NASA Technical Memorandum 82756

Interactive-Graphic Flowpath Plotting for Turbine Engines

Robert R. Corban
Lewis Research Center
Cleveland, Ohio

C21f

November 1981

NASA

INTERACTIVE-GRAPHIC FLOWPATH PLOTTING FOR TURBINE ENGINES

Robert R. Corban

National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio

SUMMARY

NASA-Lewis and the Navy jointly developed, based on a previous Navy code, an engine cycle program capable of simulating the design and off-design performance of arbitrary turbine engines. Boeing, under a NASA-Lewis contract, developed a computer code which, when used in conjunction with the cycle code, could predict the weight of the engines. Thus, the Navy/NASA Engine Program (NNEP) along with WATE-2 (Weight Analysis of Turbine Engines) determines the dimensions and weight of each major component in the engine. The output from these codes originally included a very crude and indecipherable computer drawing of the flowpath representation of the designed turbine engine. This proved to be of little use. It was desired to add a graphics subroutine to the code to enable the engineer to visualize the designed engine with more clarity by producing an overall view of the designed engine for output on a graphics device using IBM-370 graphics subroutines. In addition, with the engine drawn on a graphics screen, the program would allow for the interactive user to make changes to the inputs to WATE-2 for the engine to be redrawn and reweighed. These improvements would allow better use of the WATE-2 code in conjunction with NNEP.

INTRODUCTION

NNEP (Ref. 1) is a very general cycle analysis code that can assemble an arbitrary matrix of fans, turbines, ducts, shafts, etc. into a complete gas turbine engine and compute on- and off-design thermodynamic performance. WATE-2 (Ref. 2) is a preliminary design procedure for calculating engine weight using the component characteristics determined by NNEP. A main objective of the WATE-2 program using NNEP is to get a "feel" for what the total designed engine looks like in terms of size, weight, and its flowpath. Previously, the only output that could be examined as being the "total engine" was the flowpath configuration, shown in Figure 1, which was very hard to decipher. Development of a subprogram that incorporates the IBM-370 graphics package subroutines has now allowed the capability for a clearer output on a graphics device. The graphical output of the engine includes numbers on the components, a display of the total engine's weight, and an option of individual data output of the individual components which includes type, number of stages, and weight of the component, as shown in Figure 2.

DISCUSSION OF METHODS

Interactive Mode

In the past, if an engine was to be modified, the program running was terminated, input dataset was modified, and the program was rerun. Now available is the direct interactive creation and modification of the engine's inputs without exiting from the program. The interactive mode allows modification of any or all inputs to WATE-2 for revisions to the engine. The interactive mode is used in conjunction with the graphics subroutine (EGPLOT). Thus, the flag for graphics must be turned on in the weight code input (PLOT=T). The program will prompt the user on Unit 20 as to whether modifica-

tions are desired, thus Unit 20 must be identified to the computer as being the interactive terminal. Request of modifications will key the program to prompt the user for the input modifications. The new NAMELIST inputs entered on Unit 20 will over-write the previous inputs and proceed to perform the weight-code estimate.

Additions to the subroutine WTEST in the code WATE-2 are shown in Appendix C (marked with an arrow at each line in the subprogram that is new or has been modified).

Example Case

Appendix D contains a photocopy from an actual terminal session for interactively modifying a component for a more reasonable-designed engine. Shown in Figure 3, component Number 6 (low pressure compressor) is off-set to the fan (No. 2). The beginning of Appendix D shows the original WATE inputs, with the inputs to be changed marked. The H/T ratio of the compressor is increased (DESVAL (3,6)) and the mean diameter of duct 4 increased (DESVAL (3,4)). Thus, the newly designed engine is shown in Figure 4 with the compressor lined-up with the fan.

Non-Interactive Mode

The non-interactive mode allows the user to perform the WATE-2 program but not make any changes to the inputs. The code is written so that it will run as in the past, i.e., without the graphics subroutine, unless the user inputs a flag for graphics. Graphics is performed by putting PLOT=T in the weight code input (&W). PLOT=F will turn off the flag and the graphics will not be used. When executing the program in the non-conversational mode (BACK) the program and the procedure definition have been developed to perform with or without graphics. If plots are desired, then PLOT must be set to "True," as

above, but also an input to the procedure definition, WATE2IT, must be BACK=YES. The output of the plots will be on the Zeta12 plotter. If BACK is set to "YES" but the program is not executed in background, the program will still output plots on the Zeta12 plotter and will not prompt for any input.

Description of Subprogram

An explanation of the subprogram and subroutines is given below. A more thorough explanation of the graphics subroutines is given in Appendix A.

GRAPHICS SUBROUTINES

Line 102	GRINIT	Initializes graphics package
Line 103	CORNER	Removes corner marks on PLOT
Line 131	SETFRM	Sets no. of frames
Line 148	XAXIS	Produces the x-axis
Line 153	YAXIS	Produces the y-axis
Lines 164, 217, 248	GPLOT	Plots the given x and y array values
Line 186	GARC	Draws an arc or circle
Lines 260, 275-277, 287	NUMBER	Converts a number to an array of Printable graphics characters (EBCDIC)
Lines 261, 270, 278-282, 286, 288	CHARS	Positions printable characters (EBCDIC) anywhere on the graphics Display
Line 263	TITLE	Puts a title on the PLOT
Line 289	DISPLA	Displays the accumulated plot orders on the graphics device.

Program Description

Lines 105-135

This section of the program finds the maximum radius of the engine and scales the y-axis according to the maximum radius. The user is prompted, if in the conversation mode, for single or multiple frames and whether component output is desired on the plot. The x-axis is scaled to the engine's length and the x-axis is set according to the desired frame option.

Lines 136-153

Using the previous scaling information, the x- and y-axis are setup for the graphics output.

Lines 155-170

The centerline of the engine (radius=0) is created for the plot.

Lines 171-191

A curve for an arbitrary inlet is created using a generated arc. The slope of the line for the inner radius of the second flowpath component is matched and the arc generated from that point to the centerline. This part of the program is not necessarily needed but could help in visualizing the engine.

Lines 192-258

This section draws the basic components of the engine to be output on the graphics device. "Ducts" are drawn with a cosine function to create a "smoother" looking engine to allow for misalignment of components. This is the main section of the program. The components are drawn in the order of the flowpath (JFLOW) using output summary data from WATE-2.

Lines 259-271

The component numbers are put on the components to help in understanding the structure of the engine drawn. Also, the title of the engine is put on the plot.

Lines 272-290

If component output on the plot was requested, this section of the program will be performed. It converts data into a printable graphics character array that is put on the plots.

Lines 291-301

The total engine weight is put on the plot and then the final graphics output is displayed on the designated output graphics device.

CONCLUSIONS

The modifications to WATE-2 described in this report can help visualize needed changes to the engine's inputs and reduce time spent modifying the inputs. The user can interactively change inputs such as H/T ratio, aspect ratios of the blades, number of compressor or turbine stages, etc. to redraw the engine and see the effect on engine dimensions, flowpath and weight. The user can thus be assured that the final engine design is reasonable.

The logical structure developed for this code is adaptable to any other code in which the dimensions of the components are known.

NNEP and WATE-2 computer codes are available by written request to L. H. Fishbach, c/o NASA Lewis Research Center, 21000 Brookpark Road, Mail Stop 501-10, Cleveland, Ohio 44135. Requesters will be expected to input the necessary graphics subroutines for their system to replace the IBM 370 graphics subroutines indicated in this subprogram (EGPLOT).

REFERENCES

1. Fishbach, L. H. and Caddy, M. J.: NNEP - The Navy/NASA Engine Program. NASA TMX-71857, 1975.

2. Onat, E. and Klees, G. W.: A Method to Estimate Weight and Dimensions of Large and Small Gas Turbine Engines. NASA CR-159481, 1979.

ORIGINAL PAGE IS OF POOR QUALITY

```
CC
C CCCCCDDDDDDDDDNNNNNN
C      C                 NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
C      C
C      C
C      C
C      C
C      DDDDDDDDDD
C      CCCCCCCCCCCDDDDDDDDDDDDDDDDD
C  CCCC       CCCDDDDDDDDDDDDDDDDD                                                      TTTTTTTTTTTTTTTNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
CCCC   CCCCCCC                                                                           TTTTTT      T
C                                  CCCCCCCCCCCCCCCCCCCC    PPPPPPPPPPPPPP  TDDDDDDDDDD      TTTTT   T
                                   C              CCCCDDDDDDPPPPPPPPPPP  TTTT T              TTTD
                                   C       CCCCCCC                    PPTTTTTDDDDDDDDDD
                                   CCCCCCCC
                                   C

-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L-----C/L --------
```

Figure 1 "Old" flowpath configuration in output.

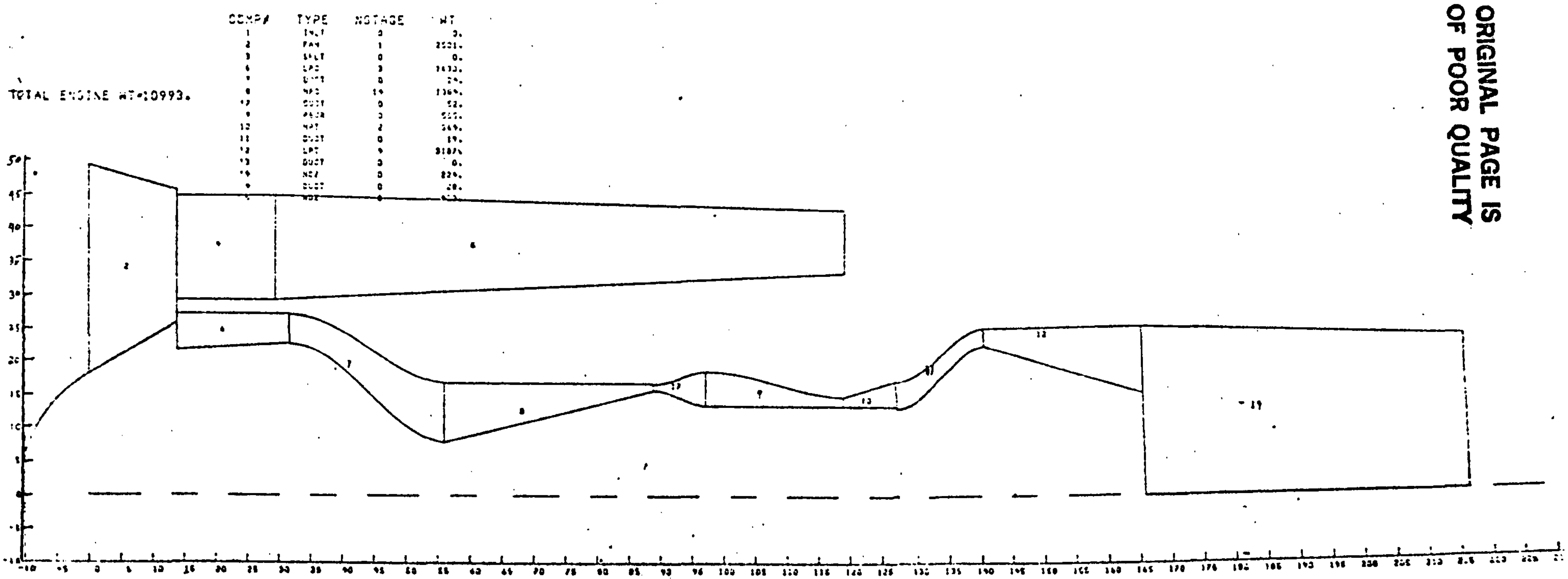
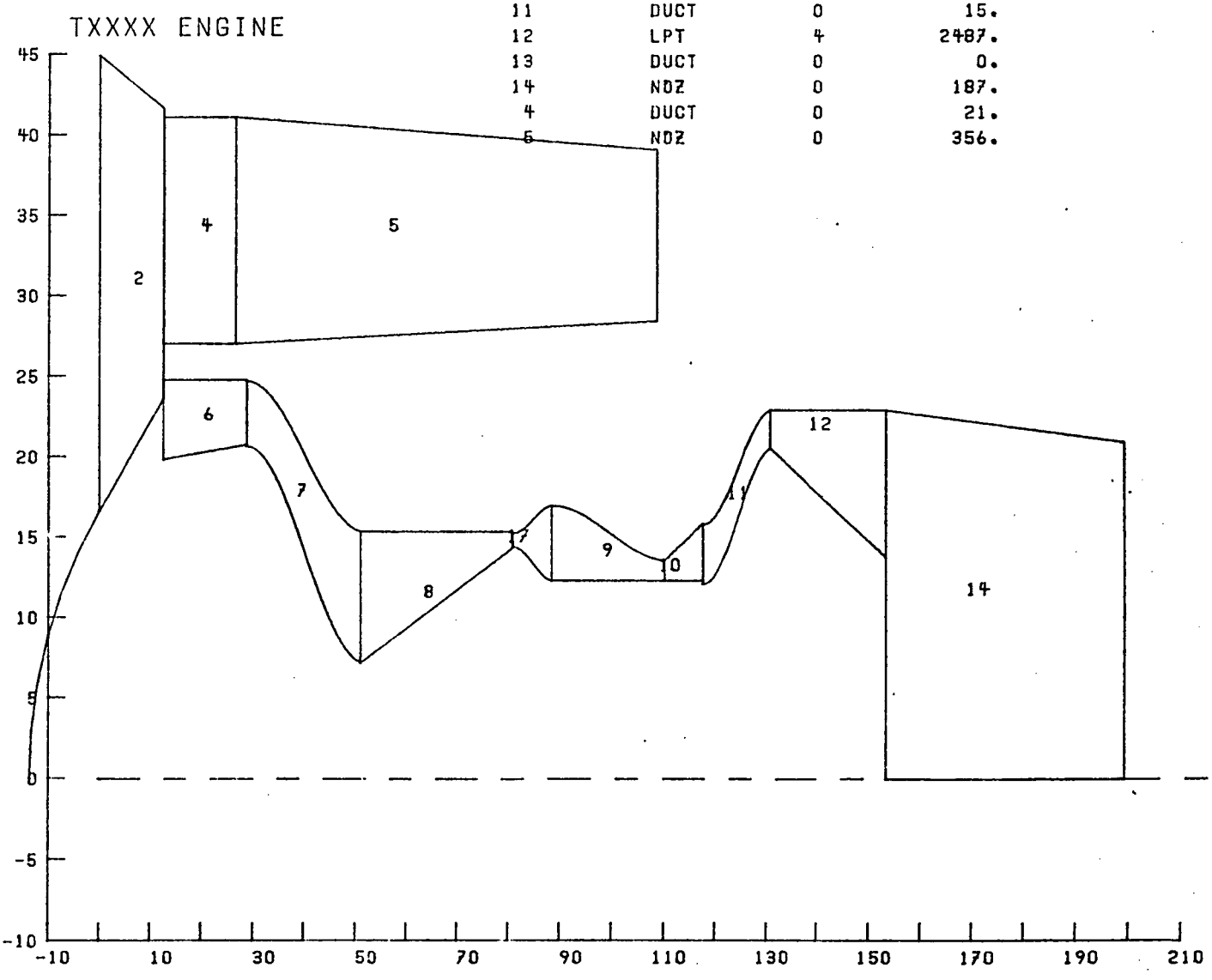


Figure 2 "New" graphics displayed flowpath representation (full length scale; reduced).

TOTAL ENGINE WT= 8749.

COMP#	TYPE	NSTAGE	WT
1	INLT	0	0.
2	FAN	1	2069.
3	SPLT	0	0.
6	LPC	3	1236.
7	DUCT	0	18.
8	HPC	14	1043.
17	DUCT	0	39.
9	PBUR	0	468.
10	HPT	2	278.
11	DUCT	0	15.
12	LPT	4	2487.
13	DUCT	0	0.
14	NOZ	0	187.
4	DUCT	0	21.
5	NOZ	0	356.



001

Figure 3 Sample engine before interactive changes (NOTE: Component 6).

COMP#	TYPE	NSTAGE	WT
1	INLT	0	0.
2	FAN	1	2069.
3	SPLT	0	0.
6	LPC	3	1490.
7	DUCT	0	20.
8	HPC	14	1043.
17	DUCT	0	39.
9	PBUR	0	468.
10	HPT	2	278.
11	DUCT	0	15.
12	LPT	3	1987.
13	DUCT	0	0.
14	NOZ	0	249.
4	DUCT	0	21.
5	NOZ	0	360.

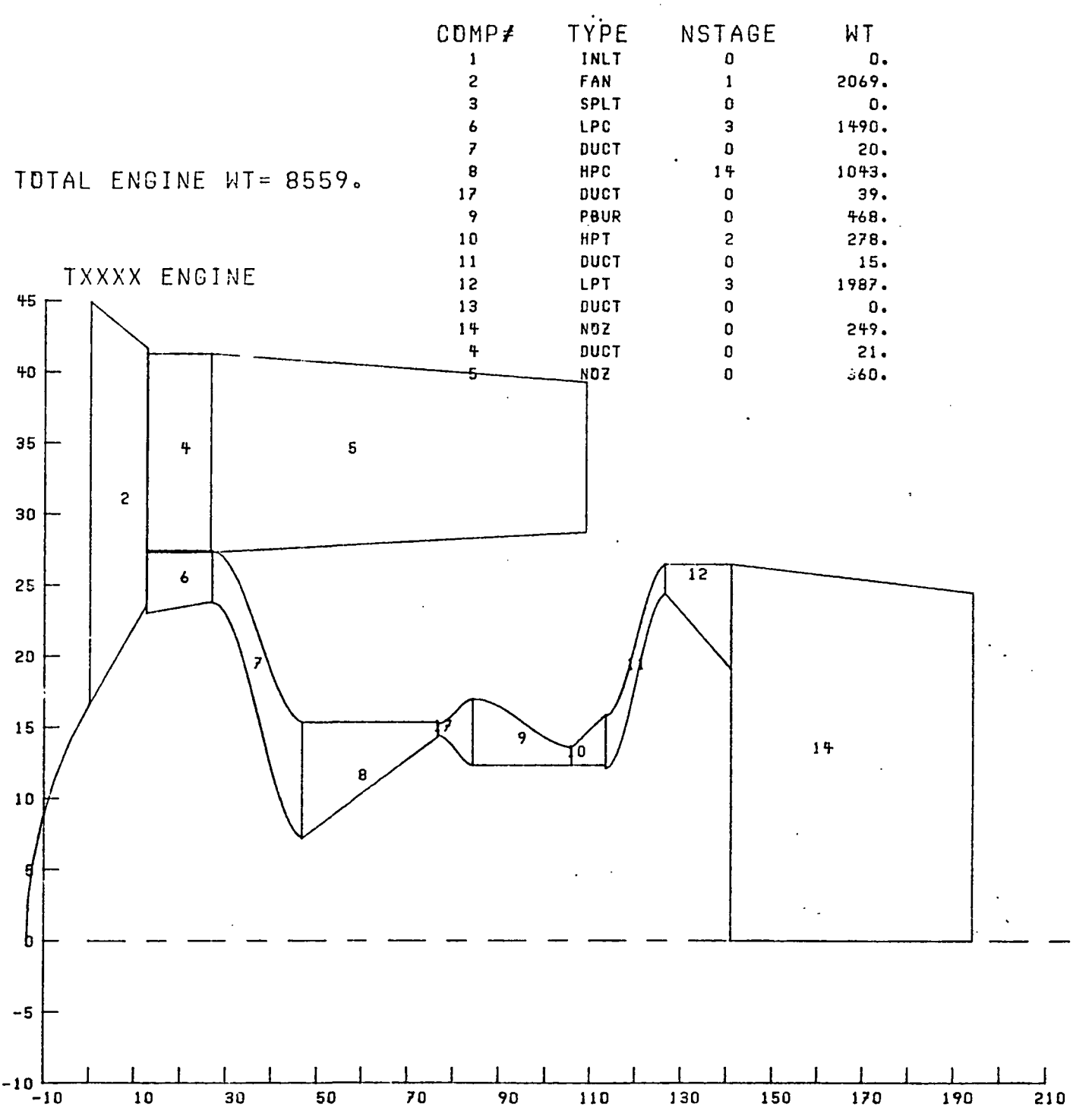


Figure 4 Sample engine after interactive modifications to inputs.

APPENDIX A

Subroutine EGPLOT User-Guide for Interactive Mode

The following user-guide pertains only to the IBM-370 computer system at NASA Lewis Research Center. However, this guide will show the steps taken when using the interactive mode for all systems. Use of the graphics devices mentioned and the set-up of the system's graphics package may differ from system to system.

The computer program is written so that it will run as in the past, i.e., without the graphics subroutine, unless you input a flag for graphics. This is done by putting PLOT=T in the &W input for the weight-code. PLOT=F will turn off the flag and the graphics will not be used.

When PLOT is set "TRUE" the graphics subroutine, EGPLOT, will be referenced. If you do not want all of the plots displayed, you will still have the option of graphics or no graphics. You will be prompted as to whether you still want the engine plotted. i.e., "DO YOU WANT A GRAPHICS PICTURE? YES = Y; NO = N." If you do not, then it will go on to the next scale factor (or terminate) without performing the graphics routine. If you request an engine graphics display, then it will use the graphics routine.

The graphics picture of the engine can be arranged in a few ways. You will be prompted by the program for the desired arrangement. The first option is whether you want multiple frames or not, i.e., "DO YOU WISH MULTIPLE FRAMES? YES = Y; NO = N." If you do not want multiple frames then you will get just one frame which has the components out of proportion with the x-axis condensed. The multiple frames gives a scaled picture with the engine in

proportion. Usually the single frame picture of the engine is sufficient for analysis of the engine, but multiple frames will give a better looking engine, especially on a Zeta or Calcomp plot. The user will also have the option as to whether component output is to be printed on the picture or not, i.e., "DO YOU WANT COMPONENT OUTPUT? YES = Y; NO = N." If the component output is desired, the component numbers along with its weight, type, and number of stages will be put on the engine plot. The total weight of the engine is still put on the plot even if the component output is not.

If a graphics device for output has not been previously defined, then it must be defined before the display of the engine can be performed. The system will prompt you for a device. Enter the desired output device. This device will stay the same for all plots until you "abend" or do a GSWITCH and redefine the output device.

Upon completion of the displayed engine, you will be prompted as to whether changes to the weight-code's inputs are desired, i.e., "DO YOU WISH TO MAKE CHANGES TO THE INPUT? YES = Y; NO = N." If using a graphics terminal for execution of the program and also for the graphics display, this question may come before initializing the graphics mode. If nothing is entered before displaying the picture, just enter the appropriate response after exiting from the graphics mode. If the bell has been turned off the terminal, backspace one and then enter the response. The program will go on with the weight-estimation program, if no changes are requested. If modifications to the inputs are requested, then the program will prompt you for the new NAMELIST inputs, i.e., "ENTER DESIRED CHANGES FROM TERMINAL, I.E., &W DESVAL (1,6) = *****, &END." Any number of changes can be made to the NAMELIST inputs. The new NAMELIST inputs will over-write the previous inputs and again carry-out the weight-estimation program.

When using offline graphics printing, such as the Calcomp or the Zeta plotters, GTERM must be entered (interactive mode) to initialize the batch task for offline printing to obtain the desired plots.

PROGRAM LISTING FOR SUBPROGRAM EGPLOT

```
C*****************************************************************************     1
C                                                                                   2
C SUBROUTINE EGPLOT                                                                 3
C ----------------                                                                  4
C                                                                                   5
C PURPOSE                                                                           6
C -------                                                                           7
C     TO MAKE A GRAPHICS PLOT OF THE TOTAL ENGINE SHOWING INDIVIDUAL                8
C     COMPONENTS ALONG THE FLOWPATH (JFLOW).                                        9
C                                                                                  10
C DESCRIPTION                                                                      11
C -----------                                                                      12
C     THE OVERALL VIEW OF THE DESIGNED ENGINE IS PRODUCED FOR OUTPUT ON            13
C     A GRAPHICS DEVICE USING IBM-370 GRAPHICS SUBROUTINES. IN THE                 14
C     INTERACTIVE MODE (BACK='NO',PLOT=T) THE USER IS PROMPTED FOR SIZE            15
C     OF THE PLOT (PROPORTIONAL LENGTH OR SINGLE FRAME) AND A REQUEST              16
C     FOR COMPONENT DATA OUTPUT ON THE PLOT. IN RUNNING THE PROGRAM IN             17
C     BACKGROUND(BACK),BACK='YES' MUST BE AN INPUT TO THE PROCDEF AND              18
C     PLOT=T SET. THE COMPONENTS ARE DRAW AS IT FOLLOWS THE FLOW PATH.             19
C USAGE                                                                            20
C -----                                                                            21
C     CALL EGPLOT(ENGLEN,WATENG)                                                   22
C                                                                                  23
C CALLING ROUTINES                                                                 24
C ----------------                                                                 25
C     WTEST---THE WEIGHT ESTIMATION ROUTINE                                        26
C                                                                                  27
C REQUIRED SUBROUTINES                                                             28
C --------------------                                                            29
C     >>>>>IBM-370 GRAPHICS PACKAGE SUBROUTINES<<<<<<                              30
C     GARC      - DRAWS AN ARC OR A CIRCLE                                         31
C     GRINIT    - INITIALIZES THE GRAPHICS PACKAGE                                 32
C     CORNER    - REMOVES CORNER MARKS FROM PLOT                                   33
C     SETFRM    - SETS THE NUMBER OF FRAMES FOR THE PLOT                           34
C     XAXIS     - CREATES A NONSTANDARD X-AXIS                                     35
C     YAXIS     - CREATES A NONSTANDARD Y-AXIS                                     36
C     GPLOT     - PLOTS DATA FROM ARRAYS                                           37
C     NUMBER    - CONVERTS A NUMBER TO CHARACTER DATA FORM (EBCDIC)                38
C     CHARS     - PLACES CHARACTER DATA ANYWHERE ON THE PLOT                       39
C     TITLE     - PRINTS AN AXIS OR PLOT TITLE                                     40
C     DISPLA    - DISPLAYS THE ACCUMULATED PLOT ORDERS                             41
C                 ON THE GRAPHICS DEVICE SELECTED                                  42
C                                                                                  43
C MODIFICATION HISTORY                                                             44
C --------------------                                                            45
C                                                                                  46
C       DATE     ID      ANALYST              DESCRIPTION                          47
C     -------- ------ -----------   ------------------------------------          48
C     MO/DA/YR IDENT  NAME          DESCRIPTION OF CHANGES                         49
C                                                                                  50
C AUTHOR/LANGUAGE/DATE                                                             51
C --------------------                                                            52
C     ROBERT CORBAN-COOP (U. OF CINCINNATI)  /FORTRAN IV / SEPT 21,1981            53
C                                                                                  54
C GLOSSARY                                                                         55
C --------                                                                         56
C     NAME    ORIGIN USAGE              DESCRIPTION                              57
C     ----    ------ -----  ------------------------------------------           58
C     IWMEC   /WMECH/ I     CONTROL INFORMATION                                    59
C     WATE    /WMECH/ I     WEIGHT OF EACH COMPONENT                               60
```

```
C      ALENG    /WMECH/  I      ACTUAL LENGTH OF EACH COMPONENT                  61
C      TLENG    /WMECH/  I      ACCUMULATED LENGTH TO END OF COMPONENT            62
C      RI       /WMECH/  I      RADIUS INNER INLET,OUTLET EACH STATION           63
C      RO       /WMECH/  I      RADIUS OUTER INLET,OUTLET EACH STATION           64
C      ENGLEN      ARG   I      TOTAL ENGINE LENGTH                              65
C      WATENG      ARG   I      TOTAL ENGINE WEIGHT                              66
C      VARS              L      ARRAY OF DATA FOR DESIRED AXIS                   67
C      IVARS             L      ARRAY OF CONTROL INFO. FOR LINE DRAWING          68
C      ITICKS            L      NO. OF TICK MARKS ON THE AXIS                    69
C      NFRAM             L      NO. OF FRAMES FOR GRAPHICS OUTPUT                70
C      AMAX              L      MAXIMUM RADIUS OF ALL COMPONENTS                 71
C      SKIPIT            I      BACKGROUND FLAG (BACK=YES),SKIP PROMPTS          72
C                                                                                 73
C                                                                                 74
       SUBROUTINE EGPLOT(ENGLEN,WATENG)                                           75
       REAL*8 DATINP,DATOUT,WTF,TOPRES,TOTEMP,FAR,CORFLO,VMACH,STATP,-            76
      1ERROR,TOL,TOLT,TOLTT,DEPV,DTOL,PERPF,RPMNT,TMTEMP,TMPRES,DATOUM,-          77
      2DATMAC,DATALT,DESLIM,TNPRES,TNTEMP,CNRFLO,CORFLM,WNTF,DATOUN,FARN,-        78
      3DANINP,DEBUG,DEPQ,SELAST,DD,TOLOPT,M                                      79
C               ******************                                                80
C               * COMMON BLOCKS *                                                81
C               ******************                                                82
       COMMON /DBL/ DATINP(15,60),DATOUT(9,60),WTF(40),TOPRES(40),TOTEMP-         83
      1(40),FAR(40),CORFLO(40),VMACH(40),STATP(40),ERROR(40),TOL,TOLT-            84
      2,TOLTT,DEPV(20),DTOL(20),PERPF(20)                                         85
       COMMON /SNGL/ JM1,JM2,JP1,JP2,JCX,LOCTBL(9,60),JCOMP(70),IWAY,NIT,-        86
      1ITAB(70),JCONF(60,4),JTYPE(60),JFLOW(70),IDEDAP(15),KKINDS(14,25),-        87
      2NCOMP,NOSTAT,NITER,NFINIS,NPASS,JCC,NTBL,NCTS,JCIND(20),JCDEP(20),-        88
      3JCVIND(20),JCVDEP(20),KDTYP(20),IDONE(60)                                  89
       COMMON /WMECH/ IWMEC(7,60),WATE(60),ALENG(60),TLENG(40),RI(2,40),-         90
      1RO(2,40),DESVAL(17,60),DSHAF(5),RPMT(60),IWT,IPLT,IERR,ISII,ISIO,-         91
      2IOUTCD,NSTAG(60)                                                           92
       COMMON /SKIP/ SKIPIT                                                       93
       DIMENSION VARS(10),X(30),IVARS(10),Y(30),SY(5),IS(60),ANGLE(3)             94
       DIMENSION DATA(7),WEIGHT(4),XS(60),YS(60)                                  95
       DATA DATA/4HCOMP,4H#   ,4HTYPE,4H   N,4HSTAG,4HE   ,4H WT /                96
       DATA WEIGHT/4HTOTA,4HL EN,4HGINE,4H WT=/                                   97
       DATA INLT,YES,COMP/4HINLT,2*4HY   /                                        98
       DOUBLE PRECISION PLOT3,PLOT4                                               99
       LOGICAL SKIPIT                                                            100
       IWMEC(1,1)=INLT                                                           101
       CALL GRINIT                                                               102
       CALL CORNER(1)                                                            103
C                                                                                104
C     FINDS MAXIMUM RADIUS OF ENGINE (AMAX)                                      105
C                                                                                106
       AMAX=0.0                                                                  107
       MAIN=0                                                                    108
       PHI=3.1415927                                                             109
       DO 20 I=1,2                                                               110
       DO 10 J=1,40                                                              111
       IF(RO(I,J).GT.AMAX) AMAX=RO(I,J)                                          112
  10   CONTINUE                                                                  113
  20   CONTINUE                                                                  114
C                                                                                115
C     PROMPTS FOR TYPE OF PLOT                                                   116
C                                                                                117
       ISTORE=AMAX/5.                                                            118
       YMAX=ISTORE*5.+5.0                                                        119
       N=YMAX/5.+2.                                                              120
```

```
      NFRAM=1                                                          121
      IF (SKIPIT) GO TO 30                                             122
      WRITE(20,500)                                                    123
      READ(20,600) FRAME                                               124
      WRITE(20,700)                                                    125
      READ(20,600)  COMP                                               126
      IF(FRAME.NE.YES) GO TO 30                                        127
      ITICKS=ENGLEN/5.+6                                               128
      IFRAM=N/7.*9.                                                    129
      NFRAM=ITICKS/IFRAM                                               130
      CALL SETFRM(NFRAM)                                               131
      GO TO 40                                                         132
   30 ITICKS=ENGLEN/10.+3                                              133
      IODD=ITICKS/2*2                                                  134
      IF(IODD.NE.ITICKS)ITICKS=ITICKS+1                                135
C                                                                      136
C     SET-UP OF X AND Y AXIS                                           137
C                                                                      138
   40 VARS(1)=7.                                                       139
      VARS(2)=9.*NFRAM                                                 140
      VARS(3)=0.                                                       141
      VARS(4)=-10.                                                     142
      VARS(5)=(ITICKS-1)*10.                                           143
      IF(FRAME.EQ.YES) VARS(5)=(ITICKS-2)*5.                           144
      XMAX=VARS(5)                                                     145
      VARS(6)=ITICKS                                                   146
      VARS(7)=2.                                                       147
      CALL XAXIS(0.5,0.5,VARS)                                         148
      VARS(2)=7.                                                       149
      VARS(3)=90.                                                      150
      VARS(5)=YMAX                                                     151
      VARS(6)=N                                                        152
      CALL YAXIS(0.5,0.5,VARS)                                         153
C                                                                      154
C     CREATION OF THE CENTERLINE FOR THE ENGINE (-- ----- -- -----)    155
C                                                                      156
      N=ITICKS-2                                                       157
      X(1)=0.0                                                         158
      X(2)=8.0                                                         159
      Y(1)=0.0                                                         160
      Y(2)=0.0                                                         161
      IVARS(1)=3                                                       162
      IVARS(2)=2                                                       163
      DO 50 I=1,N                                                      164
      CALL GPLOT(X,Y,IVARS)                                            165
      X(1)=X(2)+5.                                                     166
      J=I/2                                                            167
      IF(I.EQ.2*J) X(2)=X(2)+13.                                       168
      IF(I.NE.2*J) X(2)=X(2)+9.                                        169
   50 CONTINUE                                                         170
C                                                                      171
C     CREATE AN ARBITRARY INLET USING A GENERATED ARC                  172
C                                                                      173
      I=JFLOW(2)                                                       174
      IUP1=JCONF(I,1)                                                  175
      IDN1=JCONF(I,3)                                                  176
      Y0=RI(1,IUP1)*7.0/(YMAX+10.)                                     177
      A=(RI(2,IDN1)-RI(1,IUP1))*7.0/(YMAX+10.)                         178
      B=9*NFRAM/(XMAX+10.)*ALENG(2)                                    179
      M=A/B                                                            180
```

```
      DEGREE=PHI/2.-ATAN(M)                                                  181
      IF(DEGREE.GT.PHI/2.) DEGREE=PHI/2.                                     182
      RAD=YO/SIN(DEGREE)                                                     183
      CENTX=SQRT(RAD**2-YO**2)+0.5+10.*9*NFRAM/(XMAX+10.)                    184
      CENTY=10.*7.0/(YMAX+10.)+0.5                                           185
      ANGLE(1)=0.0                                                           186
      ANGLE(2)=90.+(PHI/2.-DEGREE)*(180./PHI)                                187
      ANGLE(3)=180.-ANGLE(2)                                                 188
      CALL GARC(CENTX,CENTY,RAD,ANGLE,2)                                     189
C                                                                            190
C     COMPONENTS ARE DRAWN FROM SUMMARY DATA (RI AND RO)                     191
C                                                                            192
      DO 170 II=1,60                                                         193
      I=JFLOW(II)                                                            194
      NC=JTYPE(I)                                                            195
      IF(I.EQ.0) GO TO 180                                                   196
      IUP1=JCONF(I,1)                                                        197
      IDN1=JCONF(I,3)                                                        198
      X(1)=TLENG(IDN1)-ALENG(I)                                              199
      X(2)=X(1)                                                              200
      SY(2)=RO(1,IUP1)                                                       201
      SY(3)=RO(2,IDN1)                                                       202
      IF(NC.EQ.8) SY(2)=SY(3)                                                203
      IF(MAIN.EQ.1 .AND. NC.EQ.9) GO TO 80                                   204
      IF(NC.EQ.9) MAIN=1                                                     205
      SY(1)=RI(1,IUP1)                                                       206
      SY(4)=RI(2,IDN1)                                                       207
      GO TO 90                                                               208
   80 AREA=DATOUT(5,I)                                                       209
      SY(1)=RI(2,ISTOR)                                                      210
      SY(4)=SQRT(SY(3)*SY(3)-AREA/PHI)                                       211
   90 AINC=ALENG(I)                                                          212
      ISTOR=IDN1                                                             213
      IVARS(2)=2                                                             214
      N=4                                                                    215
      IF(NC.EQ.2) N=1                                                        216
      DO 100 IP=1,N                                                          217
      Y(1)=SY(IP)                                                            218
      Y(2)=SY(IP+1)                                                          219
      IF(IP.EQ.4) Y(2)=SY(1)                                                 220
      CALL GPLOT(X,Y,IVARS)                                                  221
      X(1)=X(2)                                                              222
      X(2)=X(2)+AINC                                                         223
      AINC=AINC-ALENG(I)                                                     224
  100 CONTINUE                                                               225
      IF(NC.NE.2) GO TO 160                                                  226
C                                                                            227
C     CREATE A SMOOTH DUCT USING THE COSINE FUNCTION                         228
C                                                                            229
      JJ=JFLOW(II+1)                                                         230
  110 CONTINUE                                                               231
      IF(ALENG(JJ).NE.0.0 .OR. JJ.EQ.60 .OR. JTYPE(JJ).EQ.8) GO TO 120       232
      JJ=JJ+1                                                                233
      GO TO 110                                                              234
  120 IUP1=JCONF(JJ,1)                                                       235
      IUP2=JCONF(JJ,2)                                                       236
      SY(3)=RI(1,IUP1)                                                       237
      IF(JTYPE(JJ).EQ.9) SY(3)=RI(2,IDN1)                                    238
      SY(4)=RO(1,IUP1)                                                       239
      IF(JTYPE(JJ).EQ.8) SY(3)=RI(1,IUP2)                                    240
```

```
      IF(JTYPE(JJ).EQ.8) SY(4)=RO(1,IUP2)                                241
      DO 150 IJ=1,2                                                      242
      DIFF=SY(IJ+2)-SY(IJ)                                               243
      YO=SY(IJ)+DIFF/2.                                                  244
      YMAX=ABS(DIFF/2.)                                                  245
      THEATA=0.0                                                         246
      IF(DIFF.GT.0.0) THEATA=PHI                                         247
      DO 140 IK=1,30                                                     248
      Y(IK)=YMAX*COS(THEATA)+YO                                          249
      IF(IK.EQ.1)GO TO 130                                               250
      X(IK)=X(IK-1)+ALENG(I)/29.                                         251
  130 THEATA=THEATA+0.1047198                                            252
  140 CONTINUE                                                           253
      IVARS(2)=30                                                        254
      CALL GPLOT(X,Y,IVARS)                                              255
      IF(IJ.EQ.1) YSAVE=Y(15)                                            256
  150 CONTINUE                                                           257
C                                                                        258
C     LABEL COMPONENTS BY COMPONENT NUMBER (JFLOW)                       259
C                                                                        260
  160 Y1=(SY(1)+SY(2))/2.                                                261
      IF(NC.EQ.2) Y1=(Y(15)+YSAVE)/2.                                    262
      X1=TLENG(IDN1)-ALENG(I)/1.5                                        263
      IF(ALENG(I).LE.10.0 .AND. FRAME.NE.YES) X1=X1-ALENG(I)*0.5         264
      IF(ALENG(I).EQ.0.0) GO TO 170                                      265
      IF(X1.GT.ENGLEN) X1=TLENG(IDN1)-ALENG(I)*0.95                      266
      CALL NUMBER(1,I,2,0,NUMS)                                          267
      CALL CHARS(-2,NUMS,0.0,X1,Y1,10)                                   268
  170 CONTINUE                                                           269
  180 CALL TITLE(1,60,15,IDEDAP)                                         270
C                                                                        271
C     COMPONENT OUTPUT DATA IS PLACED ON THE PLOT,IF REQUESTED(COMP=YES) 272
C                                                                        273
      IF(COMP.NE.YES) GO TO 200                                          274
      X2=4.0                                                             275
      YY=9.8                                                             276
      CALL CHARS(28,DATA,0.,X2,9.8,15)                                   277
      DO 190 III=1,60                                                    278
      I=JFLOW(III)                                                       279
      IF(I.EQ.0) GO TO 200                                               280
      YY=YY-.2                                                           281
      CALL NUMBER(4,WATE(I),6,0,PLOT3)                                   282
      CALL NUMBER(1,I,2,0,PLOT1)                                         283
      CALL NUMBER(1,NSTAG(I),2,0,PLOT2)                                  284
      CALL CHARS(2,PLOT1,0.,4.2,YY,10)                                   285
      CALL CHARS(4,IWMEC(1,I),0.,5.3,YY,10)                              286
      CALL CHARS(2,PLOT2,0.,6.5,YY,10)                                   287
      CALL CHARS(6,PLOT3,0.,7.5,YY,10)                                   288
  190 CONTINUE                                                           289
C                                                                        290
C     WEIGHT OF TOTAL ENGINE IS PLACED ON THE PLOT; PLOT IS DISPLAYED    291
C                                                                        292
  200 CALL CHARS(16,WEIGHT,0.,0.2,8.5,15)                                293
      CALL NUMBER(4,WATENG,6,0,PLOT4)                                    294
      CALL CHARS(6,PLOT4,0.,2.5,8.5,15)                                  295
      CALL DISPLA(1)                                                     296
  500 FORMAT ('  DO YOU WISH MULTIPLE FRAMES? YES=Y ; NO=N')             297
  600 FORMAT(A4)                                                         298
  700 FORMAT ('  DO YOU WANT COMPONENT OUTPUT? YES=Y ; NO=N')             299
      RETURN                                                             300
      END                                                                301
```

APPENDIX C

PROGRAM LISTING FOR REVISED SUBPROGRAM WTEST

```
C*****************************************************************************
C
C SUBROUTINE WTEST
C ----------------
C
C PURPOSE
C -------
C     TO CONTROL THE CALLING OF SUBROUTINES WHICH WILL ESTIMATE THE
C     WEIGHT AND LENGTH OF INDIVIDUAL COMPONENTS
C
C DESCRIPTION
C -----------
C     THE OVERALL LENGTH OF THE ENGINE IS CALCULATED BY PROCESSING THE
C     ILENG ARRAY.ALL COMPONENTS EXCEPT DUCTS AND SHAFTS, THEN DUCTS.
C     THE REMAINING COMPONENTS EXCEPT DUCTS AND SHAFTS ARE PROCESSED.
C     THE DUCTS ARE PROCESSED AND FINIALY THE SHAFTS.
C     A BUILT-IN ASSUMPTION IN THE DUCT ROUTINE IS THAT NO DUCT IS
C     CONNECTED TO ANOTHER DUCT I.E. THE DUCT SIZE IS DETERMINED BY THE
C     ADJOINING COMPONENTS.
C     THEN THE MAXIMUM RADIUS IS FOUND. THEN DEPENDING ON THE PRINT
C     FLAG -IOUTCD- THE REQUIRED PRINTING IS DONE
C     IF THE PLOT CODE FLAG -IPLT- IS TRUE ROUTINE EPLT IS CALLED
C USAGE
C -----
C     CALL WTEST
C
C CALLING ROUTINES
C ----------------
C     FLOCAL-
C     ZTOPZ -
C
C REQUIRED SUBROUTINES
C --------------------
C     COMP    -COMPRESSOR WEIGHT/LENGTH
C     TURB    -TURBINE
C     SHAFT   -SHAFT
C     DUCTW   -DUCT
C     COMBWT  -PRIMARY BURNER WEIGHT/LENGTH
C     WTNOZ   -NOZZLE     WEIGHT/LENGTH
C     WMIXR   -MIXER
C     WSPLT   -SPLITTER
C     EPLT    -PRINTER/PLOTTER
C
C MODIFICATION HISTORY
C --------------------
C
C       DATE     ID      ANALYST              DESCRIPTION
C     -------- ------ -----------   ------------------------------------
C     MO/DA/YR IDENT  NAME          DESCRIPTION OF CHANGES
C   9/21/81          ROBERT CORBAN   INCORPORATE INTERACTIVE MODE AND GRAPHICS
C
C AUTHOR/LANGUAGE/DATE
C --------------------
C     NORMAN PREWITT-BOEING COMPUTER SERV.  /FORTRAN IV    / OCT 10,1976
C
C GLOSSARY
C --------
C     NAME    ORIGIN USAGE              DESCRIPTION
C     ----    ------ -----  ------------------------------------------
C     IWMEC   /WMECH/ I     CONTROL INFORMATION
```

```
C      WATE     /WMECH/ O      WEIGHT OF EACH COMPONENT                          61
C      ALENG    /WMECH/ O      ACTUAL LENGTH OF EACH COMPONENT                   62
C      TLENG    /WMECH/ O      ACCUMULATED LENGTH TO END OF COMPONENT            63
C      RI       /WMECH/ O      RADIUS INNER INLET,OUTLET EACH STATION            64
C      RO       /WMECH/ O      RADIUS OUTER INLET,OUTLET EACH STATION            65
C      DESVAL   /WMECH/ I      MECHANICAL DESIGN DATA OVERRIDES DEFAUL           66
C      DSHAF    /WMECH/ O      SHAFT DIAMETER INNER TO OUTER                     67
C      RPMT     /WMECH/ I      ACTUAL COMPONENT RPM                              68
C      IWT      /WMECH/ I      WEIGHT ESTIMATION FLAG TRU=  DO IT                69
C      IPLT     /WMECH/ I      PLOTTER FLAG TRUE= DO IT                          70
C      PLOT             L      GRAPHICS PLOTTING FLAG  TRUE= DO IT               71<-
C      IERR     /WMECH/ O      ERROR FLAG                                        72
C      ISIO     /WMECH/ I      OUTPUT UNITS 0=ENGLISH, 0 SI                      73
C      ISII     /WMECH/ I      INPUT UNITS 0=ENGLISH, 0 SI                       74
C      IOUTCD   /WMECH/ I      PRINT FLAG 0=SUMMARY, 1=GENERAL,2=DIAGNOSTIC      75
C      ILENG    /WMECH/ I      COMPONENTS CONTRIBUTING TO OVERALL LENGTH         76
C      IDID             O      FLAG = 0 COMPONENT NOT YET WEIGHED  =1 YES        77
C      SKIPIT           L      BACKGROUND FLAG YES=SKIP PROMPTS                  78<-
       SUBROUTINE WTEST                                                         79
       REAL*8 DATINP,DATOUT,WTF,TOPRES,TOTEMP,FAR,CORFLO,VMACH,STATP,ERRO-      80
      1R,TOL,TOLT,TOLTT,DEPV,DTOL,PERPF,RPMNT,TMTEMP,TMPRES,DATOUM,DATMAC-      81
      2,DATALT,DESLIM,TNPRES,TNTEMP,CNRFLO,CORFLM,WNTF,DATOUN,FARN,DANINP-      82
      3,DEBUG,DEPQ,SELAST,DD,TOLOPT                                             83
C               ******************                                              84
C               * COMMON BLOCKS *                                              85
C               ******************                                              86
       COMMON /DBL/ DATINP(15,60),DATOUT(9,60),WTF(40),TOPRES(40),TOTEMP(-      87
      140),FAR(40),CORFLO(40),VMACH(40),STATP(40),ERROR(40),TOL,TOLT,TOLT-      88
      2T,DEPV(20),DTOL(20),PERPF(20)                                            89
       COMMON /SNGL/ JM1,JM2,JP1,JP2,JCX,LOCTBL(9,60),JCOMP(70),IWAY,NIT,-      90
      1ITAB(70),JCONF(60,4),JTYPE(60),JFLOW(70),IDEDAP(15),KKINDS(14,25),-      91
      2NCOMP,NOSTAT,NITER,NFINIS,NPASS,JCC,NTBL,NCTS,JCIND(20),JCDEP(20),-      92
      3JCVIND(20),JCVDEP(20),KDTYP(20),IDONE(60)                                93
       COMMON /DEFAUL/ DEFAUL(15,20),ISCALE(3),SCALE(6)                         94
       COMMON /WMECH/ IWMEC(7,60),WATE(60),ALENG(60),TLENG(40),RI(2,40),R-      95
      1O(2,40),DESVAL(17,60),DSHAF(5),RPMT(60),IWT,IPLT,IERR,ISII,ISIO,IO-      96
      2UTCD,NSTAG(60)                                                           97
       COMMON /CONVER/ CONVER(15)                                               98
       COMMON /NEPOPT/ DEBUG,DEPQ,SELAST,DD,TOLOPT,NDSET,NPARTS,IOPTP,NPA-      99
      1SSO,NVOPT,NJOPT,NOPT                                                    100
       COMMON /TERMON/ TNPRES(40),TNTEMP(40),CNRFLO(40),WNTF(40),RPMNT(40-      101
      1),DATOUN(9,60),FARN(40),DANINP(15,60),TMTEMP(40),TMPRES(40),CORFLM-      102
      2(40),DATOUM(9,60),DATMAC(4,60),DATALT(4,60),DESLIM(15)                   103
       COMMON /CENTER/ CGARM(60)                                                104
       COMMON /ARM/ ACCARM                                                      105
       COMMON /DISKK/ DISKWI,ENGINE                                             106
       COMMON /SKIP/ SKIPIT                                                     107<-
C                                                                               108
C               ************************                                        109
C               * DATA STORAGE DEFINITION *                                    110
C               ************************                                        111
       LOGICAL PINP,IPLT,ISIO,ISII,PLOT,SKIPIT                                  112<-
       INTEGER IDID(60),ILENG(40)                                               113
       DIMENSION NUMNUM(17), IRNAME(17), CORFLC(40)                             114
       NAMELIST / W/IWMEC,DESVAL,ACCS,IWT,IPLT,ISII,ISIO,IOUTCD,ILENG,DES-      115
      1LIM,ISCALE,SCALE,ACCARM,DISKWI,ENGINE,PLOT                               116<-
C               ******************                                              117
C               * DATA STATEMENTS *                                            118
C               ******************                                              119
       DATA IDUC,LSHAF,ENGU,SIU,IVALV/4HDUCT,4HSHAF,4HENGL,4HSIU ,4HVALV/       120
```

```
      DATA ILPC,IHPC,IFAN,IFO,IFI,IHPT,ILPT/3HLPC,3HHPC,3HFAN,2HFO,2HFI,-   121
     13HHPT,3HLPT/                                                          122
      DATA PINP,YES/.TRUE.,4HY   /                                          123
C                                                                           124
C---- TEST WTEST FLAG                                                       125
      IF (IWT.EQ.0) GO TO 540                                               126
      IF (IWAY.GE.0) GO TO 20                                               127
      DO 10 I=1,40                                                          128
      WNTF(I)=WTF(I)                                                        129
      FARN(I)=FAR(I)                                                        130
      TNPRES(I)=TOPRES(I)                                                   131
      TNTEMP(I)=TOTEMP(I)                                                   132
      CNRFLO(I)=CORFLO(I)                                                   133
      DATOUN(1,I)=DATOUT(1,I)                                               134
      DATOUN(2,I)=DATOUT(2,I)                                               135
      DATOUN(8,I)=DATOUT(8,I)                                               136
      DATOUN(9,I)=DATOUT(9,I)                                               137
10    CONTINUE                                                              138
      DO 15 I=1,900                                                         139
15    DANINP(I,1)=DATINP(I,1)                                               140
20    DO 30 I=1,40                                                          141
      CORFLC(I)=CORFLO(I)/1.5497255                                         142
      IF (JTYPE(I).EQ.1) II=I                                               143
      IF (TOPRES(I).GT.TMPRES(I)) DATMAC(1,I)=DATINP(5,II)                  144
      IF (TOPRES(I).GT.TMPRES(I)) DATALT(1,I)=DATINP(9,II)                  145
      IF (TOTEMP(I).GT.TMTEMP(I)) DATMAC(2,I)=DATINP(5,II)                  146
      IF (TOTEMP(I).GT.TMTEMP(I)) DATALT(2,I)=DATINP(9,II)                  147
      IF (DATOUT(2,I).GT.DATOUM(2,I)) DATMAC(3,I)=DATINP(5,II)              148
      IF (DATOUT(2,I).GT.DATOUM(2,I)) DATALT(3,I)=DATINP(9,II)              149
      IF (CORFLC(I).GT.CORFLM(I)) DATMAC(4,I)=DATINP(5,II)                  150
      IF (CORFLC(I).GT.CORFLM(I)) DATALT(4,I)=DATINP(9,II)                  151
      IF (TOPRES(I).GT.TMPRES(I)) TMPRES(I)=TOPRES(I)                       152
      IF (TOTEMP(I).GT.TMTEMP(I)) TMTEMP(I)=TOTEMP(I)                       153
      IF (CORFLC(I).GT.CORFLM(I)) CORFLM(I)=CORFLC(I)                       154
      IF (DATOUT(2,I).GT.DATOUM(2,I)) DATOUM(2,I)=DATOUT(2,I)               155
      IF (JTYPE(I).NE.5) GO TO 30                                           156
      DATTRQ=DATOUT(1,I)/DATOUT(2,I)                                        157
      IF (DATTRQ.GT.DATOUM(1,I)) DATOUM(3,I)=DATOUT(2,I)                    158
      IF (DATTRQ.GT.DATOUM(1,I)) DATOUM(1,I)=DATTRQ                         159
C     THE FOLLOWING MAY BE USEFUL FOR DEBUGGING AND HAS BEEN LEFT IN        160
C     WRITE(10,992)JTYPE(I)                                                 161
C 992 FORMAT(//' COMP TYPE'I3)                                              162
C     WRITE(10,997)                                                         163
C 997 FORMAT('  DES TOTPRES   N  D   TOTEMP')                              164
C     WRITE(10,999)TOPRES(I),TNPRES(I),TOTEMP(I),TNTEMP(I)                  165
C     WRITE(10,996)                                                         166
C 996 FORMAT('  DES CORFLO N  D  M   WTF')                                  167
C     WRITE(10,999)CORFLO(I),CNRFLO(I),CORFLM(I),WNTF(I)                    168
C     WRITE(10,993)                                                         169
C 993 FORMAT('  DES EFF       N  D   PR')                                   170
C     WRITE(10,999)DATOUT(8,I),DATOUN(8,I),DATOUT(9,I),DATOUN(9,I)          171
C     WRITE(10,995)                                                         172
C 995 FORMAT('  MAX TOTPRES   N  M   TOTEMP')                               173
C     WRITE(10,999)TOPRES(I),TMPRES(I),TOTEMP(I),TMTEMP(I)                  174
C     WRITE(10,994)                                                         175
C 994 FORMAT('  MAX RPM       N  M  D')                                     176
C     WRITE(10,999)DATOUT(2,I),DATOUM(2,I),DATOUN(2,I)                      177
C     WRITE(10,991)                                                         178
C 991 FORMAT(' MTRQ  N M  TRPM MTRPM ')                                     179
C     WRITE(10,999)DATTRQ,DATOUM(1,I),DATOUT(2,I),DATOUM(3,I)               180
```

```
C 999 FORMAT(4F12.3)                                                    181
C     WRITE(10,999)DATMAC(1,I),DATMAC(2,I),DATMAC(3,I)                  182
C     WRITE(10,999)DATALT(1,I),DATALT(2,I),DATALT(3,I)                  183
30    CONTINUE                                                          184
      IF (IWT.GE.2) GO TO 40                                            185
C---- ZERO OUT OUTPUT ARRAYS                                            186
      GO TO 540                                                         187
40    JSCALE=0                                                          188
      ISAVE=IWT                                                         189<---
      IF (IWT.NE.4) GO TO 70                                            190
50    JSCALE=JSCALE+1                                                   191
      IF (JSCALE.GT.ISCALE(2)) GO TO 540                                192
      IF (JSCALE.GT.1.AND.ISCALE(1).EQ.2) IOUTCD=1                      193
      SCALEF=SCALE(JSCALE)                                              194
      IF (JSCALE.GT.1) SCALEF=SCALE(JSCALE)/SCALE(JSCALE-1)             195
55    IF(REVISE.EQ.YES) SCALEF=1.0                                      196<---
      DO 60 I=1,40                                                      197
      WNTF(I)=WNTF(I)*SCALEF                                            198
      CNRFLO(I)=CNRFLO(I)*SCALEF                                        199
      CORFLM(I)=CORFLM(I)*SCALEF                                        200
      IDID(I)=0                                                         201
      DATOUN(1,I)=DATOUN(1,I)*SCALEF                                    202
60    CONTINUE                                                          203
      IF (JSCALE.GT.1.AND.IWT.EQ.4) GO TO 120                           204
      IF(REVISE.EQ.YES) GO TO 115                                       205<---
70    DO 80 I=1,5                                                       206
80    DSHAF(I)=0.                                                       207
      DO 90 I=1,60                                                      208
      WATE(I)=0                                                         209
      NSTAG(I)=0                                                        210
      IDID(I)=0                                                         211
      RPMT(I)=0.                                                        212
90    ALENG(I)=0                                                        213
      DO 100 I=1,40                                                     214
      TLENG(I)=0                                                        215
      RI(1,I)=0                                                         216
      ILENG(I)=0                                                        217
      RI(2,I)=0                                                         218
      RO(1,I)=0                                                         219
100   RO(2,I)=0                                                         220
C                                                                       221
C---- NAMELIST READ OF WTEST DATA                                       222
C                                                                       223
      CALL NAMEPR (9,10,8,PINP)                                         224
      READ (8,W)                                                        225
      DO 110 I=1,60                                                     226
      DO 110 K=1,17                                                     227
      IF (IWMEC(1,I).EQ.NUMNUM(K)) IWMEC(1,I)=IRNAME(K)                 228
110   CONTINUE                                                          229
      GO TO 120                                                         230<---
115   WRITE(20,720)                                                     231<---
      CALL NAMEPR(20,10,8,PINP)                                         232<---
      READ(8,W)                                                         233<---
C                                                                       234
C---- PROCESS LENGTH CONTRIBUTING VECTOR EXCEPT DUCTS AND SHAFTS        235
C                                                                       236
120   DO 240 I=1,40                                                     237
      NC=ILENG(I)                                                       238
C     WRITE(10,7777)NC,ILENG(I),JTYPE(NC)                               239
      IF (NC.EQ.0) GO TO 250                                            240
```

```
      JT=JTYPE(NC)                                                          241
      GO TO (240,160,200,130,140,210,190,150,180,240,240,240,240,240),JT    242
C---- COMPRESSOR                                                           243
130   CALL COMP (NC)                                                        244
      GO TO 230                                                             245
C---- TURBINE                                                              246
140   CALL TURB (NC)                                                        247
      GO TO 230                                                             248
C---- MIXER                                                                249
150   CALL WMIXR (NC)                                                       250
      GO TO 230                                                             251
C---- PRIMARY BURNER                                                       252
160   IF (IWMEC(1,NC).EQ.IDUC) GO TO 170                                    253
      IF (IWMEC(1,NC).EQ.IVALV) GO TO 220                                   254
      CALL COMBWT (NC)                                                      255
      GO TO 230                                                             256
C---  DUCTS                                                                257
170   CALL DUCTW (NC)                                                       258
      GO TO 230                                                             259
C----  NOZZLES                                                             260
180   CALL WTNOZ (NC)                                                       261
      GO TO 230                                                             262
C---- SPLITTER                                                             263
190   CALL WSPLT (NC)                                                       264
      GO TO 230                                                             265
C    TRANSFER DIMENSIONS FOR WATER INJECTION                                266
200   CALL DUMMY (NC)                                                       267
      GO TO 230                                                             268
C    HEAT EXCHANGER WEIGHT                                                  269
210   CALL HMEC (NC)                                                        270
      GO TO 230                                                             271
C   VALVES                                                                 272
220   CALL VALVWT (NC)                                                      273
C----  ACCUME LENGTH                                                       274
230   IUP=JCONF(NC,1)                                                       275
      IDN=JCONF(NC,3)                                                       276
      TLENG(IDN)=TLENG(IUP)+ALENG(NC)                                       277
      IF (JT.EQ.6) TLENG(IDN)=TLENG(IUP)                                    278
      ID2=JCONF(NC,4)                                                       279
      IF (ID2.GT.0) TLENG(ID2)=TLENG(IUP)+ALENG(NC)                         280
      IF (JT.EQ.6) TLENG(ID2)=TLENG(IUP)                                    281
      IU2=JCONF(NC,2)                                                       282
      IF (IU2.GT.0) TLENG(IU2)=TLENG(IUP)                                   283
      IDID(NC)=1                                                            284
240   CONTINUE                                                              285
C---- LAST COMPONENT WAS A NOZZLE SET ENGINE MAXIMUM LENGTH                 286
250   ENGLEN=TLENG(IDN)                                                     287
C                                                                           288
C     PROCESS REMAINING COMPONENTS                                          289
C                                                                           290
      DO 340 I=1,60                                                         291
C     WRITE(10,7777)NC,I,IDID(I)                                            292
      IF (IDID(I).EQ.1) GO TO 340                                           293
C                                                                           294
C--   PROCESS COMPRESSORS,TURBINES,MIXERS,BURNERS,SPLITTERS                 295
      NC=JTYPE(I)                                                           296
      IF (NC.LE.0) GO TO 340                                                297
      GO TO (340,290,330,260,270,320,310,280,340,340,340,340,340,340),NC    298
C--   COMPRESSOR                                                            299
260   CALL COMP (I)                                                         300
```

```
      GO TO 340                                                          301
C--   TURBINES                                                           302
270   CALL TURB (I)                                                      303
      GO TO 340                                                          304
C--   MIXER                                                              305
280   CALL WMIXR (I)                                                     306
      GO TO 340                                                          307
C--   BURNERS                                                            308
290   IF (IWMEC(1,I).EQ.IDUC) GO TO 340                                  309
      IF (IWMEC(1,I).EQ.IVALV) GO TO 300                                 310
      CALL COMBWT (I)                                                    311
      GO TO 340                                                          312
C     VALVES                                                             313
300   CALL VALVWT (I)                                                    314
      GO TO 340                                                          315
C---  SPLITTER                                                           316
310   CALL WSPLT (I)                                                     317
      GO TO 340                                                          318
C     HEAT EXCHANGERS                                                    319
320   CALL HMEC (I)                                                      320
      GO TO 340                                                          321
C     TRANSFER DIMENSIONS FOR WATER INJECTION                            322
330   CALL DUMMY (I)                                                     323
340   CONTINUE                                                           324
C     WRITE(10,7771)JTYPE                                                325
C                                                                        326
C---- PROCESS DUCTS                                                      327
      DO 350 I=1,60                                                      328
      IF (IDID(I).EQ.1) GO TO 350                                        329
      NC=JTYPE(I)                                                        330
      IF (NC.NE.2) GO TO 350                                             331
      IF (IWMEC(1,I).NE.IDUC) GO TO 350                                  332
      CALL DUCTW (I)                                                     333
350   CONTINUE                                                           334
C     WRITE(10,7771)JTYPE                                                335
C---- PROCESS NOZZLES                                                    336
      DO 360 I=1,60                                                      337
      IF (IDID(I).EQ.1) GO TO 360                                        338
      NC=JTYPE(I)                                                        339
      IF (NC.NE.9) GO TO 360                                             340
      CALL WTNOZ (I)                                                     341
360   CONTINUE                                                           342
C     WRITE(10,7771)JTYPE                                                343
C---- ACCUME LENGTH                                                      344
      DO 380 I=1,40                                                      345
C     WRITE(10,7777)NC,I,IDID(I)                                         346
      IF (JTYPE(I).LE.0) GO TO 370                                       347
      IF (IDID(I).EQ.1) GO TO 380                                        348
      NC=JTYPE(I)                                                        349
      GO TO (380,370,370,370,370,370,370,370,370,380,380,380,380,380),NC 350
370   IUP=JCONF(I,1)                                                     351
      IU2=JCONF(I,2)                                                     352
      IDN=JCONF(I,3)                                                     353
      ID2=JCONF(I,4)                                                     354
      TLENG(IDN)=TLENG(IUP)+ALENG(I)                                     355
      IF (NC.EQ.6) TLENG(IDN)=TLENG(IUP)                                 356
      IF (IU2.GT.0) TLENG(IU2)=TLENG(IUP)                                357
      IF (ID2.GT.0) TLENG(ID2)=TLENG(IDN)                                358
      IDID(I)=1                                                          359
380   CONTINUE                                                           360
```

```
C---- PROCESS SHAFTS                                                    361
      DO 400 J=1,5                                                      362
      DO 390 I=1,25                                                     363
      NC=KKINDS(11,I)                                                   364
      IF (NC.LE.0) GO TO 400                                            365
      IF (IWMEC(1,NC).NE.LSHAF) GO TO 390                               366
      IF (IWMEC(2,NC).EQ.J) CALL SHAFT (NC)                             367
390   CONTINUE                                                          368
400   CONTINUE                                                          369
C                                                                       370
C---- FIND ENGINE MAXIMUM RADIUS                                        371
      XR=0                                                              372
      DO 420 I=1,NOSTAT                                                 373
      IF (XR.GE.RO(1,I)) GO TO 410                                      374
      XR=RO(1,I)                                                        375
410   IF (XR.GE.RO(2,I)) GO TO 420                                      376
      XR=RO(2,I)                                                        377
420   CONTINUE                                                          378
C                                                                       379
C---- GET ENGINE TOTAL WEIGHT AND  ALENG CONVERSION                     380
      WATENG=0                                                          381
      IF (ACCS.EQ.0) ACCS=.1                                            382
      WAT=0.                                                            383
      DO 430 I=1,60                                                     384
      IF (JTYPE(I).EQ.9) GO TO 430                                      385
      WAT=WATE(I)+WAT                                                   386
430   CONTINUE                                                          387
      WATACC=ACCS*WAT                                                   388
      IF (IOUTCD.GT.1) WRITE (10,580) IOUTCD                            389
      IF (ISIO) WATACC=WATACC*CONVER(3)                                 390
      IF (IOUTCD.GT.1) WRITE (10,610) WATACC                            391
      DO 440 I=1,60                                                     392
      WFACTR=1.                                                         393
      IF (DESVAL(15,I).NE.0.) WFACTR=DESVAL(15,I)                       394
      WATE(I)=WATE(I)*WFACTR                                            395
      IF (.NOT.ISIO) GO TO 440                                          396
      WATE(I)=WATE(I)*CONVER(3)                                         397
      ALENG(I)=ALENG(I)*CONVER(1)                                       398
440   WATENG=WATENG+WATE(I)                                             399
C                                                                       400
C---- CONVERT RADIAL DIMENSIONS AND TLENG                               401
      IF (.NOT.ISIO) GO TO 460                                          402
      DO 450 I=1,NOSTAT                                                 403
      RI(1,I)=RI(1,I)*CONVER(1)                                         404
      RI(2,I)=RI(2,I)*CONVER(1)                                         405
      RO(1,I)=RO(1,I)*CONVER(1)                                         406
      RO(2,I)=RO(2,I)*CONVER(1)                                         407
      TLENG(I)=TLENG(I)*CONVER(1)                                       408
450   CONTINUE                                                          409
C                                                                       410
C---- WRITE COMPONENT WEIGHT INFO                                       411
460   UNITSI=ENGU                                                       412
      IF (ISII) UNITSI=SIU                                              413
      UNITSO=ENGU                                                       414
      IF (ISIO) UNITSO=SIU                                              415
      WRITE (10,590) UNITSI,UNITSO                                      416
      IF (IWT.EQ.4) WRITE (10,550)                                      417
      IF (IWT.EQ.4) WRITE (10,560)                                      418
      IF (IWT.EQ.4) WRITE (10,570) SCALE(JSCALE)                        419
      IF (IWT.EQ.4) WRITE (10,560)                                      420
```

```
      IF (IOUTCD.LT.1) GO TO 510                                                421
      WRITE (10,600)                                                            422
      HLENG=0.                                                                  423
      CGLENG=0.                                                                 424
      CGWATE=0.                                                                 425
      CGTOTM=0.                                                                 426
      CGCOMP=0.                                                                 427
      DO 500 I=1,60                                                             428
      NC=JTYPE(I)                                                               429
      IF (NC.LE.0) GO TO 500                                                    430
C     WRITE(10,7777)JT                                                          431
      GO TO (470,470,470,470,470,470,470,470,470,500,470,500,500,500),NC       432
470   IUP1=JCONF(I,1)                                                           433
      IUP2=JCONF(I,2)                                                           434
      IDN1=JCONF(I,3)                                                           435
      IDN2=JCONF(I,4)                                                           436
      WRITE (10,620) I,WATE(I),ALENG(I),TLENG(IDN1),RI(1,IUP1),RO(1,IUP1-       437
     1),RI(1,IUP2),RO(1,IUP2),RI(2,IDN1),RO(2,IDN1),RI(2,IDN2),RO(2,IDN2-      438
     2),NSTAG(I)                                                                439
      IF (ALENG(I).EQ.0.) GO TO 480                                             440
      HLENG=ALENG(I)/2.                                                         441
      IF (IWMEC(1,I).EQ.ILPC.OR.IWMEC(1,I).EQ.IHPC) CGX=ALENG(I)-CGARM(I-       442
     1)                                                                         443
      IF (IWMEC(1,I).EQ.IFAN.OR.IWMEC(1,I).EQ.IFO) CGX=ALENG(I)-CGARM(I)        444
      IF (IWMEC(1,I).EQ.IFI.OR.IWMEC(1,I).EQ.IHPT) CGX=ALENG(I)-CGARM(I)        445
      IF (IWMEC(1,I).EQ.ILPT) CGX=ALENG(I)-CGARM(I)                             446
      IF (IWMEC(1,I).EQ.ILPC.OR.IWMEC(1,I).EQ.IHPC) HLENG=CGX                   447
      IF (IWMEC(1,I).EQ.IFAN.OR.IWMEC(1,I).EQ.IFO) HLENG=CGX                    448
      IF (IWMEC(1,I).EQ.IFI.OR.IWMEC(1,I).EQ.IHPT) HLENG=CGX                    449
      IF (IWMEC(1,I).EQ.ILPT) HLENG=CGX                                         450
      CGLENG=TLENG(IDN1)-HLENG                                                  451
      GO TO 490                                                                 452
480   IF (IWMEC(1,I).EQ.ISHAF) CGLENG=CGARM(I)                                  453
      IF (IWMEC(1,I).NE.ISHAF) GO TO 500                                        454
490   CGWATE=WATE(I)                                                            455
      CGCOMP=CGWATE*CGLENG                                                      456
      CGTOTM=CGTOTM+CGCOMP                                                      457
500   CONTINUE                                                                  458
      CENGRA=(CGTOTM+(WATACC*ACCARM))/WATENG                                    459
C                                                                               460
C---- MAKE SUMMARY PRINT                                                        461
510   IF (.NOT.ISIO) GO TO 520                                                  462
      XR=XR*CONVER(1)                                                           463
      ENGLEN=ENGLEN*CONVER(1)                                                   464
      CENGRA=CENGRA*CONVER(1)                                                   465
520   WRITE (10,630) WATENG,WATACC,ENGLEN,XR                                    466
      IF (SCALE(JSCALE).EQ.1.) SEXPO1=WATENG                                    467
      SEXPO2=1.                                                                 468
      IF (SCALE(JSCALE).NE.1..AND.IWT.EQ.4) SEXPOE=ALOG(WATENG/SEXPO1)/A-       469
     1LOG(SCALE(JSCALE)/SEXPO2)                                                 470
      IF (SCALE(JSCALE).EQ.1.) SEXPOE=1.                                        471
      WRITE (10,640) CENGRA                                                     472
      IF (IWT.EQ.4) WRITE (10,650) SEXPOE                                       473
      IF (JSCALE.GT.1.AND.ISCALE(1).EQ.2) GO TO 530                             474
      IF (ENGINE.EQ.2.) GO TO 530                                               475
      IF (IPLT) CALL ENGPLT (ENGLEN,XR)                                         476
      IF(PLOT) GO TO 525                                                        477
      GO TO 530                                                                 478
525   IF (SKIPIT) CALL EGPLOT(ENGLEN,WATENG)                                    479
      IF (SKIPIT) GO TO 530                                                     480
```

```
      WRITE(20,700)                                                     481
      READ(20,750) ANSWER                                               482
      IF(ANSWER.NE.YES) GO TO 530                                       483
      CALL EGPLOT(ENGLEN,WATENG)                                        484
      WRITE(20,710)                                                     485
      READ(20,750) REVISE                                               486
      IF(REVISE.EQ.YES) IWT=2                                           487
530   ENGINE=1.                                                         488
      IF(REVISE.EQ.YES) GO TO 55                                        489
      IWT=ISAVE                                                         490
      IF (IWT.EQ.4) GO TO 50                                            491
540   IWT=0                                                             492
      IOUTCD=2                                                          493
      RETURN                                                            494
C                                                                       495
C                                                                       496
550   FORMAT (1H /24H ENGINE SCALING DATA    )                          497
560   FORMAT (20H **********)                                           498
570   FORMAT (14H SCALE FACTOR ,F5.2)                                   499
580   FORMAT (1H /14H **********/14H *          */,14H *  ACCS WT  *-   500
     1/14H *           */13H **********,I1)                             501
590   FORMAT (1H1,26H     WEIGHT INPUT DATA IN ,A4,6H UNITS/27H     WEIG-  502
     1HT OUTPUT DATA IN ,A4,6H UNITS//)                                 503
600   FORMAT (69H    COMP   WT   COMP  ACCU    UPSTREAM RADIUS     DOWNS-  504
     1TREAM RADIUS  /77H     NO   EST   LEN   LEN   RI   RO   RI   RO  -  505
     2  RI   RO   RI   RO    NSTAGE/)                                    506
610   FORMAT (/,11H   ACCS WT=,F8.3)                                    507
620   FORMAT (I7,F6.0,F7.0,F6.0,4F5.0,F6.0,3F5.0,I8)                    508
630   FORMAT (/,27H  TOTAL BARE ENGINE WEIGHT=,F6.0,2X,12HACCESSORIES=,F-  509
     17.2,2X,23HESTIMATED TOTAL LENGTH=,F6.0,2X,25HESTIMATED MAXIMUM RAD-  510
     2IUS=,F5.0)                                                        511
640   FORMAT (30H  ESTIMATED CENTER OF GRAVITY=,F6.0)                   512
650   FORMAT (39H  ESTIMATED AIRFLOW SCALING EXPONENT IS,F6.3)          513
700   FORMAT (46H  DO YOU WISH A GRAPHICS PICTURE? YES=Y ; NO=N)        514
710   FORMAT (54H  DO YOU WISH TO MAKE CHANGES TO THE INPUT? YES=Y;NO=N)  515
720   FORMAT (42H  ENTER DESIRED CHANGES FROM TERMINAL,I.E./39H   &W  DE-  516
     1SVAL(1,6)=****(DATA)****  &END)                                   517
750   FORMAT (A4)                                                       518
      END                                                               519
```

SAMPLE CASE TERMINAL LISTING FOR INTERACTIVE MODE

```
 &W
 IWT=4,IPLT=T,PLOT=T,
 ISII=F,ISIO=F,IOUTCD=2,
 ILENG(1)=2,3,6,7,8,17,9,10,11,12,13,14,
 IWMEC(1,2)='FAN ',0,0,4,3*0,
 IWMEC(1,3)='SPLT',6*0,
 IWMEC(1,4)='DUCT',2,5*0,
 IWMEC(1,5)='NOZ ',1,5*0,
 IWMEC(1,6)='LPC ',1,5*0,
 IWMEC(1,7)='DUCT',2,5*0,
 IWMEC(1,8)='HPC ',1,5*0,
 IWMEC(1,9)='PBUR',1,5*0,
 IWMEC(1,10)='HPT ',0,8,-8,3*0,
 IWMEC(1,11)='DUCT',2,5*0,
 IWMEC(1,12)='LPT ',1,2,6,3*0,
 IWMEC(1,13)='DUCT',1,5*0,
 IWMEC(1,14)='NOZ ',1,5*0,
 IWMEC(1,15)='SHAF',1,12,3*0,6,
 IWMEC(1,16)='SHAF',2,10,3*0,8,
 IWMEC(1,17)='DUCT',2,5*0,
 DESVAL(1,2)=.55,1.8,.37,1.5,4.5,4.5,.5,0.,0.,1.,0.,2.,1.,
 DESVAL(1,3)=15*0.,
 DESVAL(1,4)=.5,1.,0.,4.,11*0.,
 DESVAL(1,5)=1.,14*0.,
 DESVAL(1,6)=.5,1.15,.8,1.5,2.,2.,.45,0,0,1,0,3.,1.,
 DESVAL(1,7)=.45,5.5.0.,8.,11*0.,
 DESVAL(1,8)=.45,1.31,.47,1.5,4.5,1.5,.3,0,0,1.,0.,3.,1.,
 DESVAL(1,9)=100.,.0180,13*0.,
 DESVAL(1,10)=.4,.20,1.5,1.5,1.5,.5,150000.,3.,1.,6*0.,
 DESVAL(1,11)=.5,3.5,0.,-1.,11*0.,
 DESVAL(1,12)=.5,.327,1.5,2.,4.,.6,150000.,1.,1.,6*0.,
 DESVAL(1,13)=.6,.5,0.,14.,11*0.,
 DESVAL(1,14)=1.,14*0.,
 DESVAL(1,15)=50000.,.3,.85,12*0.,
 DESVAL(1,16)=50000.,.3,13*0.,
 DESVAL(1,17)=.3,8.9,0.,-1.,11*0.,
 &END


ASO:WATE2IT TXXXX,RET=T,PRINT=NO,NEWLIB=JOB.BOBLIB
DEFAULTS ARE IN=WHITLOW,OUT=ANSWER.IN,PRINT=YES,MAPS=ADV3DMAP,LIMEN=W,NEWLIB=NONE,RET=P,BACK=NO
THIS RUN,IN=CF6,OUT=ANSWER.CF6,PRINT=NO,MAPS=MAPS.PWSCAR,LIMEN=W,NEWLIB=JOB.BOBLIB,RET=T,BACK=NO
JOB.BOBLIB LOADED
BLOCK DATA BLOCKWT IS LOADED
 ENGWT LOADED
00001
00002
00003
00004
00005
00006
00007
00008
 LAST QUALIFIER IS FROM NEPCAL
00009
00010
00011
00012
00013
00004
 EXECUTION HAS BEGUN
```

```
00003
 > > > > > SUCCESSFUL CONVERGENCE < < < < <
NIT=0
 DO YOU WISH A GRAPHICS PICTURE? YES=Y ; NO=N
Y
 DO YOU WISH MULTIPLE FRAMES? YES=Y ; NO=N
N
 DO YOU WANT COMPONENT OUTPUT? YES=Y ; NO=N
Y
 NOT ENOUGH ROOM BETWEEN LABELS.  NOLAB REDUCED BY 1/2.
 GRAPHICS DEVICE NOT DEFINED BY DDEF.
 ENTER UNIT NAME.  DEFAULT TO CANCEL.
LA001
  MAKE SURE GRAPHICS DEVICE STILL CONNECTED TO TEMPO/3033
 DO YOU WISH TO MAKE CHANGES TO THE INPUT? YES=Y;NO=N
Y
00012
WATENG=+.87486914E+04
WATACC=+.82054126E+03
WATE =
(1)  +.00000000E+00  +.20693882E+04  +.00000000E+00  +.21138214E+02  +.35629565E+03  +.12359275E+04  +.18432083E+0
(8)  +.10429136E+04  +.46836670E+03  +.27845117E+03  +.14609849E+02  +.24869705E+04  +.00000000E+00  +.18697968E+0
(15) +.46437256E+03  +.65491196E+02  +.39371490E+02  +.00000000E+00  +.00000000E+00  +.00000000E+00  +.00000000E+0
(22):(60) CONTAINS +.00000000E+00
 ENTER DESIRED CHANGES FROM TERMINAL,I.E.
  &W  DESVAL(1,6)=****(DATA)****  &END
 &W  DESVAL(3,6)=.84,DESVAL(3,4)=35.,0. &END
 DO YOU WISH A GRAPHICS PICTURE? YES=Y ; NO=N
Y
 DO YOU WISH MULTIPLE FRAMES? YES=Y ; NO=N
N
 DO YOU WANT COMPONENT OUTPUT? YES=Y ; NO=N
Y
 NOT ENOUGH ROOM BETWEEN LABELS.  NOLAB REDUCED BY 1/2.
 DO YOU WISH TO MAKE CHANGES TO THE INPUT? YES=Y;NO=N
N
00012
WATENG=+.85592891E+04
WATACC=+.79504028E+03
WATE =
(1)  +.00000000E+00  +.20693882E+04  +.00000000E+00  +.21235367E+02  +.35957813E+03  +.14904048E+04  +.20406189E+0
(8)  +.10429136E+04  +.46836670E+03  +.27845117E+03  +.14609849E+02  +.19867976E+04  +.00000000E+00  +.24930663E+0
(15) +.45298218E+03  +.65491196E+02  +.39371490E+02  +.00000000E+00  +.00000000E+00  +.00000000E+00  +.00000000E+0
(22):(60) CONTAINS +.00000000E+00
 DO YOU WISH A GRAPHICS PICTURE? YES=Y ; NO=N
N
00012
WATENG=+.65964336E+04
WATACC=+.60576050E+03
WATE =
(1)  +.00000000E+00  +.16723853E+04  +.00000000E+00  +.16504425E+02  +.33938574E+03  +.10575210E+04  +.14601471E+0
(8)  +.74049707E+03  +.38088965E+03  +.20086632E+03  +.10453938E+02  +.14909504E+04  +.00000000E+00  +.19944562E+0
(15) +.38575269E+03  +.59026550E+02  +.28171722E+02  +.00000000E+00  +.00000000E+00  +.00000000E+00  +.00000000E+0
(22):(60) CONTAINS +.00000000E+00
 DO YOU WISH A GRAPHICS PICTURE? YES=Y ; NO=N
N
00012
WATENG=+.10718051E+05
WATACC=+.10039109E+04
WATE =
(1)  +.00000000E+00  +.25008088E+04  +.00000000E+00  +.26188171E+02  +.37977100E+03  +.19757178E+04  +.26824646E+02
(8)  +.13843208E+04  +.55517700E+03  +.36383301E+03  +.19205093E+02  +.25465410E+04  +.00000000E+00  +.29916724E+03
(15) +.51733716E+03  +.71418152E+02  +.51754623E+02  +.00000000E+00  +.00000000E+00  +.00000000E+00  +.00000000E+00
(22):(60) CONTAINS +.00000000E+00
00006
PERPF(4)=+.5190356392337627D+05
PERPF(5)=+.3885521902236921D+00
PERPF(9)=+.5190356392337627D+05
PERPF(10)=+.3885521902236921D+00
 TERMINATED: EXIT IN USER PROGRAM
 CANCELLED: DUM.T2 UNKNOWN.
ASO:
```

IBM-370 GRAPHICS SUBROUTINES (INCLUDES ONLY THOSE USED IN RUNNING THE PROGRAM)

CHARS -- Print Character Data on a Plot

The CHARS subroutine allows the user to print character data anywhere on a plot. The routine can be used to label a point.

General Form
CALL CHARS(nchar,char,orient,x,y,isiz)

nchar
: is an integer constant or variable the absolute value of which specifies the number of characters to be printed. For nchar < 0, the x- and y-coordinates of the starting position for the character data will be in user data units. For nchar > 0, the coordinates of the starting position for the character data will be in relative (0-10), units.

char
: is the array of characters to be printed. The dimension of char must be large enough for the number of characters desired. The characters must be packed four to a word (A4 format). See Appendix A for EBCDIC and integer equivalents for characters available on each device when a stroke table is not in use.

orient
: is the angle of orientation in degrees from the horizontal. (OANGL from GRCOM will be added to this angle). If the angle is 0 degrees, and OANGL is 0 degrees, the characters will be printed right side up and horizontally.

x
: is a floating point constant or variable which represents the x-coordinate of the starting position for the character data. If x is expressed in user units, NCHAR must be less than 0. If x is expressed in relative (0-10), units, NCHAR must be greater than 0.

y
: is a floating point constant or variable which represents the y-coordinate of the starting position for the character data. If y is expressed in user units, NCHAR must be less than 0. If y is expressed in relative units, NCHAR must be greater than 0.

isiz

is an integer constant or variable the value of which specifies the size of the characters to be printed. Its value must be in the range -127 to +127 and is interpreted as follows:

< 0 Display italicized characters.

Each character is drawn in a square whose size in internal units is calculated by the following formula: box size = 128/(143-IABS(ISIZ))* (IABS(ISIZ)+1)*15.0. To find the number of characters that will fit across a frame, divide the box size (found above) into 16384 or see Appendix B.

Note: The standard character size is 15.

Programming Note: If x and y are specified in user's units, the user must have defined XMAX, YMAX, XMIN, and YMIN by previous calls to SCALE, AXIS or NAXIS. If nonstandard axes are to be used and x and y are specified in user's units, the user must have defined the plot boundaries by previous calls to AXIS or NAXIS.

Example: The following is an example of using the CHARS subroutine to label a point.

```
10      DIMENSION CH(3)
20      DATA CH/'(3.0',',5.0',')   '/
30      CALL CHARS(9,CH,0.0,3.0,4.0,15)
```

10 Defines the dimension of the array which contains the characters to be plotted.

20 Initializes the array with the information to be printed.

30 Causes the printing of the label (3.0,5.0) in standard size characters, upright and parallel to the x axis beginning at relative coordinate location (3.0,4.0).

CORNER - Delete / Restore Corner Marks

The CORNER subroutine will remove or restore the corner marks that define a frame.

General Form
CALL CORNER(iopt)

iopt
is an integer indicating whether to remove or restore the corner marks.

Specified as: 1 - remove the corner marks
0 - restore the corner marks

Functional Description: The option will remain set until changed. This call should be made before the frame is displayed (i.e., before the applicable DISPLA call).

DISPLA -- Display

The DISPLA subroutine defines the end of a display or plot and initiates transmission of orders to the device.

General Form
CALL DISPLA(OPT)

opt

is a fullword integer constant or variable the value of which indicates the status of the buffers after the call.

specified as either 0 or 1.

OPT=0 specifies that the general orders buffers are not to be cleared following the display.

OPT=1 specifies that the general orders buffers and GRCOM values will be cleared following the display.

GARC -- Approximate circles and arcs.

The GARC subroutine allows the user to approximate circles and arcs by points or vectors.

General Form
CALL GARC(centx,centy,rad,angle,iopt)

centx
is a real constant or variable the value of which is the x-coordinate of the center of the circle/arc in relative units.

centy
is a real constant or variable the value of which is the y-coordinate of the center of the circle/arc in relative units.

rad
is a real constant or variable the value of which is the radius of the circle/arc in relative units.

angle
is a real array for specifying variable data to the ARC subroutine. Its entries should contain the following information:

angle(1)	The incremental angle for points or vectors. If angle(1) is 0, the default for the device is used.
angle(2)	The beginning angle for the arc in degrees.
angle(3)	The counterclockwise sweep angle for the arc.

iopt
is an integer constant or variable which indicates whether points or vectors are to be used to approximate the arc. Its value must be one of the following:
0 - use default for device.
1 - approximate with points.
2 - approximate with vectors.

GPLOT -- Generate a Plot

The GPLOT subroutine enables the user to plot single or multiple curves or lines as vector, point, symbol, or vector-symbol plots.

General Form
CALL GPLOT(x,y,ivars)

x
is a floating point array of x-coordinates of the points to be plotted or the array of radius-values for a polar plot.

y
is a floating point array of y-coordinates of the points to be plotted or the array of theta-values (in degrees) for a polar plot.

ivars
is an integer array for communicating other options to the subroutine.

When plotting a single curve, the information supplied in ivars should be specified as follows:

ivars(1)	The number of elements in ivars.
ivars(2)	The number of points in the curve.
ivars(3)	The type of plot, same as in IVARS(2) for multiple curve plot. Default: 0 is assumed.
ivars(4)	The symbol code for a symbol plot.
ivars(5)	The symbol frequency for a symbol plot. Default: 1 is assumed.
ivars(6)	The size of the symbol to be plotted. Default: 15 is assumed.
ivars(7)	The interval option. 0 - scaled (adjusted) to include MIN and MAX. 1 - exact MIN and MAX interval. Default is 0 (adjusted).

Notes: If a vector plot is desired, ivars(3) through ivars(7) need not be specified. If a point plot is desired, ivars(4) through ivars(7) need not be

specified.

For plotting multiple curves, the entries in ivars should contain the following information:

ivars(1) The number of elements in ivars.

ivars(2) The type of plot. Its value must be one of the following:

0	vector plot
1	point plot
2	vector-symbol plot
3	symbol plot
4	polar vector plot with no call to PAXIS
5	polar point plot with no call to PAXIS
6	polar vector-symbol plot with no call to PAXIS
7	polar symbol plot with no call to PAXIS
12	polar vector plot with PAXIS call
13	polar point plot with PAXIS call
14	polar vector-symbol plot with PAXIS call
15	polar symbol plot with PAXIS call
16	log vector plot
17	log point plot
18	log vector-symbol plot
19	log symbol plot
20	log-polar vector plot with NAXIS call
21	log-polar point plot with NAXIS call
22	log-polar vector-symbol plot with NAXIS call
23	log-polar symbol plot with NAXIS call
28	log-polar vector plot with axes
29	log-polar point plot with axes
30	log-polar vector-symbol plot with axes
31	log-polar symbol plot with axes

ivars(3) The symbol code for a symbol plot (See Appendix A).

ivars(4) The frequency of printing the symbol for a symbol plot.
Example: ivars(4) = 3 would cause the printing of the special symbol at every third point.

ivars(5) Specifies whether or not duplication of coordinates is to take place when plotting multiple curves. Its value must be one of the following:

+1	duplicate x-coordinates
0	no duplication of coordinates
-1	duplicate y-coordinates

ivars(6) The size of the symbol to be plotted. Default: 15 is assumed.

ivars(7) The interval option.
- 0 - scaled (adjusted) to include MIN and MAX.
- 1 - exact MIN and MAX interval.

If not specified, an adjusted interval will be created.

ivars(8) - ivars(ivars(1))
One entry for each curve specifying the number of points in that curve to be plotted.

If there have been no calls to AXIS before PLOT is called, the standard size axes with 90 degree orientation will be drawn with no titles, but with 10 intervals and 11 grid lines or tick marks and 11 labels, (tick marks will be the default).

TITLE can be called to print titles for standard x or y axes or to print a plot title. TITLE should be called following GPLOT.

If XMAX and XMIN or YMAX and YMIN are equal, GPLOT assumes maximum and minimum have not been determined and calls SCALE to scan the array and determine the maximum and minimum coordinates. Depending on the interval option, if scaled or adjusted was requested, GINTVL will be called to create an interval containing the data MIN and MAX.

To preset XMIN, XMAX, YMIN, YMAX, use the AXIS routines, SCALE, SCLBAK, or GINTVL.

Specifying a symbol type plot allows the user to plot symbols or characters at random locations. The symbol or character will be centered at the X,Y location.

Note: In FORTRAN programming, selective array locations cannot be defaulted. For example, if only array location 1, 2, and 3 in IVALS are specified, the remaining locations will be defaulted. However, if location 7 is specified, locations 1-6 must also be specified. Zero is not recognized as a default value.

Example: The following example shows the use of the GPLOT subroutine to plot multiple curves as point plots.

```
5       DIMENSION IVARS(9)
10      DIMENSION X(30),Y(60)
20      XSTART = 0.0
30      DO 40 I=1,30
        X(I) = XSTART
        Y(I) = XSTART*XSTART + XSTART + 1
        Y(I+30) = XSTART*XSTART - XSTART + 2
        XSTART = XSTART + 0.5
```

```
40      CONTINUE
45      IVARS(1) = 9
50      IVARS(2) = 1
55      IVARS(3) = 0
60      IVARS(4) = 0
65      IVARS(5) = +1
70      IVARS(6) = 0
80      IVARS(7) = 0
90      IVARS(8) = 30
95      IVARS(9) = 30
100     CALL GPLOT(X,Y,IVARS)
```

5 Dimensions the IVARS array.
10 Defines the x and y arrays. There will be 2 y-coordinates for each x-coordinate.
20 Initializes XSTART at 0.0.
30-40 This loop places values in the X and Y arrays.
45 Defines the number of elements in IVARS to be seven.
50 Specifies a point plot.
55-60 Specifies no symbol code or index.
65 Specifies that there will be duplicate x-coordinates.
70 Specifies no symbol size (no symbol).
80 Specifies interval option adjusted.
90-95 Specify that there will be two curves with 30 points in each.

GRINIT -- Graphics package initialization.

The GRINIT subroutine initializes the graphics package.

General Form
CALL GRINIT

There are no parameters for this call.

Functional description: GRINIT causes initialization of the graphics package and prompts the user for a device (if one has not been previously defined). This call is optional in that a call to any graphics subroutine will cause initialization if not previously accomplished.

NUMBER -- Convert an Integer or Real Number to a Character Array

The NUMBER subroutine allows the user to convert an integer or real number to an array of printable characters.

General Form
CALL NUMBER(ityp,rnum,nochar,nodec,char)

ityp
is an integer constant or variable the value of which specifies the type of number to be converted. Its value must be one of the following:

- 0 Specifies an integer halfword is to be converted.
- 1 Specifies an integer fullword is to be converted.
- 2 Specifies a single precision real number is to be converted to 'E' format.
- 3 Specifies a double precision real number is to be converted to 'D' format.
- 4 specifies a single precision real number is to be converted to 'F' format.

rnum
is a constant or variable (either integer or real as specified in ityp) the value of which is the number to be converted.

nochar
is an integer constant or variable the value of which specifies the number of characters desired in the output array.

nodec
is an integer constant or variable the value of which specifies the number of positions right of the decimal point.

char
is the output array of printable characters. Char must be specified large enough to hold the number of characters requested (nochar). The characters are packed four to a word (A4 format).

Note: If ITYP = 2 or 3, and NOCHAR is less than 8 or NODEC is greater than (NOCHAR-6), or if ITYP = 0, 1 or 4, and NOCHAR is not large enough to hold the number of significant digits in RNUM, or if ITYP is invalid, asterisks are placed into the output array.

Example: The user wants to convert the real number XMAX to an array of printable characters in an array named CHARS. The printed number will be 8 digits long with 2 decimal places and will be in E format.

```
10      DIMENSION CHARS(2)
20      CALL NUMBER(2,XMAX,8,2,CHARS)
```

10 CHARS contains two elements (8 characters in A4 format).

20 Calls the NUMBER subroutine:

- 2 - convert a real number
- XMAX - contains the number to be converted
- 8 - characters in output array
- 2 - positions right of the decimal point
- CHARS - stores the printable characters

SETFRM -- Set number of frames for plot

The SETFRM subroutine allows the user to specify the number of frames to be plotted.

General Form
CALL SETFRM(frames)

frames
is an integer constant or variable the value of which specifies the number of frames in the user's plot, creating a larger physical plotting area in the x direction. Its value may range from 1 to 21 inclusive.

Example: The following is an example of using the SETFRM subroutine to set the number of frames to 2.

```
30      CALL SETFRM(2)

30      sets plot boundaries for a plot of 2 frames (up
        to 20.0 relative units in x direction).
```

TITLE -- Print Plot or Axis Title

The TITLE subroutine allows the user to print a plot title for a plot or to print titles for the axes used in the plot.

General Form
CALL TITLE(ityp,nchar,isiz,char)

ityp
is an integer constant or variable the value of which specifies what type of title is to be generated. Its value must be one of the following:

0 A plot title is to be printed, centered on the plot.
1 A plot title is to be printed, left justified on the plot.
2 A plot title is to be printed, right justified on the plot.
3 A y-axis title is to be printed. Not valid for polar plots.
4 An x-axis title is to be printed. Not valid for polar plots.

nchar
is an integer constant or variable the value of which specifies the number of characters in the title.

isiz
is an integer constant or variable the value of which specifies the size of the characters to be printed. Its value must be in the range -127 to +127 and is interpreted as follows:

< 0 Display italicized characters.

Each character is drawn in a square whose size in internal units is calculated by the following formula: box size = (128/(143-IABS(ISIZ))* (IABS(ISIZ)+1)*15.0. To find the number of characters that will fit across a frame, divide the box size (found above) into 16384 or see Appendix B.

Note: The standard character size is 15.

char

is an array of characters for the title. The dimension of CHAR must be large enough for the number of characters desired. The characters must be packed four to a word (A4 format).

Programming Note: If TITLE is called to print plot or axis titles for a plot with nonstandard placement or orientation of axes, the plot boundaries must have been defined by a previous call to AXIS or NAXIS.

Example: The following example shows how the TITLE subroutine can be used to print axes titles and a plot title.

```
10   DIMENSION XTITLE(6), YTITLE(6), PTITLE(5)
20   DATA XTITLE/'SAMP','LE X','-AXI','S TI','TLE '/
21   DATA YTITLE/'SAMP','LE Y','-AXI','S TI','TLE '/
22   DATA PTITLE/'SAMP','LE P','LOT ','TITL','E   '/
30   CALL TITLE (4,19,15,XTITLE)
40   CALL TITLE (3,19,15,YTITLE)
50   CALL TITLE (0,17,35,PTITLE)
```

10 Defines the arrays from which the titles will be printed.

20 Initializes the arrays defined in 10 with the actual titles to be printed.

30 Causes the x-axis title to be printed in standard size characters.

40 Causes the y-axis title to be printed in standard size characters.

50 Causes the plot title to be printed in large, (3/8"), characters and centered.

XAXIS, YAXIS, GAXIS -- Draw Nonstandard Axes

The axis subroutines can be used to draw axes of nonstandard lengths, in nonstandard positions, with nonstandard grid/tick mark frequency, and/or at angles other than 0 degrees and 90 degrees. It also permits the user to specify a range of plotted coordinate values. If no axes are to be drawn the user must call NAXIS before PLOT is called.

XAXIS is used to draw x axes, YAXIS is used to draw y axes, and GAXIS is used to draw additional axes. Both XAXIS and YAXIS should be called in succession to avoid defaulting the other's values. GAXIS will affect the scaling and placement of the user's data.

General Form
CALL XAXIS(x,y,vars)
CALL YAXIS(x,y,vars)
CALL GAXIS(x,y,vars)

x

is a real constant or variable which represents, for GAXIS calls, the x-coordinate of the starting position of the axis in relative units, and for XAXIS or YAXIS calls, the x-coordinate of the lower left plot boundary. A value of -1.0 represents the standard starting x-coordinate position.
NOTE 1: The values entered on the first call, (to either X or YAXIS), should be the same as the values used for the second call. (to either X or YAXIS). If they are not, the results will be unpredictable.
NOTE 2 : If the standard X and Y location is specified (-1.0), the package will calculate the X and Y location, allowing room for horizontal y-axis labels. If the X and Y location is specified, it will be used even if there is insufficient room for the labels.

y

Same definition as for x above, except that this value represents a y-coordinate.

vars

is a real array for communicating other variables and options to the subroutine. The entries in vars contain the following information:

- vars(1) The number of elements in vars. (0-9)
- vars(2) The axis length in relative units, (0-10 for Y; 0-210 for X). Default or -1. gives standard length (XUR-XLL) if XAXIS is called; (YUR-YLL) if YAXIS is called.

VARS(3) The angle of orientation in degrees. Specify as 0 degrees for x-axis and 90 degrees for y-axis. The default for vars(3) is 0 degrees + OANGL (from GRCOM) if xaxis is called; 90 degrees + OANGL if YAXIS is called.

vars(4) The value represented by starting point of axis (maximum or minimum). The default for vars(4) is (XMIN/YMIN). Note: This value must be entered unless the user has already specified XMIN/YMIN by using SCALE, SCLBAK, or GINTVL. The value entered for vars(4) replaces the current XMIN/YMIN value in GRCOM.

vars(5) The value represented by end point of axis (maximum or minimum). The default for vars(5) is (XMAX,YMAX). Note: The same rule applies as in note above for vars(4), except this value replaces XMAX|YMAX.

vars(6) The number of grid line/tick mark intervals. Its value can be one of the following:

- n the number of grid line/tick mark intervals desired.
- .5 standard number of grid line/tick mark intervals. (10 intervals--11 tick marks). This is the default value if not entered.
- -n the number of cycles for a logarithmic scale.

If 0 is input, 1 interval with 2 grid lines/tick marks will be created and the default no of labels will be set to 0. If 1 is input, 1 interval with 2 grid lines/tick marks will be created and the default no of labels will be set to 2.

vars(7) Grid line/tick mark option (for XAXIS or YAXIS calls only). Its value should be one of the following:

- 0 default for device. (this is the default if not input or if incorrect).
- 1 grid lines.
- 2 tick marks.

vars(7) The associated YAXIS angle (for GAXIS calls only). Default is vars(3) + 90 degrees + OANGL.

vars(8) The number of labels per axis. If this element is not specified, or is negative, the number of labels will correspond to the number of grid lines/tick marks, (intervals + 1), except when vars(6) was input as 0., in which case, the number of labels will be 0. The y-axis labels will be horizontal and on the counterclockwise

side of the axis.

vars(9) The user data value (DX|DY) for displacement of this axis (on the other axis) in user units. Default if not entered or invalid is zero. The DX|DY values will be calculated automatically for the user, (for XAXIS & YAXIS calls only), to cross the axes at 0,0 if -255. is entered. If this value is entered as -255 for a call to either X or YAXIS, it must also be entered as -255 for the other call.

vars(10) The size of the labels generated by this routine. Value from 0 to 127. Default if not specified will be 10.

Note: In FORTRAN programming, selective array locations cannot be defaulted. For example, if only array locations 1,2 and 3 are specified, 4 - 10 will be defaulted. However, if 1 and 2 are specified and 6 and 7 are specified, 3, 4, and 5 must also be specified (although 8, 9, and 10 can be defaulted). Zero is not recognized as a default value.

Example: The following is an example of using the AXIS subroutines to draw axes with nonstandard specifications. The user wants the x axis to be drawn at a 0 degree angle with the horizontal, the y axis to be drawn at a 90 degree angle with the horizontal, and the origin to be at user data point 0.,0..

```
      DIMENSION VARS(9)
10    VARS(1) = 9.
20    VARS(2) = 7.
30    VARS(3) = 0.
40    VARS(4) = -20.
50    VARS(5) = 20.
60    VARS(6) = 4.
70    VARS(7) = 2.
80    vars(8) = -1.
90    vars(9) = -255.
100   CALL XAXIS (1., 1., VARS)
110   VARS(3) = 90.
120   VARS(4) = -100.
130   VARS(5) = 200.
140   vars(6) = 6.
150   CALL YAXIS (1., 1., VARS)
```

10 Defines the number of elements in VARS to be 9.

20 Sets the length of the x axis equal to 7 relative units, and x upper right plot boundary to x + length or 8 relative units.

30 Sets the angle of orientation for the x axis to 0 degrees.

40 Sets the value represented by the starting point of the x axis at -20.

50 Sets the value represented by the ending point

GTERM Command

This command is required to close the graphics data set and perform any necessary cleanup functions.

Operation	Operand
GTERM	

Note: there are no operands.

Functional Description: GTERM initiates the closing of the graphics data set, and any necessary cleanup functions. For offline devices, GTERM establishes the BWQ task to transfer the dataset to tape for plotting. GTERM also frees virtual memory in which graphics entities have been saved.

PROCEDURE DEFINITION: WATE2IT

ORIGINAL PAGE IS OF POOR QUALITY

```
 PROCLIST WATE2IT
WATE2IT 0000000 PROCDEF WATE2IT
WATE2IT 0000100 PARAM IN=$IN,OUT=$OUT,PRINT=$PRINT,MAPS=$MAPS,LIMEN=$LIMEN,NEWLIB=$NEWLIB,RET=$RET,BACK=$BACK
WATE2IT 0000200 D 'DEFAULTS ARE IN=WHITLOW,OUT=ANSWER.IN,PRINT=YES,MAPS=ADV3DMAP,LIMEN=W,NEWLIB=NONE,RET=P,BACK=NO'
WATE2IT 0000300 IF '$OUT'¬='NONE'; -
WATE2IT 0000400 D 'THIS RUN,IN=$IN,OUT=ANSWER.$OUT,PRINT=$PRINT,MAPS=$MAPS,LIMEN=$LIMEN,NEWLIB=$NEWLIB,RET=$RET,BACK=$BA
WATE2IT 0000500 IF '$OUT'='NONE'; -
WATE2IT 0000600 D 'THIS RUN,IN=$IN,OUT=ANSWER.$IN,PRINT=$PRINT,MAPS=$MAPS,LIMEN=$LIMEN,NEWLIB=$NEWLIB,RET=$RET,BACK=$BAC
WATE2IT 0000700 DEFAULT LIMEN=$LIMEN
WATE2IT 0000800 IF '$OUT'¬='NONE';ERASE ANSWER.$OUT
WATE2IT 0000900 IF '$OUT'='NONE';ERASE ANSWER.$IN
WATE2IT 0001000 RELEASE FT01F001;DDEF FT01F001,VS,DUM.T1,RET=T
WATE2IT 0001100 RELEASE FT02F001;DDEF FT02F001,VS,DUM.T2,RET=T
WATE2IT 0001200 RELEASE FT03F001;DDEF FT03F001,VS,DUM.T3,RET=T
WATE2IT 0001300 RELEASE FT04F001;DDEF FT04F001,VS,DUM.T4,RET=T
WATE2IT 0001400 RELEASE FT05F001;DDEF FT05F001,VS,DUM.T5,RET=T
WATE2IT 0001500 RELEASE FT06F001;DDEF FT06F001,VS,DUM.T6,RET=T
WATE2IT 0001600 RELEASE FT08F001;DDEF FT08F001,VS,DUM.T8,RET=T
WATE2IT 0001700 RELEASE FT09F001;DDEF FT09F001,,$IN
WATE2IT 0001800 RELEASE FT13F001;DDEF FT13F001,VS,SAVEFILE
WATE2IT 0001900 RELEASE FT14F001;DDEF FT14F001,VS,AMACPTS
WATE2IT 0002000 IF '$OUT'¬='NONE'; -
WATE2IT 0002100 RELEASE FT10F001;DDEF FT10F001,VS,ANSWER.$OUT,RET=$RET
WATE2IT 0002200 IF '$OUT'='NONE'; RELEASE FT10F001;DDEF FT10F001,VS,ANSWER.$IN,RET=$RET
WATE2IT 0002300 RELEASE FT15F001;DDEF FT15F001,VS,LASTOPT
WATE2IT 0002400 RELEASE FT11F001;DDEF FT11F001.VS,DUM.T11,RET=T
WATE2IT 0002500 RELEASE FT12F001;DDEF FT12F001,VS,$MAPS
WATE2IT 0002600 IF '$OUT'='CONSOLE';RELEASE FT10F001
WATE2IT 0002700 IF '$IN'='KEYBOARD';RELEASE FT09F001
WATE2IT 0002800 DDEF XXX1,VP,GRAPHICS,OPTION=JOBLIB
WATE2IT 0002850 IF '$BACK'='YES';RELEASE GRAPHICS;DDEF GRAPHICS,VS,DSNAME=ZETA12(+1),DCB=(RECFM=V,LRECL=1028,MACRF=P),RE
WATE2IT 0003000 IF LOADED='TRUE';GOTO ENGWT
WATE2IT 0003100 RELEASE XXYYZZ
WATE2IT 0003200 RELEASE DDNNEP;DDEF DDNNEP,VP,VCELIB2,OPTION=JOBLIB
WATE2IT 0003300 RELEASE WTNNEP;DDEF WTNNEP.VP,LIB.WATE2,OPTION=JOBLIB
WATE2IT 0003400 IF '$NEWLIB'¬='NONE';RELEASE AANNEP;DDEF AANNEP,VP,$NEWLIB,OPTION=JOBLIB;D '$NEWLIB LOADED'
WATE2IT 0003500 LOAD BLOCKWT;D 'BLOCK DATA BLOCKWT IS LOADED'
WATE2IT 0003600 QUALIFY ENGWT;D ' ENGWT LOADED'
WATE2IT 0003700 SET LOADED='TRUE'
WATE2IT 0003800 REMOVE ALL
WATE2IT 0003900 QUALIFY NEPCAL
WATE2IT 0004000 AT 140;IF NVOPT=0 & NIT>0;IF  NIT=(NIT/5)*5;D ' ITERATING', NIT
WATE2IT 0004100 AT 370(2);IF NVOPT=0 & NIT=51;D ' $ $ $ $ $ FAILED TO CONVERGE $ $ $ $ $',NIT
WATE2IT 0004200 AT 370(2);IF NVOPT=0 & NIT<51;D ' > > > > > SUCCESSFUL CONVERGENCE < < < < < ',NIT
WATE2IT 0004300 AT 0(2);D ' EXECUTION HAS BEGUN'
WATE2IT 0004400 DEBUG
WATE2IT 0004500 QUALIFY WTEST
WATE2IT 0004600 AT 530;IF NVOPT=0;D WATENG,WATACC,WATE
WATE2IT 0004625 QUALIFY BWTEST
WATE2IT 0004650 AT 30(2);IF '$BACK'='YES';SET SKIPIT='.TRUE.'
WATE2IT 0004700 QUALIFY NEPCAL
WATE2IT 0004800 ENGWT
WATE2IT 0004900 CLOSE ANSWER.$OUT
WATE2IT 0005000 CLOSE ANSWER.$IN
WATE2IT 0005100 CLOSE LASTOPT
WATE2IT 0005200 CLOSE $IN
WATE2IT 0005300 CLOSE $MAPS
WATE2IT 0005400 RELEASE FT;ERASE DUM.T1;ERASE DUM.T2;ERASE DUM.T8;ERASE DUM.T11
WATE2IT 0005500 IF '$PRINT'='YES' & '$OUT'¬='NONE'; PRINT ANSWER.$OUT,,,EDIT
WATE2IT 0005600 IF '$PRINT'='YES' & '$OUT'='NONE'; PRINT ANSWER.$IN,,,EDIT
WATE2IT 0005700 IF '$BACK'='YES';GTERM
```

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-82756		

4. Title and Subtitle	5. Report Date
INTERACTIVE-GRAPHIC FLOWPATH PLOTTING FOR TURBINE ENGINES	November 1981
	6. Performing Organization Code 505-32-72
7. Author(s)	8. Performing Organization Report No.
Robert R. Corban	E-1074
	10. Work Unit No.
9. Performing Organization Name and Address	
National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Technical Memorandum
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

NASA-Lewis and the Navy jointly developed, based on a previous Navy code, an engine cycle program capable of simulating the design and off-design performance of arbitrary turbine engines. Boeing, under a NASA-Lewis contract, developed a computer code which, when used in conjunction with the cycle code, could predict the weight of the engines. Thus, the Navy/NASA Engine Program (NNEP) along with WATE-2 (Weight Analysis of Turbine Engines) determines the dimensions and weight of each major component in the engine. The output from these codes originally included a very crude and indecipherable computer drawing of the flowpath representation of the designed turbine engine. This proved to be of little use. It was desired to add a graphics subroutine to the code to enable the engineer to visualize the designed engine with more clarity by producing an overall view of the designed engine for output on a graphics device using IBM-370 graphics subroutines. In addition, with the engine drawn on a graphics screen, the program would allow for the interactive user to make changes to the inputs to WATE-2 for the engine to be redrawn and reweighed. These improvements would allow better use of the WATE-2 code in conjunction with NNEP.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Propulsion; Turbine engines; Graphics; Flowpath; Analysis	Unclassified - unlimited STAR Category 07

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified		

* For sale by the National Technical Information Service, Springfield, Virginia 22161