
      SUBROUTINE CPOINTS(L)                                        PLOTS
C       THIS SUBROUTINE PLOTS X'S ON THE CONTROL POINTS.           PLOTS
      COMMON VORS(3,500)                                           PLOTS
      COMMON/PRAMS/NVOR,NVRNPRD,RNCV,NVOR1,NVRN                    PLOTS
      INTEGER PRD,RNCV                                             PLOTS
      IF(L.EQ.2)CALL BACKUP(4,ISETS+1)                             PLOTS
      READ(4)NVOR,RNCV,PRD,NNN,NVRNPRD,NNO,ISETS                   PLOTS
      CALL SAVE(IFUG,4,IDUM,1,L,ISETS)                             PLOTS
      CALL SAVE(IFUG,4,NNN,2,L,ISETS)                              PLOTS
      DO 1 J=1,NNO,NVOR                                            PLOTS
      DO 1 I=1,NVOR                                                PLOTS
      K=(I+J-2)*4+IFUG                                             PLOTS
      K4=K+4                                                       PLOTS
      CALL MOVEA(VORS(1,K+1),VORS(3,K+1))                          PLOTS
      CALL DRAWA(VORS(1,K+2),VORS(3,K+2))                          PLOTS
      CALL MOVEA(VORS(1,K+3),VORS(3,K+3))                          PLOTS
      CALL DRAWA(VORS(1, K4),VORS(3, K4))                          PLOTS
    1 IF(K4.EQ.500)CALL SAVE(IFUG,4,NNN,2,L,ISETS)                 PLOTS
      RETURN                                                       PLOTS
      END                                                          PLOTS
C                                                                  PLOTS

      SUBROUTINE RNDLINE(L)                                        PLOTS
C       THIS DEFINES A SET OF PARAMETERS SO A CURVED MAJOR         PLOTS
C       PANEL WITH SUBPANELING CAN BE PLOTED.                      PLOTS
      INTEGER RNCV                                                 PLOTS
      COMMON VORS(3,500)                                           PLOTS
      COMMON/PRAMS/NVOR,NVRNPRD,RNCV,NVOR1,NVRN                    PLOTS
      IF(L.EQ.2)CALL BACKUP(5,ISETS+1)                             PLOTS
      READ(5)NVOR,RNCV,NVRNPRD,ISETS                               PLOTS
      NVOR1=NVOR+1                                                 PLOTS
      NVRN=NVRNPRD-NVOR                                            PLOTS
      CALL VORSUB(5HLINES,ISETS,5,L)                               PLOTS
      RETURN                                                       PLOTS
      END                                                          PLOTS
C                                                                  PLOTS
```

```
      SUBROUTINE SAVC(FUG,FILE,P,TEST,L,SETS)                            PLOTS  214
C      THIS SUBROUTINE MAKES IT POSSIBLE TO REDUCE THE CORE              PLOTS  215
C      REQUIREMENTS OF THE PROGRAM.                                      PLOTS  216
      INTEGER FUG,FILE,P,TEST,SETS                                       PLOTS  217
      COMMON VORS(3,500)                                                 PLOTS  218
      COMMON /PRAMS/NVOR,NVRNPRD,RNCV,NVOR1,NVRN                         PLOTS  219
      COMMON/ROT/KARRAY2(7),AMYTRIX(3,3)                                 PLOTS  220
      IF(TEST.EQ.1)GO TO 1                                               PLOTS  221
      FUG=-500*TEST2                                                     PLOTS  222
      IF(TEST2.EQ.SETS)RETURN                                            PLOTS  223
      TEST2=TEST2+1                                                      PLOTS  224
      IF(TEST2.LT.SETS)N=500                                             PLOTS  225
      IF(TEST2.EQ.SETS)N=MOD(P,500)                                      PLOTS  226
      IF(N.EQ.0)N=500                                                    PLOTS  227
      READ(FILE)((VORS(I,J),I=1,3),J=1,N)                                PLOTS  228
      IF(L.EQ.2)CALL REFLECT(N)                                          PLOTS  229
      KARRAY2(4)=N                                                       PLOTS  230
      CALL MATOPS(KARRAY2,AMYTRIX,VORS,VORS)                             PLOTS  231
      RETURN                                                             PLOTS  232
    1 TEST2=FUG=0                                                        PLOTS  233
      RETURN                                                             PLOTS  234
      END                                                                PLOTS  235
C                                                                        PLOTS  236


      SUBROUTINE BACKUP(I,N)                                             PLOTS  237
C       CAUSES FILE I TO BE BACKSPACED N RECORDS.                        PLOTS  238
      DO 1 J=1,N                                                         PLOTS  239
    1 BACKSPACE I                                                        PLOTS  240
      RETURN                                                             PLOTS  241
      END                                                                PLOTS  242
C                                                                        PLOTS  243
```

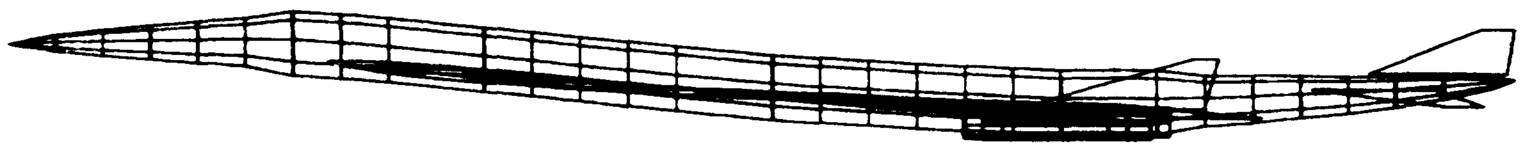
ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE SKPFILE(I,N)                                          PLOTS
C      CAUSES FILE I TO BE ADVANCED N RECORDS.                         PLOTS
      DO 1 J=1,N                                                       PLOTS
    1 READ(I)                                                          PLOTS
      RETURN                                                           PLOTS
      END                                                              PLOTS


      SUBROUTINE TURNIT(RO,PO,YO)                                      PLOTS
C                                                                      PLOTS
C        THIS SUBROUTINE DETERMINES THE MAYTRIX WHICH WILL ROTATE      PLOTS
C        THE AIRCRAFT THROUGH THE ANGLES ROLL PITCH AND YAW AS         PLOTS
C        THEY ARE NORMALY DETERMINED.                                  PLOTS
C                                                                      PLOTS
      DIMENSION ZAX(3,3),XAX(3,3),R1(3,3),R2(3,3),VECTOR(3)            PLOTS
      DIMENSION KMULT(7),MULTV(7)                                      PLOTS
      COMMON /ROT/ KARRAY2(7),AMYTRIX(3,3)                             PLOTS
      DATA(KMULT(M),M=1,7)/20,6*(3)/                                   PLOTS
      DATA(MULTV(M),M=1,7)/20,3,3,1,3,3,3/                             PLOTS
      DATA PIE/3.14159/                                                PLOTS
      DATA XAX(1,2),XAX(1,3),XAX(2,1),XAX(3,1)/4*(0.0)/                PLOTS
      DATA ZAX(1,3),ZAX(2,3),ZAX(3,1),ZAX(3,2)/4*(0.0)/                PLOTS
      DATA R1(1,2),R1(1,3),R1(2,1),R1(3,1)/4*(0.0)/                    PLOTS
      DATA R2(1,2),R2(2,1),R2(2,3),R2(3,2)/4*(0.0)/                    PLOTS
      DATA XAX(1,1),ZAX(3,3),R1(1,1),R2(2,2)/4*(1.0)/                  PLOTS
    1 AMYTRIX(1,1)=AMYTRIX(2,2)=AMYTRIX(3,3)=1.0                       PLOTS
      AMYTRIX(1,2)=AMYTRIX(1,3)=AMYTRIX(2,1)=AMYTRIX(2,3)=0.0          PLOTS
      AMYTRIX(3,1)=AMYTRIX(3,2)=0.0                                    PLOTS
      VECTOR(1)=VECTOR(2)=0.0                                          PLOTS
      VECTOR(3)=1.0                                                    PLOTS
      R=RO*PIE/180.                                                    PLOTS
      P=-PO*PIE/180.                                                   PLOTS
      Y=-YO*PIE/180.                                                   PLOTS
      RTO=PIE*.5-R                                                     PLOTS
      XAX(2,2)=XAX(3,3)=COS(R)                                         PLOTS
      SINE=SIN(R)                                                      PLOTS
      XAX(2,3)=SINE                                                    PLOTS
      XAX(3,2)=-SINE                                                   PLOTS
```

```
      ZAX(1,1)=ZAX(2,2)=COS(P)                                          PLOTS  280
      SINE=SIN(P)                                                       PLOTS  281
      ZAX(1,2)=-SINE                                                    PLOTS  282
      ZAX(2,1)=SINE                                                     PLOTS  283
      R1(2,2)=R1(3,3)=COS(RTO)                                          PLOTS  284
      SINE=SIN(RTO)                                                     PLOTS  285
      R1(2,3)=SINE                                                      PLOTS  286
      R1(3,2)=-SINE                                                     PLOTS  287
      CALL MATOPS(KMULT,XAX,AMYTRIX,AMYTRIX)                            PLOTS  288
      CALL MATOPS(KMULT,R1,AMYTRIX,AMYTRIX)                             PLOTS  289
      CALL MATOPS(KMULT,ZAX,AMYTRIX,AMYTRIX)                            PLOTS  290
C        TRANSPOSE MAYTRIX R1.                                          PLOTS  291
      R1(2,3)=-SINE                                                     PLOTS  292
      R1(3,2)=SINE                                                      PLOTS  293
      CALL MATOPS(KMULT,R1,AMYTRIX,AMYTRIX)                             PLOTS  294
    5 ZAX(1,1)=ZAX(2,2)=COS(Y)                                          PLOTS  295
      SINE=SIN(Y)                                                       PLOTS  296
      ZAX(1,2)=-SINE                                                    PLOTS  297
      ZAX(2,1)=SINE                                                     PLOTS  298
      CALL MATOPS(MULTV,AMYTRIX,VECTOR,VECTOR)                          PLOTS  299
      A=VECTOR(1)                                                       PLOTS  300
      B=VECTOR(2)                                                       PLOTS  301
      C=VECTOR(3)                                                       PLOTS  302
      V=SQRT(B*B+C*C)                                                   PLOTS  303
      BOVRV=B/V                                                         PLOTS  304
      R1(2,2)=R1(3,3)=C/V                                               PLOTS  305
      R1(2,3)=-BOVRV                                                    PLOTS  306
      R1(3,2)=BOVRV                                                     PLOTS  307
      R2(1,1)=R2(3,3)=V                                                 PLOTS  308
      R2(1,3)=-A                                                        PLOTS  309
      R2(3,1)=A                                                         PLOTS  310
      CALL MATOPS(KMULT,R1,AMYTRIX,AMYTRIX)                             PLOTS  311
      CALL MATOPS(KMULT,R2,AMYTRIX,AMYTRIX)                             PLOTS  312
      CALL MATOPS(KMJLT,ZAX,AMYTRIX,AMYTRIX)                            PLOTS  313
C         TRANSPOSE MAYTRIX R1 AND MAYTRIX R2.                          PLOTS  314
      R1(2,3)=BOVRV                                                     PLOTS  315
```

```
      R1(3,2)=-BOVRV                                            PLOTS    316
      R2(1,3)=+A                                                PLOTS    317
      R2(3,1)=-A                                                PLOTS    318
      CALL MATOPS(KMULT,R2,AMYTRIX,AMYTRIX)                     PLOTS    319
      CALL MATOPS(KMULT,R1,AMYTRIX,AMYTRIX)                     PLOTS    320
      RETURN                                                    PLOTS    321
      END                                                       PLOTS    322
```

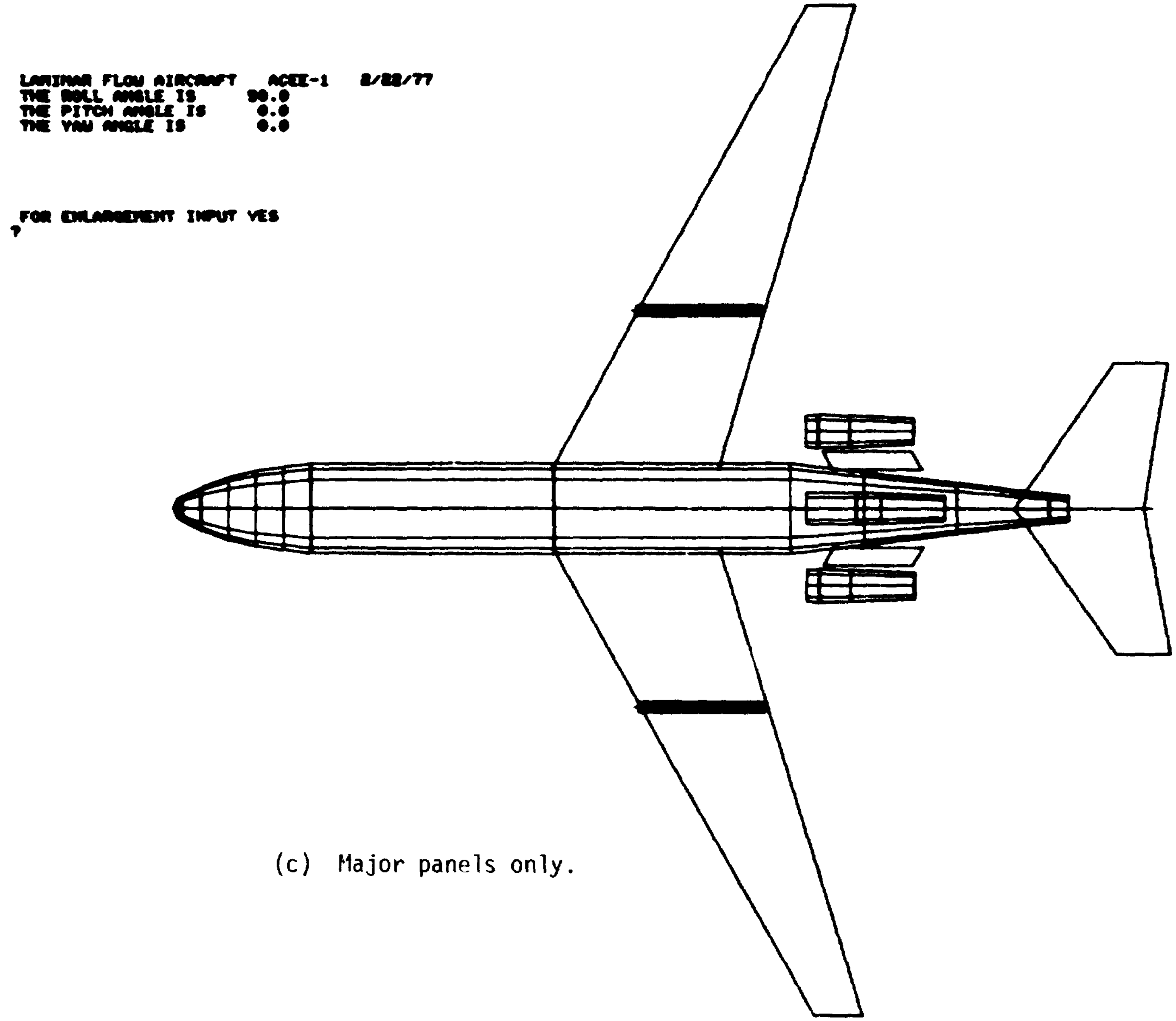
```
AST 100 - 6/6/75 PER COORD SHEET 164(REVISED),712 LB/SEC ENGINES (AST-JP-2)
 THE ROLL ANGLE IS       0.0
 THE PITCH ANGLE IS      0.0
 THE YAW ANGLE IS        0.0

 FOR ENLARGEMENT INPUT YES
?
```



(a) With panel camber displayed.

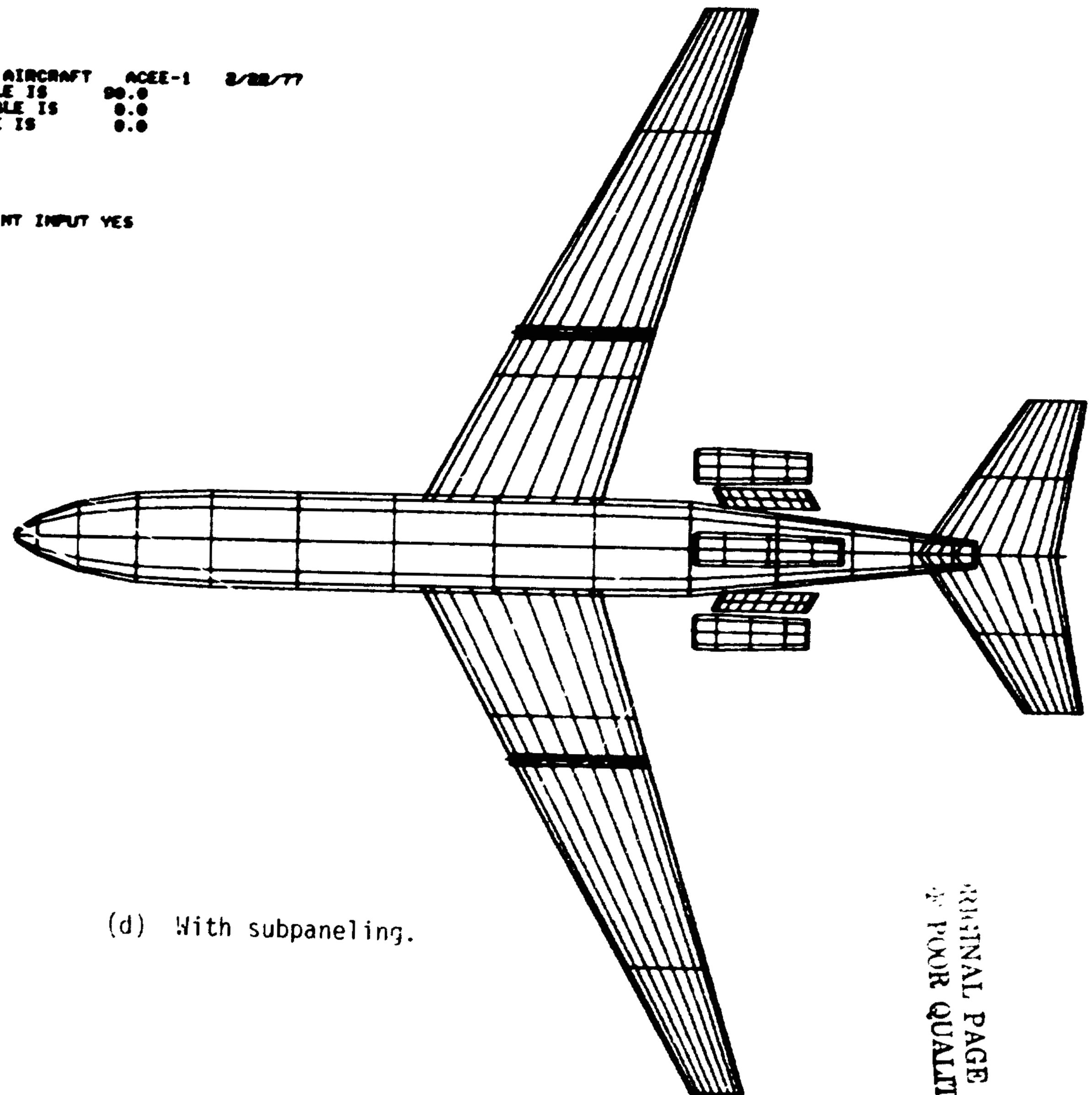
(b) Without panel camber displayed.

Figure 1. - Display of aircraft configuration.



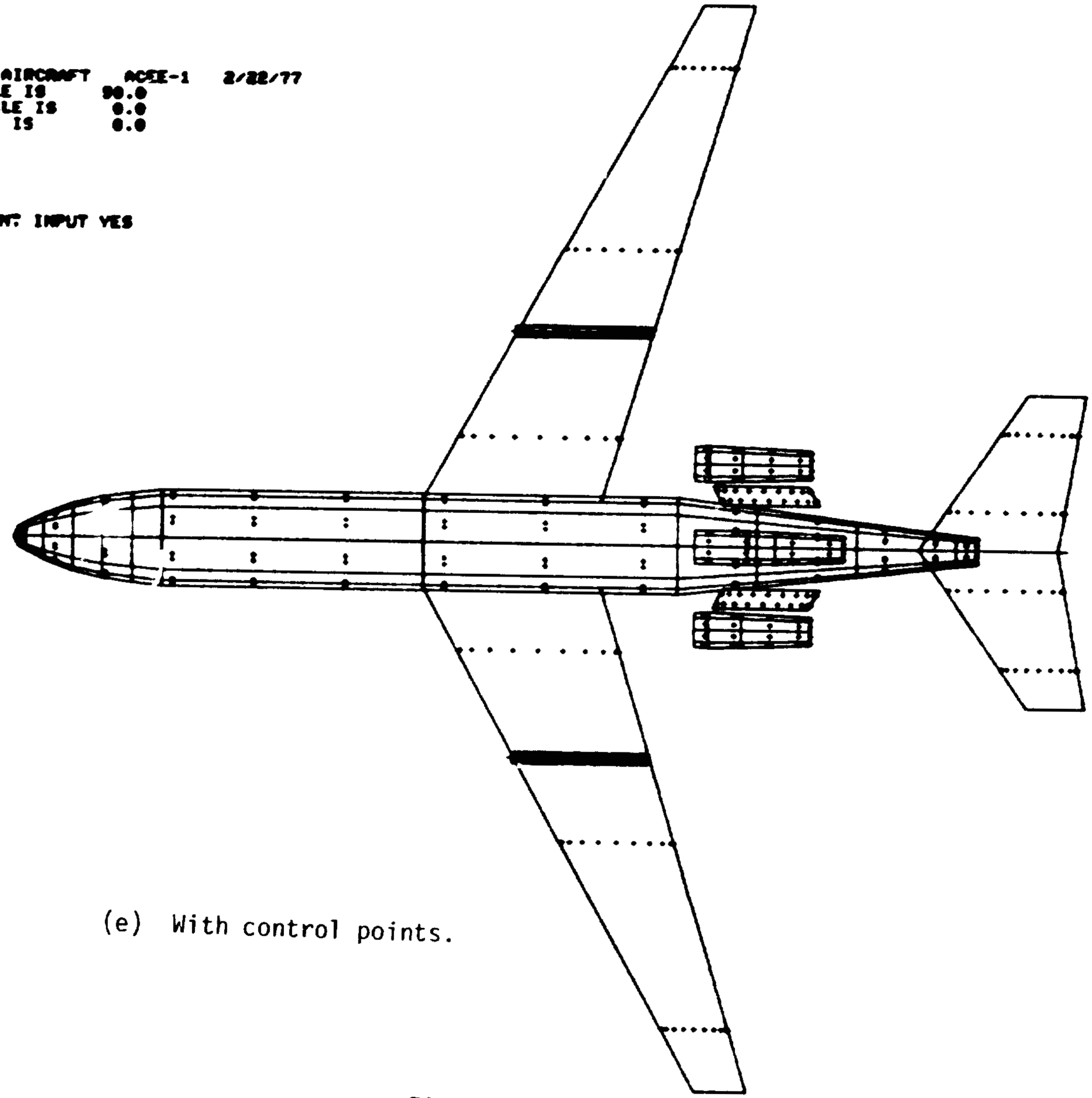
(c) Major panels only.

Figure 1. - Continued.



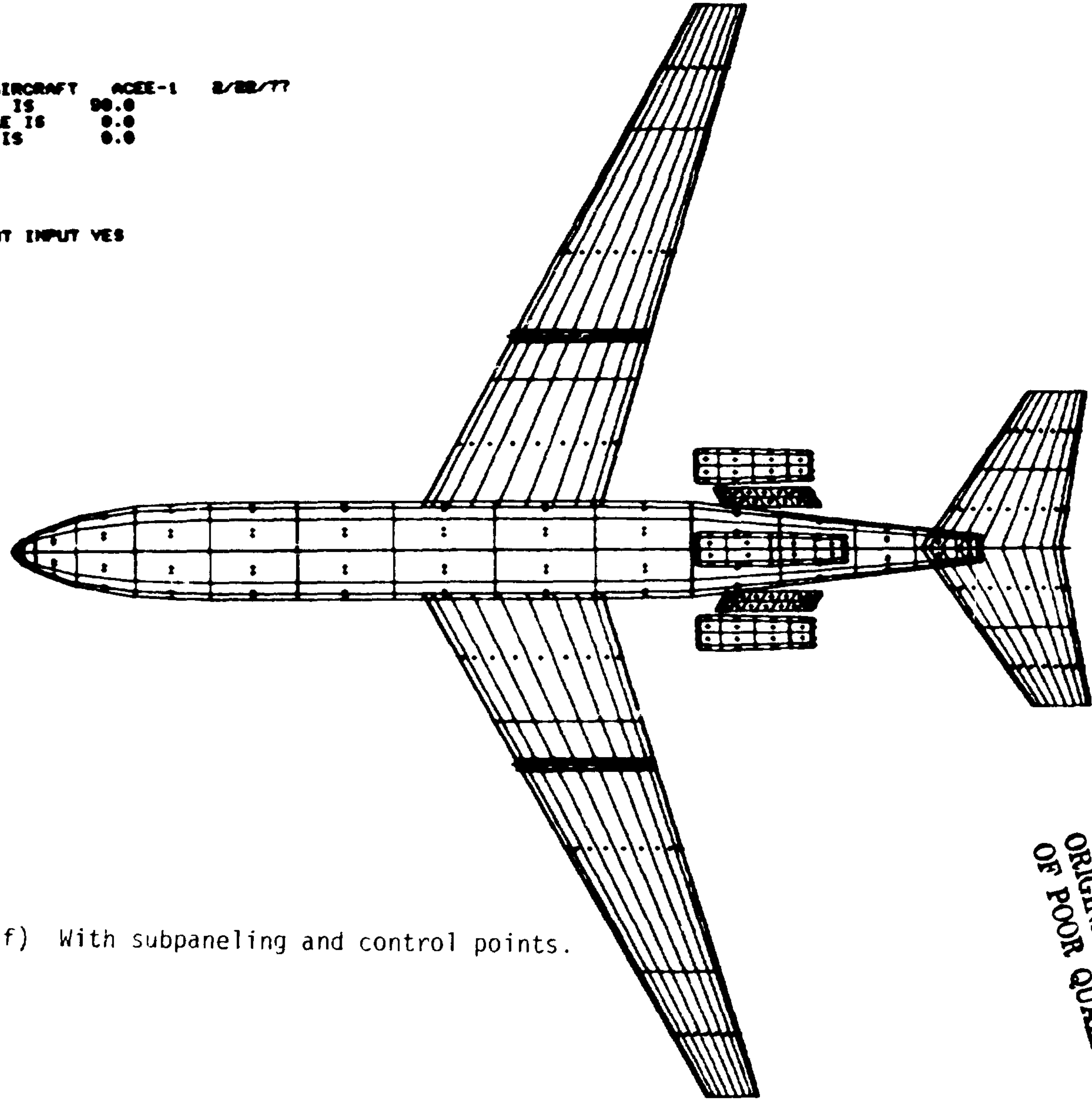
(d) With subpaneling.

Figure 1. - Continued.



(e) With control points.

Figure 1. - Continued.



(f) With subpaneling and control points.

Figure 1. - Concluded.

```
/GET,PLOTIT/UN=214737C
/CALL,PLOTIT(T=VORLAXX)
DO YOU WANT CAMBER IN THE PANELS ?
TRUE OR FALSE
? TRUE

 INPUT THE ROLL ANGLE FOR THE AIRCRAFT
 (DEG), >360 TO STOP.
? 11
PITCH ANGLE
? 12
YAW ANGLE
? 13
SUBPANELING ?  TRUE OR FALSE
? TRUE
CONTROL POINTS ? TRUE OR FALSE
? TRUE
```

Figure 2. - Sample execution of program PLOTIT.

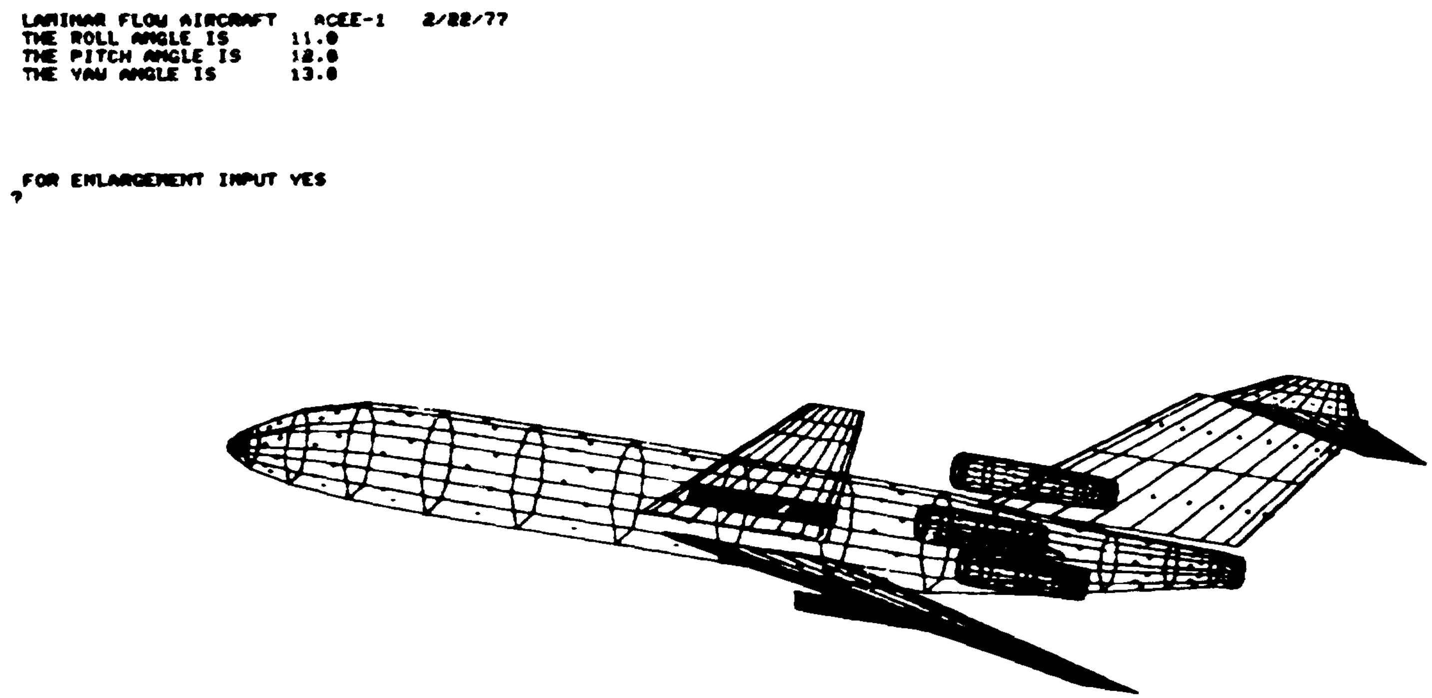
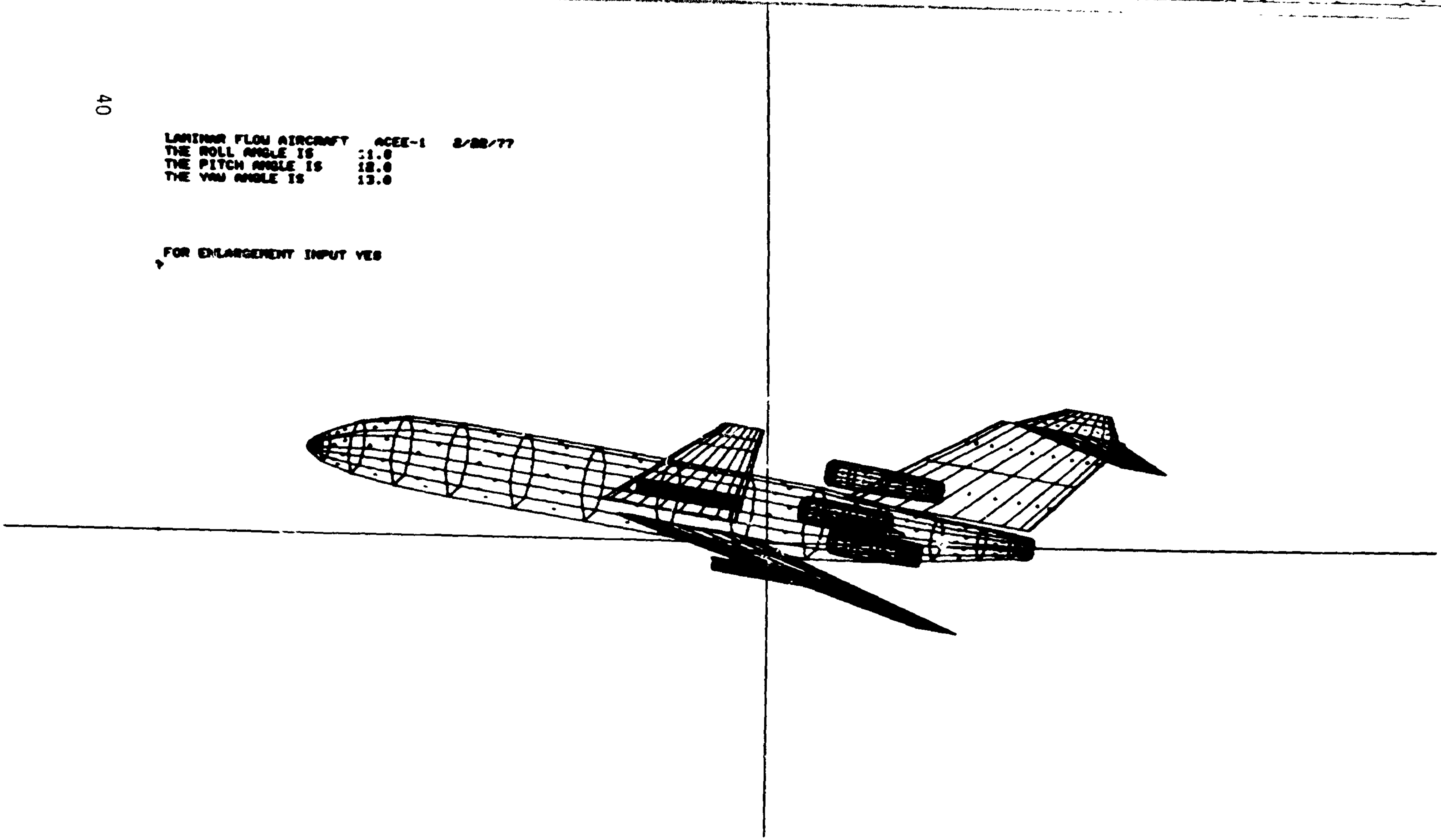
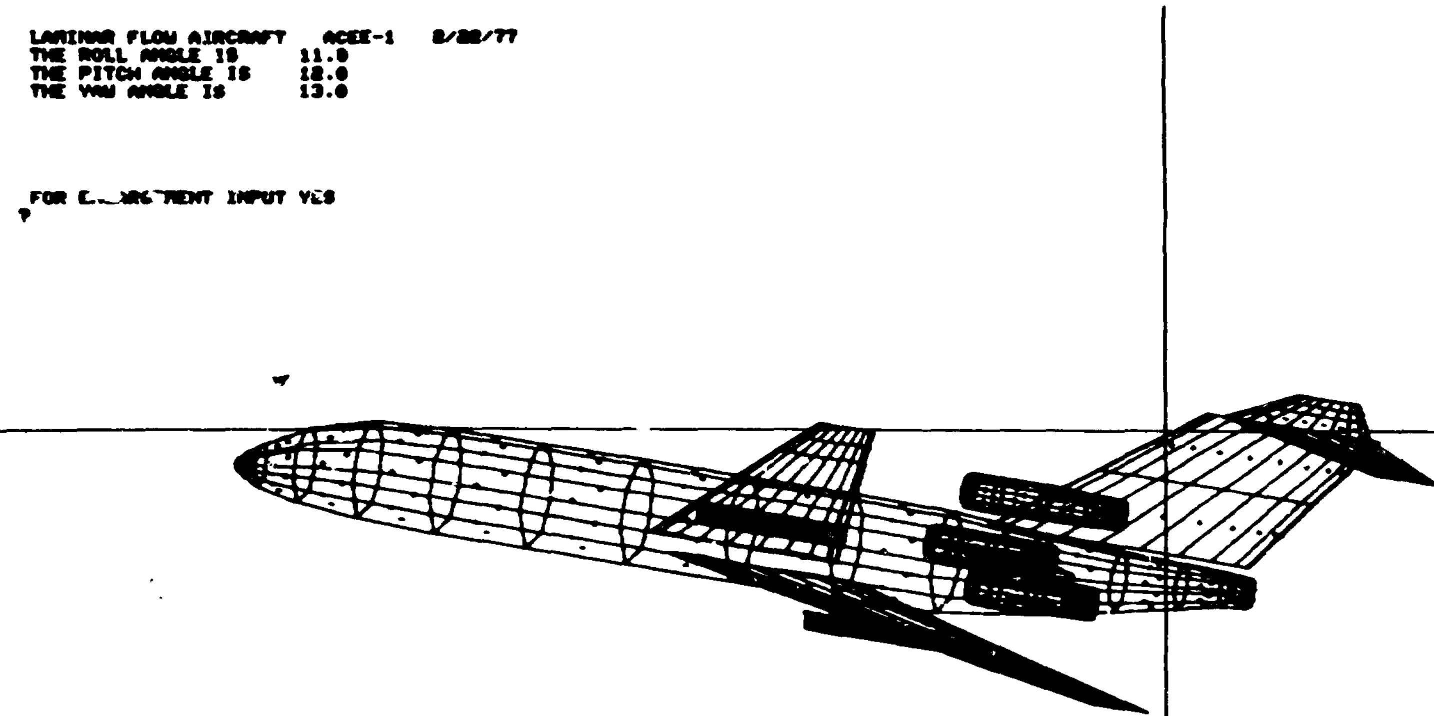


Figure 3. - Plot from sample execution in figure 2.



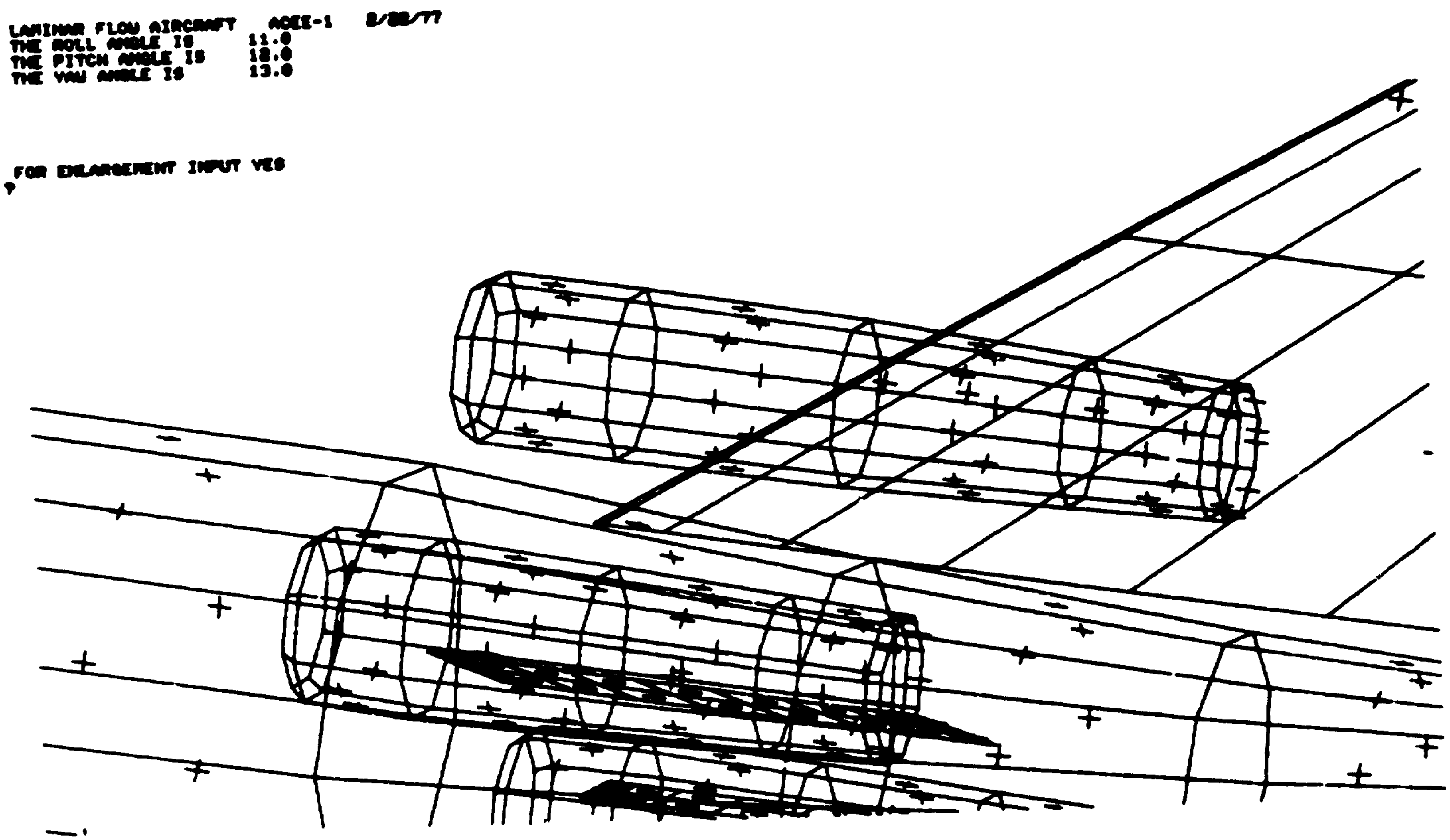
(a) First location of graphics cursor.

Figure 4. - Use of graphics cursor to enlarge a section of a configuration plot.



(b) Second location of graphics cursor.

Figure 4. - Continued.



(c) Enlarged plot resulting from cursor locations in figure 4 part (a) and part (b).

Figure 4. - Concluded.