REFERENCE

REPORT NO. DOT-TSC-OST-71-16

APPENDIX C
THE NOISE EXPOSURE MODEL MOD 4

R. H. HINCKLEY AND J. E. WESLER
TRANSPORTATION SYSTEMS CENTER
55 BROADWAY
CAMBRIDGE, MA. 02142

AUGUST 1971
TECHNICAL REPORT

Prepared for

OFFICE OF THE SECRETARY
DEPARTMENT OF TRANSPORTATION
WASHINGTON, D. C. 20590

The contents of this report reflect the views of the Office of the Secretary which is responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policy of the Department of Transportation. This report does not constitute a standard, specification or regulation.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT-TSC-OST-71-16	PB 211978	

4. Title and Subtitle	5. Report Date
Appendix C: The Noise Exposure Model MOD4	August 1971
	6. Performing Organization Code TME
7. Author(s)	8. Performing Organization Report No. DOT-TSC-OST-71-16
9. Performing Organization Name and Address Transportation Systems Center 55 Broadway Cambridge, MA 02142	10. Work Unit No. OS-207
	11. Contract or Grant No.
	13. Type of Report and Period Covered Technical Report
12. Sponsoring Agency Name and Address Office of Noise Abatement Department of Transportation 400 Seventh Street, S.W. Washington, D.C. 20590	14. Sponsoring Agency Code R-2519

15. Supplementary Notes

16. Abstract

17. Key Words	18. Distribution Statement
	Unclassified-Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	57	

INTRODUCTION

Appendix C to the basic report entitled "The Noise Exposure Model MOD 4" contains the computer program listing for the Noise Exposure Model MOD 4. This listing is bound separately from the basic report for convenience.

```
$JOB  D1133E 2 STC   FOREMAN MOD 4                    010 010
$EXECUTE       IBJOB
$IBJOB         FIOCS
$IBFTC MAIN    DECK
C MAIN PROGRAM-AIRPORT NOISE ABATEMENT PROGRAM-VERSION MOD 4
      COMMON/RYNME/RYNAM(10)
      COMMON/FACTR/AXMAX,AXMIN,AYMAX,AYMIN,DELX,DELY,AINCHX,AINCHY,
     2  SETPLT
      COMMON/ARP/ARNV(20,25)
      COMMON/JOE3/JUMP
      COMMON/JOE2/NEFFLG,AREAFG
      COMMON/ARPLOT/PRHARR(30,2,10),IMXARR(10),XNEV(10)
      COMMON/JOE1/WECPFG,CANCEL
      COMMON/ERRIN/JEICS
      COMMON/JOE/SYMFLG,REFLIN,XCOORD(2),YCOORD(2),TITLE(12)
      COMMON/GDPAR/CX,CY,X1,Y1,NX,NY
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),XLM1(3,10),
     1  XLM2(3,10),STO(10),SL(10)
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/TRNS/PI
      COMMON/NNI/INN
      COMMON/NERAP/NOITT,EPS1,EPS2
      COMMON/QR/TOL,VAL,MX,APPROX,BASEX,XDEL
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/LOGFG3/RWAYFG,GRIDFG,CONTFG
      LOGICAL NEFFLG,AREAFG
      REAL N
      LOGICAL WECPFG,CANCEL
      LOGICAL SYMFLG
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      LOGICAL RWAYFG,GRIDFG,CONTFG
      LOGICAL SETPLT
C INITIALIZATION SECTION
      ITOM=0
    1 WRITE(6,193)
  193 FORMAT(1H1)
      DO 40 I=1,10
   40 IMXARR(I)=0
      INN=0
   50 CALL READIN(INDEX)
      JEICS=0
C     INDEX        SIGNIFICANCE
C       1          PROCES ENCOUNTERED IN READING INSTRUCTIONS/DATA
C       2          NEWSET READ NOT INCLUDING FIRST CASE IF INCLUDED
C       3          ENDRUN READ
      GO TO(100,200,300),INDEX
  100 IF(PTINFG) CALL READOU
      IF(.NOT. CALFLG) GO TO 120
      IF(ITOM .EQ. 1) GO TO 120
      CALL INIPLT
      ITOM=1
  120 CONTINUE
      CALL EXTARR
```

```
      CALL DSTORE
      CALL SUMRFX
      IF(GRIDFG) CALL POINTS
      IF(.NOT. CONTFG) GO TO 50
      INN=INN+1
      CALL CONTR
      GO TO 50
C NEWSET READ BUT NOT INCLUDING FIRST CASE IF IT SHOULD APPEAR THERE
  200 IF(CALFLG) CALL CALPLT
      GO TO 1
C ENDRUN READ--THIS SIGNALS END OF INPUT
  300 IF(CALFLG) CALL CALPLT
      IF(ITOM .EQ. 1) CALL FINPLT
      STOP
      END
$IBFTC BLDA    DECK
      BLOCK DATA
      COMMON/JOE3/JUMP
      COMMON/NERAP/NOITT,EPS1,EPS2
      COMMON/DENOP/NOPARR(3),CONARR(3),S
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
      COMMON/BK/DIAG,YES,MAYBE                                          10450
      COMMON/XXX/IPRFL,IPR(100)                                         10500
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),              00010550
     1XLM1(3,10),XLM2(3,10),STO(10),SL(10)                             010600
      COMMON/BB/R0(3),FLIGHT,PFN,EPNDB
      COMMON/CC/LAMBD1(3),LAMBD2(3),TAU1,TAU2,WEIGHT,ACTYPE,         00010700
     1DIST,SLOPE,UG(3),SINB,XDIST,GLUMP                                010750
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/WTPFV/NACT,WFCTR,PFGWT(2,2,11)                          00010850
      COMMON/TOPRF/NTOP,DFCTR,TOPRF(2,8)                                010900
      COMMON/TRNS/PI                                                    10950
      COMMON/LN/VLN                                                     11000
      COMMON/NCOORD/NC                                                  11050
      COMMON/QR/TOL,VAL,MX,APPROX,BASEX,XDEL                         0 011100
      DIMENSION VT(3),AT(3),RC(3),RHO(3),B(3),C(3),S(3)              00011150
      DIMENSION UX(3),VX(3),GMX(3),XNORM(3)                          00011200
      REAL MAYBE
      REAL LAMBD1,LAMBD2,NE,NEY1,NEY2                                   011250
      REAL N,NT(3)                                                      11300
      INTEGER PREVFL,PREVSG                                             11350
      DATA JUMP/0/
      DATA NOITT,EPS1,EPS2/5,.01,.01/
      DATA NACT/13/,WFCTR/1000./,PFGWT/1.075,400.,4.825,800.,        00011400
     10.,200.,4.,400.,0.6,200.,5.,350.,1.3,200.,5.,307.5,.35,        00011450
     2100.,4.,170.,0.,60.,2.75,120.,-1.,40.25,.7,57.,-1.,24.,        00011500
     31.5,42.5,-1.,16.,1.5,27.,-1.,8.75,.9,15.,-.375,400.,           00011550
     41.75,670./,NTOP/8/,DFCTR/1000./,TOPRF/.333,-832.5,             00011600
     5.209,-752.4,.169,-929.5,.133,-904.4,                          0 011650
     6.105,-966.,.0875,-927.5,.0895,-223.75,.0812,-324.8/,          00011700
     7PI/3.1415927/,VLN/1.0E30/                                        11750
      DATA NC/3/,YES/1HY/,MAYBE/1HM/                                    11800
      DATA CONARR/36.35,30.33,35.11/
      DATA S/0./
      END
$IBFTC KGLB    DECK
      SUBROUTINE KGTOLB(A,IDIM)
```

```
      DIMENSION A(1)
      DO 1 I=1,IDIM
    1 A(I)=A(I)*2.204618750
      RETURN
      END
$IBFTC LBKG    DECK
      SUBROUTINE LBTOKG(A,IDIM)
      DIMENSION A(1)
      DO 1 I=1,IDIM
    1 A(I)=A(I)*.4535924277
      RETURN
      END
$IBFTC MFT     DECK
      SUBROUTINE MTOFT(A,IDIM)
      DIMENSION A(1)
      DO 1 I=1,IDIM
    1 A(I)=A(I)*3.280833333
      RETURN
      END
$IBFTC FTM     DECK
      SUBROUTINE FTTOM(A,IDIM)
      DIMENSION A(1)
      DO 1 I=1,IDIM
    1 A(I)=A(I)*.3048006096
      RETURN
      END
$IBFTC SUMREX  DECK
      SUBROUTINE SUMRE
C THIS SUBROUTINE WRITES THE SUMMARY TABLE OF RUNWAYS,FLIGHTS,AND
C ASSOCIATED INFORMATION
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1   RADI(10,100),NSG(100),NFLGHT
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),XLM1(3,10),
     1   XLM2(3,10),STO(10),SL(10)
      COMMON/RYNME/RYNAM(10)
      DIMENSION XTP(20),NTY(20)
      DATA XDG/1HL/,TKF/1HT/
      READ(5,1000) RYNAM
 1000 FORMAT(10A6)
      WRITE(6,1)
    1 FORMAT(1H1)
      WRITE(6,100)
  100 FORMAT(36X,45(1H*))
      WRITE(6,101)
  101 FORMAT(36X,45H* SUMMARY OF RUNWAYS AND ASSOCIATED FLIGHTS *)
      WRITE(6,102)
  102 FORMAT(36X,45H*   (INCLUDING AIRCRAFT TYPE NUMBERS AND    *)
      WRITE(6,103)
  103 FORMAT(36X,45H*        LANDING/TAKEOFF INFORMATION)        *)
      WRITE(6,100)
      WRITE(6,106)
  106 FORMAT(//118H    FLIGHTS 1    2    3    4    5    6    7    8    9
     1   10   11   12   13   14   15   16   17   18   19   20      TOTAL)
      WRITE(6,107)
  107 FORMAT(8H RUNWAYS)
      DO 500 I=1,NRW
      K=NFT(I)
      DO 400 J=1,K
```

```
      XTP(J)=XDG
      IF(TP(J,I) .EQ. 0.) XTP(J)=TKF
  400 NTY(J)=TY(J,I)
      WRITE(6,120) RYNAM(I),(XTP(J),NTY(J),J=1,K)
  120 FORMAT(1H+,A6,1X,20(1X,A1,I3))
      WRITE(6,121) K
  121 FORMAT(113X,I5/)
      WRITE(6,122)
  122 FORMAT(1H )
  500 CONTINUE
      WRITE(6,125)
  125 FORMAT(113X,5H-----)
      WRITE(6,126) NFLGHT
  126 FORMAT(107X,7HFLIGHTS,I4)
      RETURN
      END
$IBFTC POIX    DECK
      SUBROUTINE POINTS
C SUBROUTINE POINTS FINDS THE NE AT THE POINTS OF A GRID.                00053600
C THE GRID MUST BE COMPOSED OF A FINITE NUMBER OF                        00053650
C EQUALLY SPACED ABSCISSAE AND ORDINATES.THE SPACING                     00053700
C NEED NOT BE THE SAME IN EACH DIRECTION.                                0 053750
C CX AND CY ARE THE NUMBER OF ABSCISSAE AND ORDINATES.                   00053800
C THEY MAY BE POSITIVE OR NEGATIVE BUT SHOULD NOT BE ZERO.               00053850
C X1 AND Y1 ARE THE STARTING VALUES FOR X AND Y.SUBSEQUENT               00053900
C VALUES ARE FORMED BY INCREMENTING BY CX AND CY.                        00053950
C NX AND NY ARE THE TOTAL NUMBER OF X AND Y VALUES                       00054000
C TO BE CONSIDERED.                                                        54050
      COMMON/ARP/ARNV(20,25)
      COMMON/JOE1/WECPFG,CANCEL
      COMMON/GDPAR/CX,CY,X1,Y1,NX,NY
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/BB/R0(3),FLIGHT,PFN,EPNDB                                   54100
      DIMENSION ARRLOC(20)
      LOGICAL WECPFG,CANCEL
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      IF(.NOT. PTGDFG) GO TO 60
      WRITE(6,10)
      ARRLOC(1)=X1
      DO 40 I=2,NX
   40 ARRLOC(I)=ARRLOC(I-1)+CX
      IF(.NOT. MKSFLG) GO TO 18
      CALL FTTOM(ARRLOC,NX)
   18 CONTINUE
      WRITE(6,20) (ARRLOC(I),I=1,NX)
      WRITE(6,21)
   60 CONTINUE
      R0(2)=Y1-CY
      DO 100 JS=1,NY
      R0(2)=R0(2)+CY
      R0(1)=X1-CX
      DO 200 IS=1,NX
      R0(1)=R0(1)+CX
      R0(3)=0.
      IF(WECPFG) GO TO 800
```

```
      CALL EXPNF(DU,X)
      GO TO 810
  800 CALL EXPWF(DU,X)
  810 CONTINUE
      ARRLOC(IS)=X
      ARNV(IS,JS)=X
  200 CONTINUE
      IF(.NOT. PTGDFG) GO TO 100
      YR0=R0(2)
      IF(.NOT. MKSFLG) GO TO 65
      CALL FTTOM(YR0,1)
   65 WRITE(6,30) YR0,(ARRLOC(IS),IS=1,NX)
  100 CONTINUE
   10 FORMAT(37H1***NF/WECPNL/NEF VALUES IN A GRID***//)
   20 FORMAT(7H X VALS,1X,16F7.0/8X,4F7.0)
   21 FORMAT(/8H  Y VALS)
   30 FORMAT(/1X,F7.0,16F7.1/8X,4F7.1)
      RETURN                                                          55000
      END                                                             55050
$IBFTC GENF    DECK
      SUBROUTINE GENFN(M,X,MVAR,XXX)
      DIMENSION TYPE(20),XX(20,10),YY(20,10),NSEGS(20),VAR(10)        00037000
      DATA TYPE/9*1.,4*0.,2*1.,5*1./,((XX(I,J),J=1,10),I=1,20)/       00037050
     1200.,400.,700.,4000.,10000.,16000.,4*0.,200.,600.,1000.,        00037100
     22000.,4000.,6000.,10000.,16000.,2*0.,200.,1000.,16000.,         00037150
     37*0.,200.,400.,1000.,5000.,7000.,10000.,4*0.,200.,400.,1000.,   00037200
     42000.,4000.,10000.,4*0.,200.,400.,1000.,2000.,4000.,5*0.,       00037250
     5200.,3000.,16000.,7*0.,200.,3000.,16000.,7*0.,200.,2000.,       00037300
     616000.,7*0.,20.,40.,60.,80.,100.,5*0.,20.,40.,60.,80.,          00037350
     7100.,5*0.,20.,40.,60.,80.,100.,5*0.,20.,40.,60.,80.,100.,       00037400
     85*0.,200.,400.,1000.,2000.,4000.,20000.,4*0.,200.,400.,1000.,   00037450
     92000.,4000.,20000.,4*0.,200.,500.,1000.,4000.,20000.,           00037500
     15*0.,200.,500.,1000.,4000.,16000.,35*0./
      DATA ((YY(I,J),J=1,10),I=1,20)/129.38,
     1124.38,119.06,100.,91.25,87.5,4*0.,123.13,113.44,108.13,        00037650
     298.75,86.88,80.31,74.06,70.,2*0.,121.89,109.38,83.13,7*0.,      00037700
     3115.5,109.38,99.38,77.5,73.75,70.,4*0.,117.5,112.5,105.63,      00037750
     499.38,92.5,81.88,4*0.,107.5,102.5,95.,                          0 037800
     588.13,78.8,5*0.,136.88,118.75,105.63,7*0.,126.25,107.5,         00037850
     695.,7*0.,115.63,95.,79.38,7*0.,108.75,116.25,121.13,            00037900
     7123.63,124.,5*0.,106.88,111.25,                                   037950
     8114.7,117.7,120.,5*0.,104.75,108.5,111.75,114.88,117.,          00038000
     95*0.,102.50,106.5,110.13,112.5,114.,5*0.,1.88,2.5,6.88,         00038050
     111.88,14.38,15.,4*0.,1.88,2.5,5.63,8.13,9.69,10.,4*0.,          00038100
     2104.,100.,95.,83.75,70.,5*0.,100.,93.,87.5,73.8,                00038150
     360.,35*0./
      DATA NSEGS/5,7,2,5,5,4,3*2,4*4,2*5,2*4,3*0/
      DATA VAR/4H  HX,4H PEN,4H THR,4H    ,4H    ,4H    ,4H    ,
     14H    ,4H    ,4H    /
    2 FORMAT(22H GENFN WAS CALLED WITH  ,A4,2H=0)                      00038400
      IF(X.NE.0.)GO TO 4                                                 38350
C THIS FUNCTION PERFORMS A LINEAR OR LOG-LINEAR INTERPOLATION          00038450
C OR EXTRAPOLATION.TO USE THIS FN PICK TEN OR LESS POINTS              00038500
C FROM A GRAPH IN SUCH A WAY THAT THE GRAPH IS FAIRLY LINEAR           00038550
C IN THE INTERVALS THE POINTS DEFINE,E.G.                                038600
C                                                                        38650
C                                                                        38700
C                                                                        38750
C  Y                                                                     38800
```

```
C  *                                                                    38850
C  *                                                                    38900
C  *         +                                                          38950
C  *      +      +                                                      39000
C  *    +           +                                                  39050
C  * +                  +                                             39100
C  *                         +                                        39150
C  *                               +                                  39200
C  ************************************ X                             00039250
C                                                                      39300
C THE GRAPH MAY BE LINEAR OR LOGARITHMIC IN THE X DIRECTION            00039350
C BUT IS ASSUMED TO BE LINEAR IN THE Y DIRECTION THE ARRAY             00039400
C TYPE(FUNCTION NO.) SHOULD BE 1(ONE) TO INDICATE LOG OR 0 TO          00039450
C INDICATE LINEAR.20 FUNCTIONS ARE ALLOWED.                            0-039500
C THE POINT COORDINATES OF THE TEN POINTS ARE STORED IN THE            00039550
C ARRAYS XX(FN NO.,POINT NO.) AND YY(FN NO.,POINT NO.).NSEGS(FN NO.)   00039600
C SHOULD CONTAIN ONE LESS THAN THE NUMBER OF POINTS CHOSEN.            00039650
C GENFN ITERATES ON THE QUESTION  DOES X LIE IN THE RIGHT OR LEFT      00039700
C HALF OF THE INTERVALS THAT HAVE NOT YET BEEN EXCLUDED                00039750
C UNTIL ONLY ONE INTERVAL REMAINS.                                     39800
      WRITE(6,2) VAR(MVAR)                                             39850
      CALL EXIT                                                        39900
    4 N=NSEGS(M)                                                       39950
      IL=1                                                             40000
      IR=N                                                             40050
C FIND THE MIDDLE INTERVAL                                             40100
      IM=II(IL,IR)                                                     40150
    5 XL=XX(M,IL)                                                      40200
      XM=XX(M,IM)                                                      40250
      XR=XX(M,IR+1)                                                    40300
C DOES X LIE IN THE RIGHT HALF                                         40350
   10 IF(X.GT.XM)GO TO 40                                              40400
C REDEFINE THE RIGHTMOST INTERVAL AND FIND THE NEW MIDDLE              00040450
      IR=IM-1                                                          40500
      IM=II(IL,IR)                                                     40550
C HAS THE CORRECT INTERVAL BEEN LOCATED                                040600
      IF(IL.EQ.IR)GO TO 100                                            40650
      GO TO 5                                                          40700
C REDEFINE THE LEFTMOST INTERVAL TO BE THE MIDDLE                      00040750
C AND FIND THE NEW MIDDLE                                              40800
   40 IL=IM                                                            40850
      IM=II(IL,IR)                                                     40900
C IF THE CORRECT INTERVAL HAS NOT BEEN LOCATED TRY AGAIN               00040950
      IF(IL.NE.IR)GO TO 5                                              41000
  100 CONTINUE                                                         41050
C IT IS KNOWN THAT XX(M,IM)LE X LE XX(M,IM+1)                          00041100
C IT IS POSSIBLE TO INTERPOLATE BETWEEN THESE TWO                      00041150
C VALUES.NOTICE THAT IF X IS,SAY,GREATER THAN ANY OF THE               00041200
C MEMBERS OF XX GENFN WILL AUTOMATICALLY EXTRAPOLATE                   00041250
C SINCE THE RIGHTMOST INTERVAL WILL BE CHOSEN FOR THE                  00041300
C  INTERPOLATION .                                                     41350
      X1=XX(M,IM)                                                      41400
      X2=XX(M,IM+1)                                                    41450
      Y1=YY(M,IM)                                                      41500
      Y2=YY(M,IM+1)                                                    41550
      IF(TYPE(M).EQ.0.)GO TO 150                                       41600
      B=X2/X1                                                          41650
      A=(Y2-Y1)/ALOG10(B)                                              41700
      B=Y1-A*ALOG10(X1)                                                41750
```

```
      XXX=A*ALOG10(X)+B
      GO TO 200                                                    41850
  150 A=(Y2-Y1)/(X2-X1)                                            41900
      B=Y1-A*X1                                                    41950
      XXX=A*X+B
  200 CONTINUE
      RETURN
      END                                                          42070
$IBFTC I        DECK
      FUNCTION II(N1,N2)                                           42100
      II=FLOAT(N2-N1+1)/2.                                         42150
      II=II+N1                                                     42200
      RETURN                                                       42250
      END                                                          42270
$IBFTC VUN      DECK
      SUBROUTINE VUNT(C,A)                                         42300
      COMMON/NCOORD/NC                                             42350
      DIMENSION C(1),A(1)                                          42400
C THIS SUBROUTINE FORMS A UNIT VECTOR U FROM ANY VECTOR C          00042450
      CALL VMAG(A,VM)
      IF(VM.EQ.0.)VM=1.                                            42550
      DO 10 I=1,NC                                                 42600
      C(I)=A(I)/VM                                                 42650
   10 CONTINUE                                                     42700
      RETURN                                                       42750
      END                                                          42800
$IBFTC DOT      DECK
      FUNCTION VDOT(A,B)                                           42850
      COMMON/NCOORD/NC                                             42900
      DIMENSION A(1),B(1)                                          42950
C THIS SUBROUTINE PERFORMS THE VECTOR DOT PRODUCT A.B              00043000
      VDOT=0.                                                      43050
      DO 10 I=1,NC                                                 43100
   10 VDOT=VDOT+A(I)*B(I)                                          43150
      RETURN                                                       43200
      END                                                          43220
$IBFTC CROS     DECK
      SUBROUTINE VCROS(C,A,B)                                      43250
      COMMON/NCOORD/NC                                             43300
      DIMENSION A(1),B(1),C(1)                                     43350
C THIS SUBROUTINE PERFORMS THE VECTOR CROSS PRODUCT,I.E.           00043400
C C=AXB.  C MUST BE DIFFERENT FROM A OR B.                           043450
      C(1)=A(2)*B(3)-A(3)*B(2)                                     43500
      C(2)=A(3)*B(1)-A(1)*B(3)                                     43550
      C(3)=A(1)*B(2)-A(2)*B(1)                                     43600
      RETURN                                                       43650
      END                                                          43700
$IBFTC ADD      DECK
      SUBROUTINE VADD(C,A,B)                                       43750
      COMMON/NCOORD/NC                                             43800
      DIMENSION A(1),B(1),C(1)                                     43850
C THIS SUBROUTINE PERFORMS VECTOR ADDITON AS C=A+B                 00043900
      DO 10 I=1,NC                                                 43950
   10 C(I)=A(I)+B(I)                                               44000
      RETURN                                                       44050
      END                                                          44100
$IBFTC SUB      DECK
      SUBROUTINE VSUB(C,A,B)                                       44150
      COMMON/NCOORD/NC                                             44200
```

```
      DIMENSION A(1),B(1),C(1)                                          44250
C THIS SUBROUTINE PERFORMS VECTOR SUBTRACTION  C=A-B                    00044300
      DO 10 I=1,NC                                                      44350
      C(I)=A(I)-B(I)                                                    44400
   10 CONTINUE                                                          44450
      RETURN                                                            44500
      END                                                               44550
$IBFTC SCL     DECK
      SUBROUTINE VSCL(C,F,A)                                            44600
      COMMON/NCOORD/NC                                                  44650
      DIMENSION C(1),A(1)                                               44700
C THIS SUBROUTINE PERFORMS SCALAR MULTIPLICATION ,C=FA,WHERE            00044750
C F IS A SCALAR AND A,C ARE VECTORS                                     44800
      DO 10 I=1,NC                                                      44850
   10 C(I)=F*A(I)                                                       44900
      RETURN                                                            44950
      END                                                               45000
$IBFTC MAG     DECK
      SUBROUTINE VMAG(A,XXX)
      COMMON/NCOORD/NC                                                  45100
      DIMENSION A(1)                                                    45150
C THIS FUNCTION FINDS THE MAGNITUDE OF A VECTOR A                       00045200
      XXX=SQRT(VDOT(A,A))
      RETURN                                                            45300
      END                                                               45350
$IBFTC NOISE   DECK
      SUBROUTINE BNOISE(TAC,HX)                                         45400
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/BB/R0(3),FLIGHT,PFN,EPNDB                                  45450
      COMMON/BK/DIAG,YES,MAYBE                                          45500
      DIMENSION TOC(20),GTO(20),TOT(20),CL(20),                         00045550
     1GL(20),TL(20),THC(20)                                             45600
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      REAL MAYBE
      DATA TOC/1.,3.,1.,1.,3.,3.,3.,5.,5.,5.,2*16.,7.,7*0./,            00045650
     1GTO/-4.,-3.,3*0.,2*-2.,3.,-10.,2*0.,3.0,8*0./,                    00045700
     2TOT/4*100.,3*93.,3*100.,2*93.,100.,7*0./,                         00045750
     2CL/2.,4.,2*2.,3*4.,3*6.,2*17.,9.,7*0./,                           00045800
     3GL/-8.,-4.,3*0.,2*-2.,3.,3*0.,3.,8*0./,                           00045850
     3TL/42.,33.,5*42.,3*100.,3*42.,7*0./,                              045900
     4THC/11.,13.,2*10.,3*12.,13*0./                                    45950
      Z=0.                                                              46000
C TAC IS THE A/C TYPE NUMBER                                            46050
      I=TAC                                                             46100
C FLIGHT=1 IMPLIES A LANDING                                            46150
      IF(FLIGHT.EQ.1.)GO TO 50                                          46200
      NFN=TOC(I)                                                        46250
      COR=GTO(I)                                                        46300
      THR=TOT(I)                                                        46350
      GO TO 100                                                         46400
   50 NFN=CL(I)                                                         46450
      COR=GL(I)                                                         46500
      THR=TL(I)                                                         46550
  100 CONTINUE                                                          46600
      M=THC(I)                                                          46650
C COR+GENFN(NFN,HX,1) WILL GIVE THE EPNL AT A DISTANCE                  00046700
C HX,NEGLECTING GROUND ATTENUATION,SHIELDING,+CHANGE FROM               00046750
```

```
C THE ASSUMED THRUST(THR).M IS THE THRUST CORRECTION CURVE.          00046800
      CALL GENFN(NFN,HX,1,XQX)
      EPNDB=XQX+COR                                                    46900
C NO THRUST CORRECTION IS MADE FOR A/C TYPE GT 7, SINCE NO           00046950
C DATA ARE AVAILABLE                                                   47000
      IF(I.GE.8)GO TO 900                                              47050
      CALL GENFN(M,PFN,2,X)
      CALL GENFN(M,THR,3,Y)
      Z=X-Y                                                            47200
      EPNDB=EPNDB+Z                                                    47250
  900 IF(DGL1FG) CALL ENT4(NFN,COR,Z,XQX)
 1000 RETURN                                                           47350
      END                                                              47370
$IBFTC DAT      DECK
      SUBROUTINE DATA(J)                                               47400
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),            00047450
     1XLM1(3,10),XLM2(3,10),STO(10),SL(10)                           047500
      COMMON/CC/LAMBD1(3),LAMBD2(3),TAU1,TAU2,WEIGHT,ACTYPE,       00047550
     1DIST,SLOPE,UG(3),SINB,XDIST,GLUMP                               47600
      REAL LAMBD1,LAMBD2                                               47650
C THIS SUBROUTINE DETERMINES CERTAIN QUANTITIES PERTAINING TO        00047700
C THE RUNWAY GEOMETRY FOR RUNWAY NO. J.LAMBD1 + LAMBD2 ARE           00047750
C TWO POINTS ON THE RUNWAY.FLIGHT TAKES PLACE IN THE DIRECTION       00047800
C LAMBD1 GT LAMBD2,WHICH DEFINES THE UNIT VECTOR UG.GLUMP IS         00047870
C THE DISTANCE FROM LAMBD1(ALONG UG) AT WHICH THE A/C BEGINS         00047900
C THE TAKE-OFF ROLL.TAU1 IS THE DISTANCE TO TOUCHDOWN.               00047950
      DO 10 M=1,3                                                      48000
      LAMBD1(M)=XLM1(M,J)                                              48050
      LAMBD2(M)=XLM2(M,J)                                              48100
   10 CONTINUE                                                         48150
      GLUMP=STO(J)                                                     48200
      TAU1=SL(J)                                                       48250
   20 CALL VSUB(UG,LAMBD2,LAMBD1)                                      48300
   30 CALL VUNT(UG,UG)                                                 48350
      RETURN                                                           48400
      END                                                              48420
$IBFTC ATTX     DECK
      SUBROUTINE ATTENU(SB,HX,FTT,XXX)
C THIS FUNCTION GIVES THE GROUND ATTENUATION AS A FUNCTION OF THE    00048500
C DISTANCE TO THE OBSERVER AND THE ELEVATION ANGLE BETA.             00048550
      XXX=0.
      M=14                                                             48650
      B=ARSIN(SB)                                                      48700
      IF(B.GT.0.1208)GO TO 100
      T=SQRT(ABS(TAN(13.*B)))
      IF(FTT.EQ.0.)M=15                                                48850
      CALL GENFN(M,HX,1,YYY)
      TEMP1=EXP(T)
      XXX=YYY/TEMP1
  100 CONTINUE
      RETURN
      END                                                              48970
$IBFTC SLP      DECK
      SUBROUTINE SLPDS(NZ,ACWT,DIST,IE,XXX)
C INPUT                                                              49050
C    NZ,THE AIRCRAFT TYPE (TAKEOFF IF  ,LANDING IF -).               00049100
C    ACWT,THE AIRCRAFT WEIGHT IN POUNDS.                             49150
C OUTPUT                                                             49200
C   SLPDS,THE TANGENT OF THE CLIMBOUT(OR DESCENT )ANGLE.             00049250
```

```
C    DIST THE ROLL DISTANCE FOR TAKEOFF.                                    049300
C    IE AND ERROR INDICATOR.                                                 49350
      COMMON/WTPFV/NACT,WFCTR,PFGWT(2,2,11)                                0 049400
C INITIALIZE SLPDS FOR THE LANDING CASE.                                     049450
      COMMON/TOPRF/NTOP,DFCTR,TOPRF(2,8)                                     049500
      COMMON/TRNS/PI                                                          49550
      COMMON/LN/VLN                                                           49600
      COMMON/ZXX/IPRFL,IPR(100)
      LOGICAL IBND,BND                                                        49700
      TOL=10./WFCTR                                                           49750
      IE=0                                                                    49800
C DETERMINE IF THE AIRCRAFT TYPE IS ACCOMODATED BY THE PROGRAM.          00049850
      DIST=0.                                                                 49900
      XXX=TAN(DGTRD(3.))
      IF(IBND(IABS(NZ),1,NACT))GO TO 10                                       50000
      IE=IF+1
C PRINT THE ERROR MESSAGE AND CHANGE THE ERROR INDICATOR.                00050100
      WRITE(6,100)                                                            50150
      GO TO 40                                                                50200
   10 CONTINUE                                                                50250
      IF(NZ.LT.0)GO TO 40                                                     50300
C IF THIS IS A LANDING RETURN.                                               50350
      IPRFL=NZ-4                                                              50400
C SET UP THE INDEX NZ FOR THE PROPER WT-PROFILE CURVE.                   00050450
      IF(NZ.GE.11.AND.NZ.LE.12)GO TO 30                                       50500
      IF(NZ.GE.13)NZ=NZ-2                                                     50550
      GFW=ACWT/WFCTR                                                          50600
      L1=INT(PFGWT(1,1,NZ)+2.0001)                                            50650
      L2=INT(PFGWT(1,2,NZ)+1.9999)                                            50700
      IF(BND(GFW,PFGWT(2,1,NZ)-TOL,PFGWT(2,2,NZ)+TOL))GO TO 20           00050750
      IE=IE+2                                                                 50800
      L1=1                                                                    50850
      L2=NTOP                                                                 50900
   20 CONTINUE                                                                50950
C FIND THE INDEX FOR THE TAKEOFF PROFILE CURVE.                          00051000
      IPRFL=INT(XINT(PFGWT(1,1,NZ),GFW)+2.0)                             00051050
      IPRFL=MAXO(L1,MINO(L2,IPRFL))                                           51100
   30 CONTINUE                                                                51150
C FIND THE TAKEOFF SLOPE AND ROLL DISTANCE.                              00051200
      XXX=TOPRF(1,IPRFL)
C ZERO SLOPE INDICATES AN ERROR.                                             51300
      IF(XXX .EQ. 0.) GO TO 50
      DIST=-TOPRF(2,IPRFL)/XXX
   40 CONTINUE                                                                51450
      RETURN                                                                  51500
   50 CONTINUE                                                                51550
      IE=IE+4                                                                 51600
      WRITE(6,400)                                                            51650
      GO TO 40                                                                51700
  100 FORMAT(33H ERROR- THERE IS NO CURVE FOR A/C )                        00051750
  200 FORMAT(27H  ERROR-WT IS OUT OF RANGE )
  400 FORMAT(20H ERROR-SLOPE IS ZERO )                                       051850
      END                                                                     51900
$IBFTC IBN     DECK
      LOGICAL FUNCTION IBND(IV,IS,IH)                                        051950
      IBND=.FALSE.                                                            52000
      IF(IV.GE.IS.AND.IV.LE.IH)IBND=.TRUE.                                   052050
      RETURN                                                                  52100
      END                                                                     52120
```

```
$IBFTC BN      DECK
      LOGICAL FUNCTION BND(V,S,H)                                      52150
      BND=.FALSE.                                                      52200
      IF(V.GE.S.AND.V.LE.H)BND=.TRUE.                                 052250
      RETURN                                                           52300
      END                                                              52320
$IBFTC DGT     DECK
      FUNCTION DGTRD(DG)                                               52350
      COMMON/TRNS/PI                                                   52400
      DGTRD=(DG*PI)/180.                                               52450
      RETURN                                                           52500
      END                                                              52520
$IBFTC XIN     DECK
      FUNCTION XINT(PT,Y)                                              52550
      COMMON/LN/VLN                                                    52600
      DIMENSION PT(4)                                                  52650
      YI=PT(4)-PT(2)                                                   52700
      IF(YI.EQ.0.)GO TO 10                                             52750
      XINT=(PT(3)-PT(1))*(Y-PT(2))/YI+PT(1)                           052800
      RETURN                                                           52850
   10 CONTINUE                                                         52900
      IF(Y.NE.PT(2))GO TO 20                                           52950
      XINT=PT(1)                                                       53000
      RETURN                                                           53050
   20 CONTINUE                                                         53100
      WRITE(6,100) Y,PT                                                53150
      XINT=VLN                                                         53200
  100 FORMAT(12H INT. ERROR ,F10.3,6X,2F10.3,3X,2F10.3)
      RETURN                                                           53300
      END                                                              53350
$IBFTC NAC     DECK
      FUNCTION NACM (FL,ACT)                                           53400
      NACM=INT(SIGN(ACT+0.5-FL))                                       53450
      RETURN                                                           53500
      END                                                              53520
C THE SUBROUTINES ENT2 THROUGH ENT5 THAT FOLLOW BELOW                 00055100
C ARE CALLED TO WRITE OPTIONAL OUTPUT TO FILE NO. 1 (OUT).            00055150
C ENT2,ENT3,AND ENT5 ARE CALLED FROM FUNCTION EXPOSE,                 00055200
C AND ENT4 IS CALLED FROM SUBROUTINE BNOISE.THE ENT                   00055250
C SUBROUTINES ARE CALLED ON THE CONDITION THAT                        00055300
C DIAG=YES OR DIAG=MAYBE WHERE YES IS  Y  AND                         00055350
C MAYBE IS  M .DIAG OBTAINS IT VALUE IN SUBROUTINE READIN             00055400
C WHERE THE QUESTION  VERIFICATION AT A POINT   IS POISED             00055450
C TO THE USER.THE PURPOSE OF THESE SUBROUTINES IS                     00055500
C SHOW THE INTERMEDIATE VALUES OBTAINED IN THE CALCULATION            00055550
C OF NE AT A CONTOUR POINT.THESE RESULTS ARE ESPECIALLY               00055600
C USEFULL IN CHECKING HAND CALCULATIONS.                               055650
$IBFTC ET2     DECK
      SUBROUTINE ENT2(CAT,TW,PNF,IXX,FLT)                              055700
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      DIMENSION AX(3),BX(5),CX(5),DX(9),IA(12),IB(12),IC(12)           00055750
      DATA SSX/5HSST  /
      DATA AX/5H2 ENG,5H3 ENG,5H4 ENG/,BX/5H HBPR,5H LBPR,             00055800
     15H PROP,5H EXEC,5H     /,CX/5H TFAN,5H TJET,5H  JET,             00055850
     25H FANS,5H     /,DX/2HAA,2H A,2H B,2H C,2H D,2H E,               00055900
     32HPA,2HPB,2HLD/,IA/3,2,3,3,2,1,1,3,1,1,1,3/,                    00055950
     4IB/1,1,2,5,2,2,2,4,2,4,3,3/,IC/1,1,1,2,1,1,4,3,4,3,5,5/         00056000
  3   FORMAT(/5X,13H A/C TYPE =  ,3A5)
```

```
     4 FORMAT(7X,10H WEIGHT =  ,1X,I6)                                          56100
     5 FORMAT(5X,11H THRUST =   ,I3)                                            56150
     6 FORMAT(6X,12H PROFILE =   ,A2)                                           56200
       IQ=CAT                                                                   56250
       IF(IQ .NE. 13) GO TO 8
       WRITE(6,3) SSX
       GO TO 9
     8 ITEMP1=IA(IQ)
       ITEMP2=IB(IQ)
       ITEMP3=IC(IQ)
       WRITE(6,3) AX(ITEMP1),BX(ITEMP2),CX(ITEMP3)
     9 IQ=TW
       IF(MKSFLG)IQ=TW*.453592+.5
       WRITE(6,4)IQ                                                            56400
       IQ=PNF                                                                   56450
       WRITE(6,5)IQ                                                            56500
       IF(FLT.EQ.1.)GO TO 10                                                    56550
       IX=IXX                                                                   56600
       GO TO 20                                                                 56650
    10 IX=9                                                                     56700
    20 WRITE(6,6)DX(IX)                                                         56750
       RETURN                                                                   56800
       END                                                                      56820
$IBFTC ET3      DECK
       SUBROUTINE ENT3(XQ,NQ,DQ)
       COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
       DIMENSION NQ(3)
       REAL NQ
       LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
     7 FORMAT(1X,17HDAY OPERATIONS=   ,F7.3/1X,17HEVN OPERATIONS=   ,F7.3/1
      1X,17HNGT OPERATIONS=   ,F7.3)
     8 FORMAT(12X,6H H =   ,I6)                                                 56950
     9 FORMAT(9X,9H BETA =   ,F6.2)                                             57000
       WRITE(6,7) (NQ(IJ),IJ=1,3)
       IQ=DQ                                                                    57150
       IF(MKSFLG)IQ=DQ*.3048+.5
       WRITE(6,8)IQ                                                            57200
       WRITE(6,9)XQ                                                            57250
       RETURN                                                                   57300
       END                                                                      57320
$IBFTC ET4      DECK
       SUBROUTINE ENT4(IFN,XCOR,XTH,XEPN)                                       57350
    10 FORMAT(//5X,8H CURVE   ,I2,8HGIVES     ,F6.2,6HEPNDB )                00057400
 16    FORMAT(5X,19H THRUST CORR IS     ,F7.2)
 20    FORMAT(5X,19H CORRECTION  IS     ,F7.2)
       WRITE(6,10)IFN,XEPN                                                      57550
       WRITE(6,20)XCOR                                                          57600
       WRITE(6,16)XTH                                                           57650
       RETURN                                                                   57700
       END                                                                      57720
$IBFTC EX       DECK
       INTEGER FUNCTION IEXT(IS,JS)                                             58650
C THIS FUNCTION GIVES THE EXTENSION NO. IF ONE EXISTS AND IS 0 OTHERWISE00058700
C THIS VALUE THEN SHOULD BE USED WHEN REFERENCING THE                   00058750
C THE ARRAYS OF COMMON/EXTS/FOR INFORMATION ABOUT                       00058800
C THE 1ST EXTENSION(3RD SEGMENT) OF THAT FLIGHT.IF                      00058850
C JEXT(IEXT) IS TWO THEN THE FLIGHT HAS TWO EXTENSIONS                  00058900
C AND THE IEXT+1 POSITION OF THE ARRAYS IN COMMON/EXTS/                 00058950
C SHOULD BE USED TO GET INFORMATION ABOUT THE SECOND EXT.               00059000
```

```
C SHOULD BE USED TO GET INFORMATION ABOUT THE SECOND EXT.                00059000
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
      IEXT=0                                                               59150
      DO 100 KS=1,NEXT                                                     59200
      IF(JRW(KS).NE.JS.OR.IFT(KS).NE.IS)GO TO 100                        0 059250
      IEXT=KS                                                              59300
      GO TO 120                                                            59350
  100 CONTINUE                                                             59400
  120 RETURN                                                               59450
      END                                                                  59470
$IBFTC VCT      DECK
      SUBROUTINE VCTPRP(POS,UNIT,PT,PRP)                                   59500
C THIS SUBROUTINE GIVES A VECTOR PRP THAT IS PERPENDICULAR               00059550
C TO THE LINE DEFINED AS PASSING THROUGH THE POINT POS                   00059600
C AND HAVING UNIT DIRECTION UNIT.FURTHERMORE THE                         00059650
C MAGNITUDE OF PRP IS THE DISTANCE FROM THE POINT HAVING                 00059700
C POSITION VECTOR PT TO THAT LINE.                                          59750
      DIMENSION POS(3),UNIT(3),PT(3),A(3),PRP(3)                         00059800
      CALL VSUB(A,POS,PT)                                                  59850
      UCOF=VDOT(A,UNIT)                                                    59900
      CALL VSCL(PRP,UCOF,UNIT)                                             59950
      CALL VSUB(PRP,A,PRP)                                                 60000
      RETURN                                                               60050
      END                                                                  60100
$IBFTC THR      DECK
      FUNCTION THRST(TH1,TH2,X)                                            60150
C AN A/C HAS THRUST T1 ON ONE SEGMENT AND T2 ON THE NEXT,                00060200
C THE TRANSITION WILL BE SMOOTHED IN SUCH A WAY THAT IT WILL             00060250
C BE 99 PERCENT COMPLETE AFTER THE A/C HAS TRAVELED ABOUT ONE MILE       00060300
C AND APPROACHES 100 PERCENT AS A LIMIT.                                   060350
C THE TRANSITION ALWAYS TAKES PLACE ON THAT SEGMENT                      00060400
C WHICH IS FARTHER FROM THE RUNWAY,WHETHER ON TAKE-OFF                   00060450
C OR LANDING.                                                              60500
      THETA=ABS(X)/5000.                                                   60550
      IF(ABS(X).GT.5000.)THETA=1.                                          60600
      THRST=TH1+THETA*(TH2-TH1)                                            60650
      RETURN                                                               60700
      END                                                                  60750
$IBFTC CNTX     DECK
      SUBROUTINE CONTR
      RETURN
      END
$IBFTC FAIX     DECK
      SUBROUTINE FALLIN(NSEGM,NFL,X)
C THIS SUBROUTINE DETERMINES IF A THE PROJECTED OBSERVER POINT LIES
C WITHIN,TO THE LEFT, TO THE RIGHT OF A SEGMENT OR WHETHER IT IS UNDE-
C FINED. IT SETS UP THE ARRAY IDTOSI AS FOLLOWS- 1 LEFT OF SEGMENT, 2 ON
C SEGMENT, 3 RIGHT OF SEGMENT, 4 UNDEFINED.
C
      COMMON/TRNS/PI
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/MNDIS/IDTOSI(10),DTSXYZ(3,10),DTSMIN(10)
      DIMENSION VEC(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
```

```
     1  PTCOFG,PTALFG
C TEST TO SEE IF LINEAR SEGMENT
      IF(X .NE. -100.) GO TO 100
C LINEAR SEGMENT
C     TEST TO SEE IF FINAL SEGMENT OF FLIGHT
      IF(NSEGM .NE. NSG(NFL)) GO TO 12
C     FINAL SEGMENT
      CALL VSCL(VEC,100000.,UNC(1,NSEGM,NFL))
      CALL VADD(GIMC(1,NSEGM+1,NFL),GIMC(1,NSEGM,NFL),VEC)
   12 CONTINUE
      DO 10 I=1,3
      S1=GIMC(I,NSEGM,NFL)
      S2=GIMC(I,NSEGM+1,NFL)
      IF(ABS(S1-S2) .GT. .1) GO TO 50
   10 CONTINUE
   50 T=DTSXYZ(I,NSEGM)
      IF(T .EQ. -99999.) GO TO 93
      IF(S1 .LT. S2) GO TO 60
      SR=S1
      SL=S2
      K1=2
      GO TO 70
   60 SL=S1
      SR=S2
      K1=1
   70 IF(T .GE. SL .AND. T .LE. SR) GO TO 90
      IF(T .GE. SL .AND. T .GE. SR) GO TO 91
      IF(T .LE. SL .AND. T .LE. SR) GO TO 92
   90 K=2
      GO TO 1000
   91 IF(K1 .EQ. 1) K=3
      IF(K1 .EQ. 2) K=1
      GO TO 1000
   92 IF(K1 .EQ. 1) K=1
      IF(K1 .EQ. 2) K=3
      GO TO 1000
   93 K=4
      GO TO 1000
C HELICAL SEGMENT
C TEST TO SEE IF FINAL SEGMENT OF FLIGHT
  100 TH2PI=2.*PI
      THMX=.75*PI
      IF(NSEGM .NE. NSG(NFL)) THMX=ZETI(NSEGM,NFL)
      IF(X .NE. -99999.) GO TO 900
      K=4
      GO TO 1000
  900 XRD=AMOD(X,TH2PI)
      IF(XRD .GE. 0. .AND. XRD .LE. THMX) K=2
      IF(XRD .LT. 0.) K=1
      IF(XRD .GT. THMX) K=3
 1000 IDTOSI(NSEGM)=K
C TEST FOR DIAGNOSTIC
      IF(.NOT. DGL3FG) GO TO 1500
      WRITE(6,300)
  300 FORMAT(12H FROM FALLIN)
      WRITE(6,310) I,NSEGM,NFL,K1,K
  310 FORMAT(17H I,NSEGM,NFL,K1,K/5I10)
  320 FORMAT(31H S1,S2,T,SR,SL,THMX,X,XRD,TH2PI/10F12.3)
      WRITE(6,320) S1,S2,T,SR,SL,THMX,X,XRD,TH2PI
```

```
 1500 RETURN
      END
$IBFTC HBEX    DECK
      SUBROUTINE HBETA(HX,BETAS,PFN,MAXSEG,OBR,ISEG,NFL)
C THIS SUBROUTINE DETERMINES THE DISTANCE HX FROM OBSERVER OBR TO FLIGHT
C PATH TOGETHER WITH SINE    OF ELEVATION ANGLE BETAS AND PERCENT THRUST
C PFN. OTHER ARGUMENTS ARE
C       MAXSEG   SEGMENT NUMBER THAT YIELDS MINIMUM DISTANCE
C       ISEG     NUMBER OF SEGMENTS IN FLIGHT
C       NFL      NO. OF FLIGHT(CUMULATIVE)
      COMMON/ERRIN/JEICS
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/MNDIS/IDTOSI(10),DTSXYZ(3,10),DTSMIN(10)
      DIMENSION VEC(3),VECSM(3),VEC1,3),OBR(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      NSEG=ISEG
      DO 50 I=1,10
   50 IDTOSI(I)=0
      DO 100 I=1,NSEG
C     LINEAR SEGMENT TEST
      IF(RADI(I,NFL) .NE. 0.) GO TO 80
C LINEAR SEGMENT
      CALL VCTPRP(GIMC(1,I,NFL),UNC(1,I,NFL),OBR,VEC)
      CALL VMAG(VEC,TEMP)
      DTSMIN(I)=TEMP
      CALL VADD(VECSM,VEC,OBR)
      DO 70 II=1,3
   70 DTSXYZ(II,I)=VECSM(II)
C DETERMINE IF PROJECTED POINT WITHIN LINEAR SEGMENT
      CALL FALLIN(I,NFL,-100.)
      GO TO 100
C     HELICAL SEGMENT
   80 CALL HELMN(OBR,I,NFL,VEC,TH)
      CALL VADD(VECSM,VEC,OBR)
      DO 700 II=1,3
  700 DTSXYZ(II,I)=VECSM(II)
C DETERMINE IF TH FALLS WITHIN HELICAL SEGMENT
      CALL FALLIN(I,NFL,TH)
  100 CONTINUE
C CHOOSE CORRECT SEGMENT OF FLIGHT
      JEICS=0
      CALL SEGCH(ISN,NFL,OBR)
      IF(JEICS .EQ. 99) RETURN
      MAXSEG=ISN
      CALL VSUB(VEC,DTSXYZ(1,ISN),GIMC(1,ISN,NFL))
      CALL VMAG(VEC,X)
C IF CHOSEN SEGMENT IS 1 OR 2, NO DISCONTINUITY EXISTS BETWEEN THEM IN
C THRUST VALUES
      IF(ISN .EQ. 1 .OR. ISN .EQ. 2) GO TO 120
C CALCULATE CORRECT THRUST FOR HELICAL PROJECTED POINT
      RXX=RADI(ISN,NFL)
      IF(RXX .EQ. 0.) GO TO 118
      CALL HELMN(OBR,ISN,NFL,VEC,TH)
      X=ABS(RXX)*TH
```

```
  118 CONTINUE
      TH1=PTHRI(ISN-1,NFL)
      TH2=PTHRI(ISN,NFL)
C USE THRST TO SMOOTH DISCONTINUITY BETWEEN SEGMENT AND PRECEDING SEGMNT
      PFN= THRST(TH1,TH2,X)
      GO TO 125
  120 PFN=PTHRI(ISN,NFL)
  125 CONTINUE
C DETERMINE HX AND BETAS
      IF(ISN .EQ. 1 .AND. IDTOSI(1) .EQ. 1) GO TO 140
      CALL VSUB(VEC1,DTSXYZ(1,ISN),OBR)
      CALL VMAG(VEC1,HX)
C     CALCULATE SINE OF ELEVATION ANGLE BETAS
      BETAS=VEC1(3)/HX
      GO TO 145
  140 CALL VSUB(VEC1,GIMC(1,1,NFL),OBR)
      CALL VMAG(VEC1,HX)
      BETAS=0.
  145 CONTINUE
C DIAGNOSTICS FOR SUBROUTINE HBETA
  130 IF(.NOT. DGL3FG) GO TO 250
      WRITE(6,190)
  190 FORMAT(11H FROM HBETA)
      WRITE(6,195) HX,BETAS,PFN,OBR,VEC,VECSM,VEC1,TH
  195 FORMAT(35H HX,BETAS,PFN,OBR,VEC,VECSM,VEC1,TH/(8F15.3))
      WRITE(6,196) MAXSEG,ISEG,NFL,NSEG
  196 FORMAT(21H MAXSEG,ISEG,NFL,NSEG/4I15)
  250 RETURN
      END
$IBFTC F1X      DECK
      FUNCTION F1(X)
C FIRST COMPONENT OF VECTOR FROM OBSERVER TO HELIX
      COMMON/CHELX/R,S,T,RAD,E,F,ALPHA,Q,ZH
      F1=R+ABS(RAD)*(E*COS(X)+Q*F*SIN(X))
      RETURN
      ENTRY F1P(X)
      F1P=ABS(RAD)*(-E*SIN(X)+Q*F*COS(X))
      F1=F1P
      RETURN
      ENTRY F1PP(X)
      F1PP=-F1+R
      F1=F1PP
      RETURN
      END
$IBFTC F2X      DECK
      FUNCTION F2(X)
C SECOND COMPONENT OF VECTOR FROM OBSERVER TO HELIX
      COMMON/CHELX/R,S,T,RAD,E,F,ALPHA,Q,ZH
      F2=S+ABS(RAD)*(F*COS(X)-Q*E*SIN(X))
      RETURN
      ENTRY F2P(X)
      F2P=ABS(RAD)*(-F*SIN(X)-Q*E*COS(X))
      F2=F2P
      RETURN
      ENTRY F2PP(X)
      F2PP=-F2+S
      F2=F2PP
      RETURN
      END
```

```
$IBFTC F3X      DECK
      FUNCTION F3(X)
C THIRD COMPONENT OF VECTOR FROM OBSERVER TO HELIX
      COMMON/CHELX/R,S,T,RAD,E,F,ALPHA,Q,ZH
      F3=T+ABS(RAD)*X*TAN(ALPHA)
      RETURN
      ENTRY F3P(X)
      F3P=ABS(RAD)*TAN(ALPHA)
      F3=F3P
      RETURN
      ENTRY F3PP(X)
      F3PP=0.
      F3=F3PP
      RETURN
      END
$IBFTC SECX     DECK
      SUBROUTINE SEGCH(NSEGM,NFL,OBR)
C THIS SUBROUTINE CHOOSES THE SEGMENT ONTO WHICH THE PROJECTED DISTANCE
C FALLS AND WHICH YIELDS THE MINIMUM DISTANCE.
C       NFL   NUMBER OF FLIGHT(CUMULATIVE)
C       NSEGM RETURNED SEGMENT NUMBER
C
      COMMON/ERRIN/JEICS
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/MNDIS/IDTOSI(10),DTSXYZ(3,10),DTSMIN(10)
      DIMENSION VEC(3),OBR(3)
      DIMENSION NTAB(5)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      D=1000000.
      K1=0
      DO 50 I=1,5
   50 NTAB(I)=0
      J=1
      K2=NSG(NFL)
      DO 100 I=1,K2
      IF(IDTOSI(I) .NE. 2) GO TO 100
      NTAB(J)=I
      J=J+1
  100 CONTINUE
  101 IF(J .EQ. 1) GO TO 200
C PROJECTED DISTANCE FALLS WITHIN AT LEAST ONE SEGMENT
C SPECIAL TEST
      IF(IDTOSI(1) .NE. 1) GO TO 140
      CALL VSUB(VEC,GIMC(1,1,NFL),OBR)
      CALL VMAG(VEC,D)
      NSEGM=1
  140 CONTINUE
      JJJ=J-1
      DO 150 I=1,JJJ
      K1=NTAB(I)
      IF(DTSMIN(K1) .GT. D) GO TO 150
      D=DTSMIN(K1)
      NSEGM=K1
  150 CONTINUE
```

```
      GO TO 300
C PROJECTED DISTANCE NOT WITHIN ANY SEGMENT
  200 IF(IDTOSI(1) .EQ. 1 .AND. IDTOSI(K2) .EQ. 3) GO TO 250
      IF(IDTOSI(1) .EQ. 1) GO TO 225
      IF(IDTOSI(K2) .EQ. 3) GO TO 230
C     WRITE(6,205)
C 205 FORMAT(50H PROGRAM CANNOT DETERMINE MINIMUM DISTANCE SEGMENT)
      JEICS=99
      GO TO 305
  250 IF(DTSMIN(K2) .GT. DTSMIN(1)) GO TO  252
      D=DTSMIN(K2)
      NSEGM=K2
      GO TO 280
  225 D=DTSMIN(1)
      NSEGM=1
      GO TO 280
  230 D=DTSMIN(K2)
      NSEGM=K2
      GO TO 280
  252 D=DTSMIN(1)
      NSEGM=1
  280 CONTINUE
  300 IF(.NOT. DGL3FG) GO TO 305
      WRITE(6,301)
  301 FORMAT(11H FROM SEGCH)
      WRITE(6,302) NSEGM,NFL,K1,K2,J,IDTOSI
  302 FORMAT(25H NSEGM,NFL,K1,K2,J,IDTOSI/15I7)
      WRITE(6,303) DTSMIN
  303 FORMAT(13H DTSMIN ARRAY/10F12.3)
  305 RETURN
      END
$IBFTC PARX    DECK
C THIS SUBROUTINE CALCULATES THE VALUES IN THE CHELX COMMON ARRAY
      SUBROUTINE PARSU1(OBR,NSEGM,NFL)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/HLX/HELCN(3,10,75)
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/CHELX/R,S,T,RAD,E,F,ALPHA,Q,ZH
      DIMENSION VEC(3),VEC1(3),OBR(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      R1=RADI(NSEGM,NFL)
      IF(R1 .EQ. 0.) GO TO 100
      RAD=R1
      ZH=HELCN(3,NSEGM,NFL)
      ALPHA=THETAI(NSEGM,NFL)
      IF(R1 .LT. 0.) Q=-1.
      IF(R1 .GT. 0.) Q=1.
      CALL VSUB(VEC,GIMC(1,NSEGM,NFL),HELCN(1,NSEGM,NFL))
      CALL VUNT(VEC1,VEC)
      E=VEC1(1)
      F=VEC1(2)
      IF(OBR(1) .EQ. -9999.) GO TO 100
      R=HELCN(1,NSEGM,NFL)-OBR(1)
      S=HELCN(2,NSEGM,NFL)-OBR(2)
      T=HELCN(3,NSEGM,NFL)-OBR(3)
```

```
  100 IF(.NOT. DGL3FG) GO TO 300
      WRITE(6,120)
  120 FORMAT(12H FROM PARSU1)
      WRITE(6,130) R,S,T,RAD,E,F,ALPHA,Q,ZH
  130 FORMAT(25H R,S,T,RAD,E,F,ALPHA,Q,ZH/9F12.3)
  300 RETURN
      END
$IBFTC FNHX    DECK
      SUBROUTINE FNHEV(TH,VEC)
C THIS SUBROUTINE FINDS THE VECTOR FROM CENTER OF HELIX TO HELIX FOR ANY
C GIVEN TH(RADIANS) IN I,J,K COORDINATES
      COMMON/CHELX/R,S,T,RAD,E,F,ALPHA,Q,ZH
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      DIMENSION VEC(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      VEC(1)=F1(TH)-R
      VEC(2)=F2(TH)-S
      VEC(3)=F3(TH)-T
      IF(.NOT. DGL3FG) GO TO 100
      WRITE(6,50)
   50 FORMAT(11H FROM FNHEV)
      WRITE(6,60) VEC,TH,R,S,T
   60 FORMAT(18H VEC ARRA,TH,R,S,T/7F12.3)
  100 RETURN
      END
$IBFTC FUDX    DECK
      SUBROUTINE FUDVH(TH,VEC1)
C THIS SUBROUTINE FINDS THE UNIT DERIVATIVE VECTOR TO A HELIX AT A GIVEN
C TH(RADIANS) VALUE.
      COMMON/CHELX/R,S,T,RAD,E,F,ALPHA,Q,ZH
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      DIMENSION VEC(3),VEC1(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      VEC(1)=F1P(TH)
      VEC(2)=F2P(TH)
      VEC(3)=F3P(TH)
      CALL VUNT(VEC1,VEC)
      IF(.NOT. DGL3FG) GO TO 100
      WRITE(6,50)
   50 FORMAT(11H FROM FUDVH)
      WRITE(6,60) VEC,VEC1,TH
   60 FORMAT(12H VEC,VEC1,TH/7F12.3)
  100 RETURN
      END
$IBFTC ELUX    DECK
      SUBROUTINE ELUV(NSEGM,NFL)
C THIS SUBROUTINE CALCULATES THE UNIT VECTOR AT THE INITIAL POINT OF A
C SEGMENT GIVEN THE UNIT VECTOR AT THE TERMINAL POINT OF THE PRECEDING
C SEGMENT AND THE ELEVATION ANGLE OF THE CURRENT SEGMENT.
C          NSEGM    CURRENT SEGMENT NUMBER
C          NFL      FLIGHT NUMBER(CUMULATIVE)
C
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
```

```
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      DIMENSION VEC(3),VEC1(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      K1=NSEGM-1
C FIND PROJECTED VECTOR OF PRECEDING UNIT VECTOR ONTO XY PLANE
      VEC(1)=UNC(1,K1,NFL)
      VEC(2)=UNC(2,K1,NFL)
      VEC(3)=0.
C TAKE ITS UNIT VECTOR
      CALL VUNT(VEC1,VEC)
      TH=THETAI(NSEGM,NFL)
C DEVELOP CURRENT UNIT VECTOR
      UNC(1,NSEGM,NFL)=COS(TH)*VEC1(1)
      UNC(2,NSEGM,NFL)=COS(TH)*VEC1(2)
      UNC(3,NSEGM,NFL)=SIN(TH)
      IF(.NOT. DGL3FG) GO TO 100
      WRITE(6,50)
   50 FORMAT(10H FROM ELUV)
      WRITE(6,55)
   55 FORMAT(60H PRECEDING UNIT VECTOR, NEW ELEVATION ANGLE, NEW UNIT VE
     1CTOR)
      WRITE(6,60) (UNC(K2,K1,NFL),K2=1,3),TH,(UNC(K2,NSEGM,NFL),K2=1,3)
   60 FORMAT(7F12.3)
  100 RETURN
      END
$IBFTC EXTX    DECK
      SUBROUTINE EXTARR
C THIS SUBROUTINE SETS UP THE EXTENSION ARRAYS(OF DIMENSION 100 EACH)
C FOR ALL THE EXTENSIONS IN ONE PROCESSING PASS.
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      REAL N
      K=0
      KS=0
      DO 200 J=1,NRW
      K1=NFT(J)
      DO 100 I=1,K1
C K IS RUNNING TOTAL OF NUMBER OF FLIGHTS(MAXIMUM ALLOWED 25)
      K=K+1
      NOS=NSG(K)
      NOEXT=NOS-2
      IF(NOEXT .EQ. 0) GO TO 100
      DO 50 L=1,NOEXT
      KS=KS+1
C KS IS RUNNING TOTAL OF NUMBER OF EXTENSIONS(MAXIMUM 100)
      ALT(KS)=ZETI(L+2,K)
      CANGLE(KS)=THETAI(L+2,K)
      THRUST(KS)=PTHRI(L+2,K)
      CURV(KS)=RADI(L+2,K)
      JEXT(KS)=NOEXT
```

```
      JRW(KS)=J
      IFT(KS)=I
   50 CONTINUE
  100 CONTINUE
  200 CONTINUE
      NEXT=KS
C TEST FOR DIAGNOSTICS
      IF(.NOT. DGL3FG) GO TO 400
      WRITE(6,300)
  300 FORMAT(12H FROM EXTARR)
      WRITE(6,310) (ALT(L),L=1,KS)
  310 FORMAT(10H ALT ARRAY/(8F15.3))
      WRITE(6,320) (JEXT(L),JRW(L),IFT(L),L=1,KS)
  320 FORMAT(20H JEXT,JRW,IFT ARRAYS/(8I15))
  400 RETURN
      END
$IBFTC HMNDX   DECK
      SUBROUTINE HMIND(THET1,THET2,THET,VEC)
C THIS SUBROUTINE FINDS THE VALUE OF THET YIELDING THE MINIMUM DISTANCE
C FROM A POINT(THE OBSERVER) TO A HELIX, THE COMPONENTS OF THE HELIX ARE
C DEFINED IN THE FUNCTION SUBPROGRAMS F1,F2,F3 TOGETHER WITH THEIR DE- V.
C RIVATIVES. THET=-99999. IMPLIES FAILURE TO CONVERGE.
C MIDWAY ANGLE IS INITIAL APPROXIMATION
      COMMON/NERAP/NOITT,EPS1,EPS2
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      DIMENSION VEC(3)
      DIMENSION XARR(4)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      XARR(1)=ABS(THET2-THET1)/2.
      XARR(2)=THET1
      XARR(3)=THET2
      XARR(4)=2./3.*XARR(1)
      DO 150 J=1,4
      X=XARR(J)
      DO 100 I=1,NOITT
      G=F1(X)*F1P(X)+ F2(X)*F2P(X) + F3(X)*F3P(X)
      G1=F1(X)*F1PP(X) + F1P(X)**2
      G2=F2(X)*F2PP(X) + F2P(X)**2
      G3=F3(X)*F3PP(X) + F3P(X)**2
      GP=G1+G2+G3
      IF(ABS(GP) .LT. EPS2) GO TO 150
      XNEW=X-G/GP
C TEST FOR CONVERGENCE
      IF(ABS(XNEW-X) .LE. EPS1) GO TO 200
C TEST FOR DIAGNOSTICS
      IF(.NOT. DGL3FG) GO TO 75
      WRITE(6,70)
   70 FORMAT(11H FROM HMIND)
      WRITE(6,71) X,G,G1,G2,G3,GP,XNEW
   71 FORMAT(21H X,G,G1,G2,G3,GP,XNEW/8F15.3)
   75 CONTINUE
      X=XNEW
  100 CONTINUE
  150 CONTINUE
      THET=-99999.
      GO TO 400
  200 THET=XNEW
```

```
      VEC(1)=F1(XNEW)
      VEC(2)=F2(XNEW)
      VEC(3)=F3(XNEW)
  400 RETURN
      END
$IBFTC HLMNX   DECK
      SUBROUTINE HELMN(OBR,I,NFL,VEC,TH)
C THIS SUBROUTINE FINDS THE MINIMUM VECTOR FROM THE OBSERVER POSITION TO
C THE HELIX AND ALSO THE ANGLE TH WHICH YIELDS THIS MINIMUM.
C
      COMMON/TPNS/PI
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/MNDIS/IDTOSI(10),DTSXYZ(3,10),DTSMIN(10)
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      DIMENSION OBR(3),VEC(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
C SET THE VALUES IN CHELX COMMON BLOCK TO APPROPRIATE VALUES
      CALL PARSU1(OBR,I,NFL)
C CALL MINIMIZATION ROUTINE
      THET2=ZETI(I,NFL)
      IF(THET2 .EQ. 0. .AND. I .EQ. NSG(NFL)) THET2=.75*PI
      CALL HMIND(0.,THET2,TH,VEC)
      IF(TH .EQ. -99999.) GO TO 280
      CALL VMAG(VEC,TMNDS)
C STORE MAGNITUDE IN DTSMIN ARRAY
      DTSMIN(I)=TMNDS
  280 CONTINUE
C TEST FOR DIAGNOSTICS
      IF(.NOT. DGL3FG) GO TO 400
      WRITE(6,300)
  300 FORMAT(11H FROM HELMN)
      WRITE(6,310) OBR,VEC,TH,TMNDS,THET2
  310 FORMAT(23H OBR,VEC,TH,TMNDS,THET2/(8F15.3))
      WRITE(6,320) I,NFL
  320 FORMAT(10H I,NFL    /2I10)
  400 RETURN
      END
$IBFTC CENHX   DECK
      SUBROUTINE CENHEL(NSEGM,NFL)
C THIS SUBROUTINE FINDS THE COORDINATES OF THE CENTER OF A HELIX(CIRCU-
C LAR).ASSUMED KNOWN IS THE POSITION VECTOR OF THE INITIAL POINT, INI-
C TIAL UNIT VECTOR, AND ELEVATION ANGLE(ABOVE X-Y PLANE) OF HELICAL
C SEGMENT.
C
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/HLX/HELCN(3,10,75)
      DIMENSION PUV(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
C Z-COORDINATE SAME AS THAT OF INITIAL POINT
      HELCN(3,NSEGM,NFL)=GIMC(3,NSEGM,NFL)
C PUV IS PROJECTED UNIT VECTOR ONTO X-Y PLANE
```

```
      DO 10 I=1,3
      PUV(I)=UNC(I,NSEGM,NFL)
      IF(I .EQ. 3) PUV(I)=0.
   10 CONTINUE
      RAD=RADI(NSEGM,NFL)
      ARAD=ABS(RAD)
C YA - YH
      TEMP=PUV(1)*ARAD/SQRT(PUV(1)**2+PUV(2)**2)
C XA - XH
      TEMP3=ABS(PUV(1))
      IF(TEMP3.LT..001) GO TO 15
      TEMP1=-TEMP*PUV(2)/PUV(1)
      GO TO 16
   15 TEMP1=-ARAD
   16 CONTINUE
      CP=PUV(1)*TEMP-PUV(2)*TEMP1
C SIGNS OF CP AND RADIUS AGREE
      IF(CP .GT. 0..AND.RAD .GT. 0. .OR. CP .LT. 0..AND. RAD .LT. 0.)
     1  GO TO 20
C SIGNS DISAGREE
      TEMP=-TEMP
      TEMP1=-TEMP1
      GO TO 20
   20 HELCN(2,NSEGM,NFL)=GIMC(2,NSEGM,NFL)-TEMP
      HELCN(1,NSEGM,NFL)=GIMC(1,NSEGM,NFL)-TEMP1
C DIAGNOSTIC TEST
      IF(.NOT. DGL3FG) GO TO 100
      WRITE(6,30)
   30 FORMAT(12H FROM CENHEL)
      WRITE(6,35) PUV,(HELCN(I,NSEGM,NFL),I=1,3),RAD,ARAD,TEMP,TEMP1,CP
   35 FORMAT(33H PUV,HELCN,RAD,ARAD,TEMP,TEMP1,CP/(8F15.3))
      WRITE(6,40) NSEGM,NFL
   40 FORMAT(10H NSEGM,NFL/2I10)
  100 RETURN
      END
$IBFTC RDOX     DECK
      SUBROUTINE READOU
C DUMMY
      RETURN
      END
$IBFTC EEXPNE  DECK
      SUBROUTINE EXPNE(DUMY,XXX)
      COMMON/ERRIN/JEICS
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),                  00028700
     1XLM1(3,10),XLM2(3,10),STO(10),SL(10)                                28750
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
      COMMON/BK/DIAG,YES,MAYBE                                            29000
      COMMON/TRNS/PI                                                      29050
      COMMON/BB/R0(3),FLIGHT,PFN,EPNDB                                    29100
      COMMON/CC/LAMBD1(3),LAMBD2(3),TAU1,TAU2,WEIGHT,ACTYPE,             00029150
     1DIST,SLOPE,UG(3),SINB,XDIST,GLUMP                                   29200
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
```

```
      COMMON/ZXX/IPRFL,IPR(100)
      REAL N
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      REAL MAYBE
      REAL LAMBD1,LAMBD2
      DATA XLN10/2.302585/                                              29400
      AXE=0.                                                            29450
      NFL=0                                                             29500
      MXX=0                                                             29550
      XNEM=0.                                                           29600
      DO 100 J=1,NRW                                                    29650
      IF(DGL1FG) WRITE(6,1) J
    1 FORMAT(/////12X,12H RUNWAY NO.=,I3/12X,1H ,12(1H*)//)          00029750
    2 FORMAT(12H FLIGHT NO.=,2X,I2/1H ,12(1H*))                        029800
      M=NFT(J)                                                          29850
      DO 100 I=1,M                                                      29900
      FLIGHT=TP(I,J)                                                    29950
      WEIGHT=W(I,J)                                                     30000
      ACTYPE=TY(I,J)                                                    30050
      NFL=NFL+1                                                         30100
      IF(DGL1FG) WRITE(6,2) I
      MX=IEXT(I,J)                                                      30200
      IF(MX.EQ.0)GO TO 20                                               30250
      ISEG=JEXT(MX)+2
      GO TO 25                                                          30350
   20 ISEG=2                                                            30400
   25 IF(DGL2FG) WRITE(6,120) (R0(IL),IL=1,3),ISEG,NFL
C AT THIS POINT,MX IS ZERO AND ISEG IS 2 IF THIS                       030500
C FLIGHT HAS NO EXTENSIONS.IF IT DOES HAVE SOME EXTENSIONS          00030550
C THEN MX IS THE APPROPRIATE POSITION IN THE ARRAYS IN              00030600
C COMMON/EXTS/TO LOOK FOR INFORMATION REGARDING THE                 00030650
C OF SEGMENTS ON THIS FLIGHT.IF ISEG IS 4 THEN THE                  00030700
C INFORMATION FOR THE SECOND EXTENSION MAY BE FOUND                 00030750
C IN POSITION MX+1 OF THE ARRAYS OF COMMON/EXTS/.                   00030800
      CALL HBETA(XDIST,SINB,PFN,MXS,R0,ISEG,NFL)                    0-030850
C HBETA DETERMINED H(=XDIST),SIN(BETA)(=SINB),THE PERCENT           00030950
C THRUST(=PFN),AND THE SEGMENT USED IN CALCULATING THESE            00031000
C (=MXS).                                                              31050
      IF(JEICS .EQ. 99) GO TO 100
      IPRF=IPR(NFL)                                                     31100
      IF(.NOT. DGL1FG) GO TO 28
      CALL ENT2(ACTYPE,WEIGHT,PFN,IPRF,FLIGHT)                       031150
   28 X=ARSIN(SINB)*180./PI                                             31200
C N IS THE NUMBER OF IDENICAL OPERATIONS                                31250
C     Y1=N(I,J)                                                         31300
C     THIS IS A TEMPORARY PATCH TO RESTRICT THE EVENTUAL 3-TYPE
C     OPERATION AS INDICATED IN INPUT TO THE SINGLE TYPE OPERATION OF
C     THE MOD1 VERSION.  N REPLACED BY NTEMP
      Y1=N(1,I,J)+16.67*N(3,I,J)
C     END OF PATCH
      IF(DGL1FG) CALL ENT3(X,N(1,I,J),XDIST)
C BNOISE FINDS THE OBSERVED EPNL NEGLECTING SHIELDING,              00031400
C GROUND ATTENUATION ,AND CHANGES FOR IDENICAL OPERATIONS.          00031450
   30 CALL BNOISE(ACTYPE,XDIST)                                         31500
C THE CORRECTION FOR EXTRA GROUND ATTENUATION IS-                   00031550
      CALL ATTENU(SINB,XDIST,FLIGHT,X)
C THE SHIELDING TERM IS -                                              31650
```

```
      Y=0.
C CALLING THE CORRECTED EPNL EXPOSE ,WE GET -                          031750
      XXX=EPNDB-(X+Y)
C THE CORRECTION FOR Y1 IDENICAL OPERATIONS IS -                      00031850
      Z=10.*ALOG10(Y1)                                                  31900
C THE OBSERVED EPNL IS GIVEN BY XNE -                                   31950
      XNE=XXX+Z
 9000 CONTINUE
      IF(XNE.LT.XNEM)GO TO 90                                           32050
      MFL=NFL                                                           32100
      MXSG=MXS                                                          32150
      MXX=MX                                                            32200
      XNEM=XNE                                                          32250
C ONCE ALL FLIGHTS ARE CONSIDERED AXE WILL CONTAIN THE                00032300
C SUM OF THE ANTILOGS OF ONE TENTH THE OBSERVED EPNLS                 00032350
C CONTRIBUTED FROM EACH FLIGHT.                                         32400
   90 AXE=AXE+EXP(XLN10*(XNE/10.))                                      32450
      IF(DGL1FG) CALL ENT5N(X,Y,Z,XNE,AXE)
  100 CONTINUE                                                          32550
C CALLING THE NOISE EXPOSURE EXPOSE WE GET -                            32600
      XXX=10.*ALOG10(AXE)-88.01
      IF(DGL1FG) WRITE(6,110) XXX
  110 FORMAT(/22X,10H NO. 88.01/22X,1H ,7(1H*)//9X,6H NE IS,21X,F7.2)   00032750
  120 FORMAT(21H FROM EXPOSE TO HBETA /12H RO,ISEG,NFL /1H ,3F12.3,2I4) 00032800
 9100 CONTINUE
      RETURN                                                            32850
      END                                                               32870
$IBFTC ET5N    DECK
      SUBROUTINE ENT5N(X1,Y1,Z1,XNE1,AXE1)
 13   FORMAT(5X,16H ATTENUATION IS ,3X,F7.2)
 14   FORMAT(7X,14H SHIELDING IS ,3X,F7.2)
 15   FORMAT(5X,19H OPERATIONS  IS    ,F7.2)
   17 FORMAT(/22X,7(1H*)/7X,16H NET EPNDB IS      ,F6.2//5X,             00057950
     1 15H EPNDB SUBTOTAL,4X,F6.2)                                      58000
   18 FORMAT(44H AXE1 WAS 0 IN ENT5...SUBTOTALS ARE SPURIOUS )          00058050
      X1=-X1                                                            58100
      Y1=-Y1                                                            58150
      WRITE(6,13)X1                                                     58200
      WRITE(6,14)Y1                                                     58250
      WRITE(6,15)Z1                                                     58300
      IF(AXE1.NE.0.)GO TO 30                                            58350
      AXE1=10.                                                          58400
      WRITE(6,18)                                                       58450
   30 XE1=10.*ALOG10(AXE1)                                              58500
      WRITE(6,17)XNE1,XE1                                               58550
      RETURN                                                            58600
      END                                                               58620
$IBFTC EEXPWE  DECK
      SUBROUTINE EXPWE(DUMY,XXX)
      COMMON/ERRIN/JEICS
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/DENOP/NOPARR(3),CONARR(3),S
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),                 00028700
     1XLM1(3,10),XLM2(3,10),STO(10),SL(10)                              28750
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
```

```
      COMMON/BK/DIAG,YES,MAYBE                                          29000
      COMMON/TRNS/PI                                                    29050
      COMMON/BB/R0(3),FLIGHT,PFN,EPNDB                                  29100
      COMMON/CC/LAMBD1(3),LAMBD2(3),TAU1,TAU2,WEIGHT,ACTYPE,          00029150
     1DIST,SLOPE,UG(3),SINB,XDIST,GLUMP                                 29200
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/ZXX/IPRFL,IPR(100)
      DIMENSION XR0(3)
      DIMENSION AXEARR(3),Y1ARR(3),ZARR(3),ECPNLA(3),XNEARR(3)
      REAL NOPARR
      REAL MAYBE
      REAL LAMBD1,LAMBD2,N                                              29350
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      DATA XLN10/2.302585/                                              29400
      TT1=0.
      TT2=0.
      TT3=0.
      DO 5 IJ=1,3
      ZARR(IJ)=0.
      XNEARR(IJ)=0.
      ECPNLA(IJ)=0.
    5 AXEARR(IJ)=0.
      NFL=0                                                             29500
      MXX=0                                                             29550
      XNFM=0.                                                           29600
      DO 100 J=1,NRW                                                    29650
      IF(DGL1FG) WRITE(6,1) J
    1 FORMAT(/////12X,12H RUNWAY NO.=,I3/12X,1H ,12(1H*)//)           00029750
    2 FORMAT(12H FLIGHT NO.=,2X,I2/1H ,12(1H*))                       0 029800
      M=NFT(J)                                                          29850
      DO 100 I=1,M                                                      29900
      FLIGHT=TP(I,J)                                                    29950
      WEIGHT=W(I,J)                                                     30000
      ACTYPE=TY(I,J)                                                    30050
      NFL=NFL+1                                                         30100
      IF(DGL1FG) WRITE(6,2) I
      MX=IFXT(I,J)                                                      30200
      IF(MX.EQ.0)GO TO 20                                               30250
      ISEG=JEXT(MX)+2
      GO TO 25                                                          30350
   20 ISEG=2                                                            30400
   25 IF(.NOT. DGL2FG) GO TO 26
      DO 27 IJ=1,3
   27 XR0(IJ)=R0(IJ)
      IF(MKSFLG) CALL FTTOM(XR0,3)
      WRITE(6,120) (XR0(IL),IL=1,3),ISEG,NFL
   26 CONTINUE
C AT THIS POINT,MX IS ZERO AND ISEG IS 2 IF THIS                        030500
C FLIGHT HAS NO EXTENSIONS.IF IT DOES HAVE SOME EXTENSIONS            00030550
C THEN MX IS THE APPROPRIATE POSITION IN THE ARRAYS IN                00030600
C COMMON/EXTS/TO LOOK FOR INFORMATION REGARDING THE                   00030650
C OF SEGMENTS ON THIS FLIGHT.IF ISEG IS 4 THEN THE                    00030700
C INFORMATION FOR THE SECOND EXTENSION MAY BE FOUND                   00030750
C IN POSITION MX+1 OF THE ARRAYS OF COMMON/EXTS/.                     00030800
      CALL HBETA(XDIST,SINB,PFN,MXS,R0,ISEG,NFL)                      00030850
      IF(JEICS .EQ. 99) GO TO 100
      IF(.NOT. DGL1FG) GO TO 28
```

```
C HBETA DETERMINED H(=XDIST),SIN(BETA)(=SINB),THE PERCENT                00030950
C THRUST(=PFN),AND THE SEGMENT USED IN CALCULATING THESE                 00031000
C (=MXS).                                                                   31050
      IPRF=IPR(NFL)                                                         31100
      CALL ENT2(ACTYPE,WEIGHT,PFN,IPRF,FLIGHT)                            031150
   28 X=ARSIN(SINB)*180./PI                                                 31200
C N IS THE NUMBER OF IDENICAL OPERATIONS                                  031250
      NOPARR(1)=N(1,I,J)
      NOPARR(2)=N(2,I,J)
      NOPARR(3)=N(3,I,J)
      Y1=NOPARR(1)+NOPARR(2)+NOPARR(3)
      IF(DGL1FG) CALL ENT3(X,NOPARR,XDIST)
C BNOISE FINDS THE OBSERVED EPNL NEGLECTING SHIELDING,                   00031400
C GROUND ATTENUATION ,AND CHANGES FOR IDENICAL OPERATIONS.               00031450
   30 CALL BNOISE(ACTYPE,XDIST)                                             31500
C THE CORRECTION FOR EXTRA GROUND ATTENUATION IS-                        00031550
      CALL ATTENU(SINB,XDIST,FLIGHT,X)
C THE SHIELDING TERM IS -                                                   31650
      Y=0.
C CALLING THE CORRECTED EPNL EXPOSE ,WE GET -                            + 031750
      XXX=EPNDB-(X+Y)
 9000 CONTINUE
      IF(XXX .LT. XNEM) GO TO 90
      MFL=NFL                                                               32100
      MXSG=MXS                                                              32150
      MXX=MX                                                                32200
      XNEM=XXX
   90 CONTINUE
      DO 95 IJ=1,3
      Y1ARR(IJ)=NOPARR(IJ)
      IF(Y1ARR(IJ) .EQ. 0.) GO TO 95
      ZARR(IJ)=10.*ALOG10(Y1ARR(IJ))
      XNEARR(IJ)=XXX+ZARR(IJ)
      AXEARR(IJ)=AXEARR(IJ)+EXP(XLN10*(XNEARR(IJ)/10.))
   95 CONTINUE
      IF(DGL1FG) CALL ENT5W(X,Y,ZARR,XNEARR,AXEARR)
      DO 96 IJ=1,3
   96 XNEARR(IJ)=0.
  100 CONTINUE                                                              32550
      DO 97 IJ=1,3
      IF(AXEARR(IJ) .EQ. 0.) GO TO 97
      ECPNLA(IJ)=10.*ALOG10(AXEARR(IJ))-CONARR(IJ)
   97 CONTINUE
      IF(AXEARR(1) .EQ. 0.) GO TO 300
      TT1=.5*EXP(XLN10*(ECPNLA(1)/10.))
  300 IF(AXEARR(2) .EQ. 0.) GO TO 301
      TT2=.125*EXP(XLN10*((ECPNLA(2) + 5.)/10.))
  301 IF(AXEARR(3) .EQ. 0.) GO TO 302
      TT3=.375*EXP(XLN10*((ECPNLA(3) + 10.)/10.))
  302 CONTINUE
      WECPNL=10.*ALOG10(TT1+TT2+TT3)+S
      XXX=WECPNL
      IF(DGL1FG) WRITE(6,110) ECPNLA,XXX
  110 FORMAT(/22H SUBTRACTIVE CONSTANTS,8H   36.35,10X,5H30.33,10X,5H35.
     111/16H ECPNL FOR D,E,N,F14.2,2F15.2/15H ***WECPNL IS  ,F15.2,4H **
     2*)
  120 FORMAT(21H FROM EXPOSE TO HBETA /12H RO,ISEG,NFL /1H ,3F12.3,2I4) 00032800
 9100 CONTINUE
      RETURN                                                                32850
```

```
      END                                                                32870
$IBFTC ET5W    DECK
      SUBROUTINE ENT5W(X1,Y1,Z1,XNE1,AXE1)
      DIMENSION XE1(3)
      DIMENSION Z1(3),AXE1(3),XNE1(3)
 13   FORMAT(5X,16H ATTENUATION IS ,3X,F7.2)
 14   FORMAT(7X,14H SHIELDING IS ,3X,F7.2)
   18 FORMAT(44H AXE1 WAS 0 IN ENT5...SUBTOTALS ARE SPURIOUS )          00058050
      X1=-X1                                                             58100
      Y1=-Y1                                                             58150
      WRITE(6,13)X1                                                      58200
      WRITE(6,14)Y1                                                      58250
      WRITE(6,15) Z1(1)
      WRITE(6,150) Z1(2)
      WRITE(6,1500) Z1(3)
   15 FORMAT(24H  DAY OPERATIONS  IS    ,F7.2)
  150 FORMAT(24H  EVN OPERATIONS  IS    ,F7.2)
 1500 FORMAT(24H  NGT OPERATIONS  IS    ,F7.2)
      IF(AXE1(1) .NE. 0. .OR. AXE1(2) .NE. 0. .OR. AXE1(3) .NE. 0.)
     1   GO TO 30
      WRITE(6,18)                                                        58450
   30 WRITE(6,35) XNE1
   35 FORMAT(/15H NET EPNDB IS  ,3F15.2)
      DO 1600 I=1,3
      IF(AXE1(I) .EQ. 0.) GO TO 1600
      XE1(I)=10.*ALOG10(AXE1(I))
 1600 CONTINUE
      WRITE(6,36) XE1
   36 FORMAT(15H EPNDB SUBTOTAL,3F15.2)
      RETURN                                                             58600
      END                                                                58620
$IBFTC STORE   DECK
      SUBROUTINE DSTORE                                                  66050
C THIS SUBROUTINE DEVELOPES THE VALUES FOR THE ARRAYS                    00066100
C IN COMMON/XNEW/.                                                       66150
      COMMON/XNEW/GIMC(3,10,75),UNC(3,10,75)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1  PTGDFG,PTCOFG,PTALFG
      COMMON/HLX/HELCN(3,10,75)
      COMMON/CC/LAMBD1(3),LAMBD2(3),TAU1,TAU2,WEIGHT,ACTYPE,             00066300
     1DIST,SLOPE,UG(3),SINB,XDIST,GLUMP                                  66350
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),                  00066500
     1XLM1(3,10),XLM2(3,10),STO(10),SL(10)                               66550
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/TRNS/PI                                                     66650
      COMMON/ZXX/IPRFL,IPR(100)
      COMMON/BK/DIAG,YES,MAYBE                                           66750
      DIMENSION XGIM1(3),XGIM2(3),XGIM3(3),XGIM4(3)
      DIMENSION UV(3)                                                    66800
      DIMENSION XLMBD1(3)
      DIMENSION VEC(3),VEC1(3),VEC2(3)
      DIMENSION VOBR(3)
      REAL NL
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
```

```
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1  PTCOFG,PTALFG
      REAL LAMBD1,LAMBD2
      REAL MAYBE
      NFL=0                                                          66850
      DO 1000 JS=1,NRW                                               66900
      CALL DATA(JS)                                                  66950
      M=NFT(JS)                                                      67000
      DO 1000 IS=1,M                                                 67050
      KS=IEXT(IS,JS)                                                 67100
      MEXT=JEXT(KS)                                                  67150
      NFL=NFL+1                                                      67200
      FLIGHT=TP(IS,JS)                                               67250
      WEIGHT=W(IS,JS)                                                67450
      ACTYPE=TY(IS,JS)                                               67500
      ITEMP1=NACM(FLIGHT,ACTYPE)
C     TEST TO SEE IF TAKEOFF ANGLE AND DIST ARE INPUT
      TEMP1=THETAI(2,NFL)
      IF(TEMP1 .EQ. 0.) GO TO 38
C THEY ARE INPUT
      DIST=ZETI(1,NFL)
      SLOPE=TAN(TEMP1)
      GO TO 39
   38 CONTINUE
      CALL SLPDS(ITEMP1,WEIGHT,DIST,IE,SLOPE)
   39 CONTINUE
      IPR(NFL)=IPRFL                                                 67600
C FLIGHT=1 IMPLIES A LANDING.                                        67650
      IF(FLIGHT.EQ.1.)GO TO 40                                       67700
      IU=1                                                           67750
      GO TO 45                                                       67800
   40 IU=-1                                                          67850
C UG IS IN THE DIRECTION OF FLIGHT.ON BOTH TAKEOFF                   00067900
C AND LANDINGS THE U AND GIM ARRAYS SHOULD INCREASE                  00067950
C IN THE DIRECTION AWAY FROM THE RUNWAY.                             68000
C HENCE FOR LANDING LET UV=-UG.                                      68050
 45   CONTINUE
      FFIU=IU
C     SET UP POSITION AND UNIT VECTORS FOR FIRST TWO SEGMENTS OF FLIGHT
      CALL VSCL(UV,FFIU,UG)
      DO 100 LX=1,3                                                  68150
      IF (FLIGHT.EQ.1.)GO TO 50                                      68200
C ON TAKEOFF GIM1 IS GLUMP FEET DOWN THE RUNWAY.                     0 068250
C GLUMP IS THE DISTANCE TO TAKEOFF ROLL.                             68300
      XLMBD1(LX)=LAMBD1(LX)
      GIMC(LX,1,NFL)=XLMBD1(LX)+GLUMP*UV(LX)
C DIST IS THE ROLL DISTANCE FOR TAKEOFF.                             68400
      GIMC(LX,2,NFL)=GIMC(LX,1,NFL)+DIST*UV(LX)
      GO TO 60                                                       68500
   50 CONTINUE                                                       68550
C TAU1 IS THE DISTANCE TO TOUCHDOWN.                                 68600
C THE AIRCRAFT IS ASSUMED TO ROLL 1 MILE AFTER TOUCHDOWN.            00068650
      XLMBD1(LX)=LAMBD1(LX)
      GIMC(LX,1,NFL)=XLMBD1(LX)-(TAU1+5280.)*UV(LX)
      GIMC(LX,2,NFL)=GIMC(LX,1,NFL)+5280.*UV(LX)
   60 CONTINUE                                                       68800
C SLOPE IS THE TANGENT OF THE CLIMBOUT(OR DESCENT) ANGLE             00068850
C FOUND FROM THE AIRCRAFT WEIGHT AND TYPE.                           68900
      THETA1=ATAN(SLOPE)                                             68950
```

```
C       STORE CLIMBOUT(DESCENT) ANGLE IN INPUT ELEVATION ARRAY
        THETAI(2,NFL)=THETA1
        SINX=SIN(THETA1)                                                    69000
        UNC(LX,1,NFL)=UV(LX)
        UNC(LX,2,NFL)=COS(THETA1)*UV(LX)
        IF(LX .EQ. 3) UNC(LX,2,NFL)=UNC(LX,2,NFL)+SINX
  100 CONTINUE
C IF NO EXTENSIONS EXIST, GO TO STATEMENT 1000
        IF(KS .EQ. 0)GO TO 1000
        NOSG=NSG(NFL)
        NOXT=NOSG-2
        DO 900 LX=3,NOSG
C DETERMINE IF PRECEDING SEGMENT LINEAR. IF NOT GO TO STATEMENT 700
        IF(RADI(LX-1,NFL) .NE. 0.) GO TO 700
C PRECEDING SEGMENT LINEAR
        TEMP=ZETI(LX-1,NFL)
        TEMP1=THETAI(LX-1,NFL)
        TEMP1=TEMP*TAN(TEMP1)
        TEMP=SQRT(TEMP**2+TEMP1**2)
        CALL VSCL(VEC,TEMP,UNC(1,LX-1,NFL))
        CALL VADD(GIMC(1,LX,NFL),VEC,GIMC(1,LX-1,NFL))
        CALL ELUV(LX,NFL)
        GO TO 750
C PRECEDING SEGMENT NON-LINEAR
  700 TH=ZETI(LX-1,NFL)
C       SET INDICATOR TO INDICATE OBSERVER NOT RELEVANT
        VOBR(1)=-9999.
        CALL PARSU1(VOBR,LX-1,NFL)
        CALL FNHEV(TH,VEC)
        CALL VADD(VEC,VEC,HELCN(1,LX-1,NFL))
        DO 701 IJ=1,3
  701 GIMC(IJ,LX,NFL)=VEC(IJ)
        CALL FUDVH(TH,VEC)
        DO 702 IJ=1,3
  702 UNC(IJ,LX-1,NFL)=VEC(IJ)
        CALL ELUV(LX,NFL)
C DETERMINE IF CURRENT SEGMENT IS NON-LINEAR
  750 IF(RADI(LX,NFL) .EQ. 0.) GO TO 800
C CURRENT SEGMENT NON-LINEAR. AT THIS POINT THE POSITION VECTOR OF THE
C INITIAL POINT, THE INITIAL UNIT VECTOR OF THE SEGMENT, AND THE ELEVA-
C TION ANGLE ARE ALL KNOWN.
C FIND CENTER OF HELIX
        CALL CENHEL(LX,NFL)
  800 CONTINUE
  900 CONTINUE
 1000 CONTINUE
C IF DIAGNOSTIC LEVEL 1 WANTED, OUTPUT    SEGMENT POS. AND UNIT VECTORS
        IF(.NOT. DGL1FG .AND. .NOT. PTTBFG) GO TO 2000
        WRITE(6,1600)
        WRITE(6,1604)
        NFL=0
        DO 1570 IJ=1,NRW
        IJK=NFT(IJ)
        DO 1570 JI=1,IJK
        NFL=NFL+1
        K1=NSG(NFL)-2
        WRITE(6,1601) IJ,JI,K1
        WRITE(6,1602)
        K1=K1+2
```

```
      DO 1570 K2=1,K1
      DO 1565 K3=1,3
      VEC1(K3)=GIMC(K3,K2,NFL)
 1565 VEC2(K3)=HELCN(K3,K2,NFL)
      IF(RADI(K2,NFL) .EQ. 0.) GO TO 1560
      IF(MKSFLG) CALL FTTOM(VEC2,3)
      WRITE(6,1606) VEC2
 1560 CONTINUE
      IF(MKSFLG) CALL FTTOM(VEC1,3)
      WRITE(6,1605) K2,PTHRI(K2,NFL),VEC1,(UNC(K3,K2,NFL),K3=1,3)
 1570 CONTINUE
      PTTBFG=.FALSE.
 1600 FORMAT(1H1)
 1601 FORMAT(/8H RUNWAY ,I2,10H   FLIGHT ,I2,20H   NO. OF EXTENSIONS,I3)
 1602 FORMAT(120H SEG  THRUST     GAMMA X      GAMMA Y      GAMMA Z       U
     1NIT X       UNIT Y       UNIT Z HELIX CEN X HELIX CEN Y HELIX CEN Z)
 1604 FORMAT(120H ***POSITION VECTORS OF END POINTS OF SEGMENTS, UNIT VE
     1CTORS OF SEGMENTS, AND POSITION VECTORS OF HELIX CENTERS***       )
 1605 FORMAT(1H ,I3,F8.1,6F12.3)
 1606 FORMAT(1H+,83X,3F12.3)
 2000 RETURN
      END
$IBFTC SUB50F  DECK
      SUBROUTINE READIN(INDEX)
      COMMON/JOE3/JUMP
      COMMON/JOE2/NEFFLG,AREAFG
      COMMON/JOE1/WECPFG,CANCEL
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),XLM1(3,10),XLM2(3      AA
     2,10),STO(10),SL(10)                                                     AA
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/FACTR/AXMAX,AXMIN,AYMAX,AYMIN,DELX,DELY,AINCHX,AINCHY,
     2SETPLT
      LOGICAL NEFFLG,AREAFG
      LOGICAL SETPLT
      LOGICAL CONVRT
      LOGICAL MKSFLG,PTINFG,PTTBFG,PTOUFG,PTGDFG,PTCOFG,PTALFG,DGL3FG,
     2DGL2FG,DGL1FG,DGL0FG,CALFLG,SSIFLG,LSIFLG,HESGFG,CONTFG,SYMFLG,
     3LINEFG,GRIDFG,RWAYFG
      LOGICAL CANCEL,WECPFG
      DATA PLOTS/6HSETPLT,6HDELTAX,6HDELTAY,6HAINCHX,6HAINCHY,6HAXMAXV,6
     2HAXMINV,6HAYMAXV,6HAYMINV/
      DATA  HEADER/6HTITLEE/
      DATA  CONTRL/6HFEETLB,6HMKSSYS,6HPRTINP,6HPRTTAB,6HPRTOUT,6HAREACL    DATA1
     2,6HPRTCTR,6HPRTALL,6HNEFCAL,6HDIAGL1,6HDIAGL2,6HCLCMPS,6HCLCMPL,6H    DATA1
     3DIAGL3,6HWECPNL,6HNEEVAL/
      DATA RUNWAY/6HRUNWAY,6HXCOORD,6HYCOORD,6HZCOORD,6HTKGDRL,6HTCHDWN,     DATA2
     26HNOFLTS,6H      ,6H      /
      DATA FLIGHT/6HFLIGHT,6HLANDNG,6HTAKOFF,6HACWGHT,6HTHRUST,6HACTYPE,     DATA3
     26HNDAYOP,6HNEVNOP,6HNNGTOP,6HNOEXTS,6HCLIMBA,6HPRSEG2,6HPRSEG1/       DATA
      DATA EXTNSN/6HEXTNSN,6HEXTENT,6HELEVAT,6HTHRUST,6HRADIUS,6H      /     DATA4
      DATA GRID/6HGRIDCL,6HDELTAX,6HDELTAY,6HFIRSTX,6HFIRSTY,6HNOOFXS,6H
     2NOOFYS,6H      ,6H      /                                              DATA6
      DATA CONTOR/6HCONTOR,6HNOISEV,6HDELTAX,6HFIRSTX,6HNOOFXS,6HTOLNCE,     DATA7
     26HYAPPRX,6HSYMMET,6HXCOORD,6HYCOORD,6HRXAXIS,6HRYAXIS,6H       /      DATA7
      DATA RUNNFO/6HPROCES,6HREPEAT,6HENDRUN,6HNEWSET,6H      ,6H      /     DATA8
      DIMENSION PLOTS(9)
      DIMENSION CONTRL(16),RUNWAY(9),FLIGHT(13),EXTNSN(6),GRID(9),CONTOR     DIM1A
     2(13),RUNNFO(6)                                                         DIM1B
      INTEGER ERROR,GROUP
```

```
      DIMENSION STORE(5000)
      DIMENSION PSEUDO(72)
      DIMENSION CLIMBA(100)
      DIMENSION ICHECK(7),ICAT(7)
      EQUIVALENCE (CONTRL(1),PSEUDO(1)),(RUNWAY(1),PSEUDO(17)),(FLIGHT(1   EQUIV1
     2),PSEUDO(26)),(EXTNSN(1),PSEUDO(39)),(GRID(1),PSEUDO(45)),(CONTOR(   EQUIV2
     31),PSEUDO(54)),(RUNNFO(1),PSEUDO(67))
C   THE EQUIVALENCE IS SET UP TO CUT DOWN ON RUN TIME SEARCHING FOR A MATCH  TO
C   EACH OF THE ALPHABETIC INPUT CONTROL VARIABLES THIS WILL ALLOW US TO COMPARE
C   ONLY PARTICULAR SECTIONS OF PSEUDO
      DATA ICHECK/0,0,5,5,4,6,6/
      REAL N
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/GDPAR/CX,CY,X1,Y1,NX,NY
      COMMON/QR/TOL,VAL,MX,APPROX,BASEX,XDEL
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,     LOGFG2
     2PTGDGG,PTCOFG,PTALFG                                               LOGFG2
      COMMON/LOGFG3/RWAYFG,GRIDFG,CONTFG
      COMMON/JOE/SYMFLG,REFLIN,XCOORD(2),YCOORD(2),TITLE(12)
      ASSIGN 131 TO ERROR
      IF(JUMP.NE.0) READ(5,1000) ALPHA
      IF(JUMP.NE.0) GO TO 2040
C     OVER ALL CONTROL SECTION
      IPHIL=0
      IST=1
      ASSIGN 10 TO ERROR
      GROUP=2
   10 READ(5,1000) ALPHA
      WRITE(6,1003) ALPHA
      STORE(IST)=ALPHA
      STORE(IST+1)=-1.
      IST=IST+2
C     CHECK FOR TITLE
      IF(ALPHA.EQ.HEADER) READ(5,1000) (TITLE(I),I=1,12)
      IF(ALPHA.EQ.HEADER)
     1 WRITE(6,5050) (TITLE(I),I=1,12)
 5050 FORMAT(1X,12A6)
      IF(ALPHA.EQ.RUNNFO(4).OR.JUMP.EQ.0) GO TO 500
 1000 FORMAT(12A6)
      IF(ALPHA.EQ.HEADER) GO TO 10
      IF(ALPHA.EQ.RUNWAY(1)) GO TO 50
      DO 20 J=1,16
      IF(ALPHA.EQ.PSEUDO(J    )) GO TO (21,22,23,24,25,26,27,28,29,30,31
     2,32,33,34,35,36),J
C     PUT  ERROR MESSAGES IN HERE FOR WHEN THERE ARE NO MATCHES
C     A COUNTER IS KEPT UNTIL THE PROCESS CARD IS ENCOUNTERED IF THE COUNTER
C     IS NOT  0 NE STOP BECAUSE OFFATAL ERRORS
   20 CONTINUE
      IF(ALPHA.NE.PLOTS(1)) GO TO 2000
  200 SETPLT=.TRUE.
      DO 203 I=1,8
      READ(5,1002) ALPHA,DUMMY1
      STORE(IST)=ALPHA
      STORE(IST+1)=DUMMY1
      STORE(IST+2)=-1.
      IST=IST+3
      DO 202 J=1,9
```

```
      IF(ALPHA.EQ.PLOTS(J)) GO TO (204,205,206,207,208,209,210,211,212),J
     2J
  202 CONTINUE
      WRITE(6,300)
  300 FORMAT(56H SOME VARIABLE USED WITH SETPLT IS MISSING OR MISSPELLED
     2)
      CANCEL=.TRUE.
      GO TO 2000
  204 SETPLT=.TRUE.
      GO TO 203
  205 DELX=DUMMY1
      GO TO 203
  206 DELY=DUMMY1
      GO TO 203
  207 AINCHX=DUMMY1
      GO TO 203
  208 AINCHY=DUMMY1
      GO TO 203
  209 AXMAX=DUMMY1
      GO TO 203
  210 AXMIN=DUMMY1
      GO TO 203
  211 AYMAX=DUMMY1
      GO TO 203
  212 AYMIN=DUMMY1
  203 CONTINUE
      GO TO 10
 2000 IF(ALPHA.EQ.FLIGHT(1)) GO TO 65
 2010 IF(ALPHA.EQ.EXTNSN(1)) GO TO 85
 2020 IF(ALPHA.EQ.GRID(1)) GO TO 95
 2030 IF(ALPHA.EQ.CONTOR(1)) GO TO 110
 2040 IF(ALPHA.EQ.RUNNFO(1)) GO TO 130
 2050 IF(ALPHA.EQ.RUNNFO(2)) GO TO 131
 2060 IF(ALPHA.EQ.RUNNFO(3)) GO TO 140
 2070 IF(ALPHA.EQ.RUNNFO(4)) GO TO 136
 2080 IF(ALPHA.EQ.RUNNFO(5)) GO TO 154
 2090 IF(ALPHA.EQ.RUNNFO(6)) GO TO 155
      IF(ALPHA.EQ.RUNWAY(1)) GO TO  50
C     IF WE GET HERE THERE IS AN ERROR IN THE INITIAL IDENTIFIER
 1003 FORMAT(1X,A6,2X,F15.5,2X,F15.5)
      IPHIL=500
       WRITE(6,5000) ALPHA,GROUP
      CANCEL=.TRUE.
 5000 FORMAT(24H FATAL ERROR VARIABLE = ,A6,14HCHECK CATEGORY,I3)
      GO TO ERROR,(10,51,66,86,96,111,131)
   21 MKSFLG=.FALSE.
      GO TO 10
   22 MKSFLG=.TRUE.
      GO TO 10
   23 PTINFG=.TRUE.
      GO TO 10
   24 PTTBFG=.TRUE.
      GO TO 10
   25 PTOUFG=.TRUE.
   27 PTCOFG=.TRUE.
      IF(ALPHA.NE.CONTRL(8)) PTALFG=.FALSE.
      GO TO 10
   26 AREAFG=.TRUE.
      GO TO 10
```

```
   28 PTALFG=.TRUE.
      PTTBFG=.TRUE.
      PTINFG=.TRUE.
C
C     BRANCH TO SET UP THE REST OF THE PRINT FLAGS
      GO TO 25
   34 DGL3FG=.TRUE.
   31 DGL2FG=.TRUE.
   30 DGL1FG=.TRUE.
      DGL0FG=.FALSE.
      GO TO 10
   32 CALFLG=.TRUE.
      SSIFLG=.TRUE.
      GO TO 10
   33 CALFLG=.TRUE.
      LSIFLG=.TRUE.
      GO TO 10
   29 CONTINUE
      NEFFLG=.TRUE.
      GO TO 10
   35 CONTINUE
      WECPFG=.TRUE.
      GO TO 10
   36 CONTINUE
      WECPFG=.FALSE.
      GO TO 10
   50 NUMRUN=NUMRUN+1
      NUMFLI=0
      ASSIGN 51 TO ERROR
      GROUP=3
   51 READ(5,1002) ALPHA,DUMMY1,DUMMY2
      STORE(IST+1)=-1.
 1002 FORMAT(A6,2X,F15.5,2X,F15.5)
      WRITE(6,1003) ALPHA,DUMMY1,DUMMY2
      STORE(IST)=ALPHA
 1001 FORMAT(A6,F10.2,6X,F10.2)
      DO 52 J=17,25
      JUMP2=J
      IF(ALPHA.EQ.PSEUDO(J)) JUMP2=J-16
      IF(JUMP2.NE.J) GO TO (53,54,55,56,57,58,59,60,61),JUMP2
   52 CONTINUE
      STORE(IST+1)=-1.
      IST=IST+2
C     NO MATCH FOR A VARIABLE WITHIN THE RUNWAY SET CHECK TO SEE IF IT IS A NEW
C     IDENTIFIER OTHER THAN A RUNWAY IDENTIFIER
      GO TO 2000
   53 NUMRUN=NUMRUN+1
       IST=IST+2
      GO TO 511
C SETTING OF ENDPOINTS OF RUNWAYS
   54 XLM1(1,NUMRUN)=DUMMY1
      XLM2(1,NUMRUN)=DUMMY2
      ICAT(GROUP)=ICAT(GROUP)+1
      GO TO 511
   55 XLM1(2,NUMRUN)=DUMMY1
      XLM2(2,NUMRUN)=DUMMY2
      ICAT(GROUP)=ICAT(GROUP)+1
      GO TO 511
   56 XLM1(3,NUMRUN)=DUMMY1
```

```
      XLM2(3,NUMRUN)=DUMMY2
      GO TO 511
   57 STO(NUMRUN)=DUMMY1
      ICAT(GROUP)=ICAT(GROUP)+1
      GO TO 5111
   58 SL(NUMRUN)=DUMMY1
      ICAT(GROUP)=ICAT(GROUP)+1
      GO TO 5111
   59 CONTINUE
      NOFLTS=IFIX(DUMMY1)
      NFT(NUMRUN)=NOFLTS
      ICAT(GROUP)=ICAT(GROUP)+1
      GO TO 5111
   60 CONTINUE
C     EXPANSION
      GO TO 511
   61 CONTINUE
      GO TO 511
  511 CONTINUE
      STORE(IST+1)=DUMMY1
      STORE(IST+2)=DUMMY2
      STORE(IST+3)=-1.
      IST=IST+4
      GO TO 51
 5111 STORE(IST+1)=DUMMY1
      STORE(IST+2)=-1.
      IST=IST+3
      GO TO ERROR,(10,51,66,86,96,111,131)
C     EXPANSION
C LOOP TO SETUP FLIGHT PARAMETERS
   65 NUMFLI=NUMFLI+1
      NFLGHT=NFLGHT+1
      NEXTS=0
      ASSIGN 66 TO ERROR
      GROUP=4
   66 READ(5,1002) ALPHA,DUMMY1
      WRITE(6,1003) ALPHA,DUMMY1
      STORE(IST)=ALPHA
      STORE(IST+1)=-1.
      DO 67 J=26,38
      JUMP2=J
      IF(ALPHA.EQ.PSEUDO(J)) JUMP2=J-25
      IF(JUMP2.NE.J) GO TO (68,69,70,71,72,73,74,75,76,77,78,79,80)
     2,JUMP2
   67 CONTINUE
      IST=IST+2
16000 FORMAT(55H NO. OF FLIGHTS PROVIDED DOESNT AGREE WITH NUMBERSTATED)
      IF(NUMFLI.NE.NOFLTS) WRITE(6,16000)
      IF(NUMFLI.NE.NOFLTS)CANCEL=.TRUE.
      IF(CLIMBA.NE.0..AND.CHECK.EQ.0.) GO TO 83
      IF(ZETI(1,NFLGHT).NE.0..AND.CHECK.EQ.0.) GO TO 83
      GO TO 2000
   68 NUMFLI=NUMFLI+1
       NFLGHT=NFLGHT+1
      IST=IST+2
      GO TO 66
   69 TP(NUMFLI,NUMRUN)=1.
      ICAT(GROUP)=ICAT(GROUP)+1
      IST=IST+2
```

```
      GO TO 66
   70 TP(NUMFLI,NUMRUN)=0.
      ICAT(GROUP)=ICAT(GROUP)+1
C     THIS IS A TAKEOFF
      IST=IST+2
      GO TO 66
   71 W(NUMFLI,NUMRUN)=DUMMY1
C     IF NECESSARY CONVERT BETWEEN FEETLB AND MKS SYSTEMS
      GO TO 666
   72 PT(NUMFLI,NUMRUN)=DUMMY1
      PTHRI(1,NFLGHT)=PT(NUMFLI,NUMRUN)
      PTHRI(2,NFLGHT)=PT(NUMFLI,NUMRUN)
      ICAT(GROUP)=ICAT(GROUP)+1
C  PERCENTAGE THRUST OVER SEGMENTS 1 AND 2
      GO TO 666
   73 TY(NUMFLI,NUMRUN)=DUMMY1
C     TYPE OF PLANE
      ICAT(GROUP)=ICAT(GROUP)+1
      GO TO 666
   74 N(1,NUMFLI,NUMRUN)=DUMMY1
      ICAT(GROUP)=ICAT(GROUP)+1
C     NUMBER OF DAY OPERATIONS FOR EACH FLIGHT AND EACH RUNWAY
      GO TO 666
   75 IF(WECPFG) GO TO 755
      WRITE(6,10000)
10000 FORMAT(41H EVENING OPERATIONS CURRENTLY NOT ALLOWED)
      WRITE(6,10001)
10001 FORMAT(59H THESE FLIGHTS WILL BE IGNORED ONLY THE DAY OPS WILL BE
     2RUN)
      GO TO 666
C     WHEN AVAILABLE CHANGE THIS SECTION FOR EVENING FLIGHTS
C     MAKE  75 A CONTINUE
C     N(2,NUMFLI,NUMRUN)=DUMMY1
  755 N(2,NUMFLI,NUMRUN)=DUMMY1
      GO TO 666
   76 IF(WECPFG) GO TO 766
      IF(NFFFLG) GO TO 766
      WRITE(6,10002)
10002 FORMAT(32H NIGHT OPS CURRENTLY NOT ALLOWED)
      WRITE(6,10001)
      GO TO 666
C
C     CHANGES WHEN AVAILABLE
C
C     N(3,NUMFLI,NUMRUN)=DUMMY1
  766 N(3,NUMFLI,NUMRUN)=DUMMY1
      GO TO 666
   77 NOEXTS=IFIX(DUMMY1)
      NSG(NFLGHT)=NOEXTS+2
      GO TO 666
   78 CONTINUE
      CLIMBA(NFLGHT)=DUMMY1/180.*3.1415927
      THETAI(2,NFLGHT)=CLIMBA(NFLGHT)
      GO TO 666
   79 CONTINUE
      ZETI(2,NFLGHT)=DUMMY1
      IF(.NOT. MKSFLG) GO TO 666
      CALL MTOFT(ZETI(2,NFLGHT),1)
      GO TO 666
```

```
   80 CONTINUE
      ZETI(1,NFLGHT)=DUMMY1
      IF(.NOT. MKSFLG) GO TO 666
      CALL MTOFT(ZETI(1,NFLGHT),1)
      GO TO 666
C
   83 CONTINUE
      WRITE(6,777) THETAI(2,NFLGHT),ZETI(1,NFLGHT)
  777 FORMAT(25H CLIMBA OR PRSEG1 MISSING,2(2X,F15.5))
      CANCEL=.TRUE.
      GO TO 2000
  666 CONTINUE
      STORE(IST+1)=DUMMY1
      STORE(IST+2)=-1.
      IST=IST+3
      GO TO 66
   85 CONTINUE
C     WE ARE NOW IN THE EXTENSION SECTION
      NEXTS=NEXTS+1
      ASSIGN 86 TO ERROR
      GROUP=5
      IF(J.NE.39) GO TO 86
      IF(RADI(NEXTS+1,NFLGHT).NE.0.) ZETI(NEXTS+1,NFLGHT)=ZETI(NEXTS+1,
     2NFLGHT)/180.*3.1415927
      LOC=NEXTS+1
      IF(RADI(NEXTS+1,NFLGHT).EQ.0..AND.MKSFLG) CALL MTOFT(ZETI(LOC,NFLG
     2HT),1)
      IST=IST+2
   86 READ(5,1002) ALPHA,DUMMY1
      WRITE(6,1003) ALPHA,DUMMY1
      STORE(IST)=ALPHA
      STORE(IST+1)=-1.
      DO 87 J=39,44
      JUMP2=J
      IF(ALPHA.EQ.PSEUDO(J)) JUMP2=J-38
      IF(JUMP2.NE.J) GO TO (85,88,89,90,91,92,93),JUMP2
   87 CONTINUE
      IST=IST+2
      IF(RADI(NEXTS+2,NFLGHT).NE.0.) ZETI(NEXTS+2,NFLGHT)=ZETI(NEXTS+2,
     2NFLGHT)/180.*3.1415927
      LOC=NEXTS+2
      IF(RADI(NEXTS+2,NFLGHT).EQ.0..AND.MKSFLG) CALL MTOFT(ZETI(LOC,NFLG
     2HT),1)
15000 FORMAT(61H NUMBER OF EXTENSIONS PROVIDED DOES NOT AGREE WITH NO. S
     2TATED)
      IF(NEXTS.NE.NOEXTS) WRITE(6,15000)
      IF(NEXTS .NE.NOEXTS) CANCEL=.TRUE.
      GO TO 2000
C     EXTENT   GROUND PROJECTION DISTANCE OF LINEAR SEGMENT OR ANGULAR EXTENT
C     IN DEGREES IF HELICAL SEGMENT
   88 ZETI(NEXTS+2,NFLGHT)=DUMMY1
      GO TO 866
   89 THETAI(NEXTS+2,NFLGHT)=DUMMY1/180.*3.1415927
C     THIS WAS READ IN AS DEGREES AND IS NOW IN RADIANS
      GO TO 866
   90 PTHRI(NEXTS+2,NFLGHT)=DUMMY1
C     PERCENTAGE THRUST OVER EACH SEGMENT
      GO TO 866
   91 RADI(NEXTS+2,NFLGHT)=DUMMY1
```

```
C      IF RADI =  0  THIS IS A LINEAR SEGMENT  OTHERWISE EXTENT IS IN DEGREES
       IF(RADI.NE.0.) HESGFG=.TRUE.
C
        GO TO 866
   92 CONTINUE
        GO TO 866
   93 CONTINUE
        GO TO 866
  866 CONTINUE
       STORE(IST+1)=DUMMY1
       STORE(IST+2)=-1.
       IST=IST+3
       GO TO 86
C      WE ARE NOW PROCESSING THE GRID INFORMATION
   95 CONTINUE
       ASSIGN 96 TO ERROR
       GROUP=6
       GRIDFG=.TRUE.
   96 READ(5,1002) ALPHA,DUMMY1
       WRITE(6,1003) ALPHA,DUMMY1
       STORE(IST)=ALPHA
       DO 97 J=45,53
       JUMP2=J
       IF(ALPHA.EQ.PSEUDO(J)) JUMP2=J-44
       IF(JUMP2.NE.J) GO TO ( 95,99,100,101,102,103,104,105,106),JUMP2
   97 CONTINUE
       STORE(IST+1)=-1.
       IST=IST+2
       GO TO 2000
   99 CX=DUMMY1
       IF(.NOT. MKSFLG) GO TO 966
       CALL MTOFT(CX,1)
       GO TO 966
  100 CY=DUMMY1
       IF(.NOT. MKSFLG) GO TO 966
       CALL MTOFT(CY,1)
       GO TO 966
  101 X1=DUMMY1
       IF(.NOT. MKSFLG) GO TO 966
       CALL MTOFT(X1,1)
       GO TO 966
  102 Y1=DUMMY1
       IF(.NOT. MKSFLG) GO TO 966
       CALL MTOFT(Y1,1)
       GO TO 966
  103 NX=IFIX(DUMMY1)
       GO TO 966
  104 NY=IFIX(DUMMY1)
       GO TO 966
  105 CONTINUE
       GO TO 966
  106 CONTINUE
       GO TO 966
  966 CONTINUE
       STORE(IST+1)=DUMMY1
       STORE(IST+2)=-1.
       IST=IST+3
       GO TO 96
C      WE ARE NOW PROCESSING THE CONTOURS
```

```
  110 CONTINUE
      CONTFG=.TRUE.
      IF(J.EQ.54) IST=IST+2
C
  111 READ(5,1002) ALPHA,DUMMY1,DUMMY2
      WRITE(6,1003) ALPHA,DUMMY1,DUMMY2
      STORE(IST)=ALPHA
      STORE(IST+1)=-1.
      ASSIGN 111 TO ERROR
      GROUP=7
      DO 112 J=54,63
      JUMP2=J
      IF(ALPHA.EQ.PSEUDO(J)) JUMP2=J-53
      IF(JUMP2.NE.J) GO TO (110,113,114,115,116,117,118,119,120,121,122,
     1123,124),JUMP2
  112 CONTINUE
      IST=IST+2
      GO TO 2000
  113 VAL=DUMMY1
      GO TO 1111
  114 XDEL=DUMMY1
      IF(.NOT. MKSFLG) GO TO 1111
      CALL MTOFT(XDEL,1)
      GO TO 1111
  115 BASEX=DUMMY1
      IF(.NOT. MKSFLG) GO TO 1111
      CALL MTOFT(BASEX,1)
      GO TO 1111
  116 MX=IFIX(DUMMY1)
      GO TO 1111
  117 TOL=DUMMY1
      GO TO 1111
  118 APPROX=DUMMY1
      IF(.NOT. MKSFLG) GO TO 1111
      CALL MTOFT(APPROX,1)
      GO TO 1111
  119 SYMFLG=.TRUE.
      GO TO 1111
  120 XCOORD(1)=DUMMY1
      XCOORD(2)=DUMMY2
      LINEFG=.TRUE.
      GO TO 1111
  121 YCOORD(1)=DUMMY1
      YCOORD(2)=DUMMY2
      LINEFG=.TRUE.
      GO TO 1111
  122 REFLIN=1.
C     IMPLIES REFLECTION IS ABOUT X AXIS
      LINEFG=.TRUE.
      GO TO 1111
  123 REFLIN=2.
      LINEFG=.TRUE.
C     IMPLIES REFLECTION IS ABOUT Y AXIS
      GO TO 1111
  124 CONTINUE
      GO TO 1111
C
 1111 CONTINUE
      STORE(IST+1)=DUMMY1
```

```
      IF(JUMP2.LT.9) GO TO 11111
      STORE(IST+ 2)=DUMMY2
      STORE(IST+3)=-1.
       IST=IST+4
      GO TO 111
11111 STORE(IST+2)=-1.
      IST=IST+3
      GO TO 111
C     PROCESS THIS CASE
  130 CONTINUE
      INDEX=1
      IF(.NOT. MKSFLG) GO TO 6053
      IF(CONVRT) GO TO 6053
      CALL MTOFT(RADI,750)
      CALL MTOFT(STO,NUMRUN)
      CALL MTOFT(SL ,NUMRUN)
      CALL MTOFT(XLM1,30)
      CALL MTOFT(XLM2,30)
      CALL KGTOLB(W,200)
      CONVRT=.TRUE.
 6053 CONTINUE
      NRW=NUMRUN
      JUMP=1
      ISTART=1
      WRITE(6,6000)
 6000 FORMAT(1H1)
 4000 JOE=0
      IEND=ISTART+3
      DO 150 ILOOP=ISTART,IEND
      JOE=JOE+1
      IF(STORE(ILOOP).EQ.-1.) GO TO 4005
  150 CONTINUE
 4005 K=ISTART+JOE-2
      WRITE(6,1003)(STORE(ILOOP),ILOOP=ISTART,K)
      IF(STORE(ISTART) .EQ. HEADER) WRITE(6,5050) (TITLE(I),I=1,12)
      ISTART=K+2
      IF(ISTART.GE.IST) GO TO 155
 4010 FORMAT(A6,2F16.8)
      GO TO 4000
  131 CONTINUE
      ASSIGN 131 TO ERROR
      GROUP=8
C     THIS IS A REPEAT CASE WITH CERTAIN CHANGES
      DO 1311 J=1,NFLGHT
      IF(CLIMBA(J).NE.0.) GO TO 1311
      THETAI(2,J)=0.
 1311 CONTINUE
      READ(5,1002) ALPHA
      STORE(IST)=ALPHA
      STORE(IST+1)=-1.
      IST=IST+2
      IF(ALPHA.EQ.FLIGHT(1)) GO TO 132
      IF(ALPHA.EQ.RUNWAY(1)) GO TO 133
      IF(ALPHA.EQ.EXTNSN(1)) GO TO 134
      IF(ALPHA.EQ.GRID(1)) GO TO 95
      IF(ALPHA.EQ.CONTOR(1)) GO TO 110
  500 NEXTS=0
      NUMRUN=0
      NUMFLI=0
```

```
      NFLGHT=0
      DGL1FG=.FALSE.
      DGL2FG=.FALSE.
      DGL3FG=.FALSE.
      CONVRT=.FALSE.
      CANCEL=.FALSE.
      WECPFG=.FALSE.
      NEFFLG=.FALSE.
      SETPLT=.FALSE.
      LSIFLG=.FALSE.
      SSIFLG=.TRUE.
      MKSFLG=.TRUE.
      HESGFG=.FALSE.
      PTALFG=.TRUE.
      PTTBFG=.TRUE.
      PTINFG=.TRUE.
      DGLOFG=.TRUE.
      CALFLG=.FALSE.
      GRIDFG=.FALSE.
      PTOUFG=.TRUE.
      PTGDFG=.TRUE.
      PTCOFG=.TRUE.
      CONTFG=.FALSE.
      AREAFG=.FALSE.
      JUMP=1
      SYMFLG=.FALSE.
      LINEFG=.FALSE.
  600 CONTINUE
      DO 610 J=1,100
      DO 610 I=1,10
      RADI(I,J)=0.
      ZETI(I,J)=0.
      THETAI(I,J)=0.
      PTHRI(I,J)=0.
      NSG(J)=0
      CLIMBA(J)=0.
  610 NFT(I)=0
      DO 608 I=1,20
      DO 608 J=1,10
      TP(I,J)=0.
      PT(I,J)=0.
       W(I,J)=0.
      TY(I,J)=0.
      XLM1(1,J)=0.
      XLM1(2,J)=0.
      XLM1(3,J)=0.
      XLM2(1,J)=0.
      XLM2(2,J)=0.
      XLM2(3,J)=0.
      STO(J)=0.
      SL(J)=0.
      N(1,I,J)=0.
      N(2,I,J)=0.
  608 N(3,I,J)=0.
      GO TO 10
  132 CONTINUE
  133 CONTINUE
  134 CONTINUE
  135 CONTINUE
```

```
  136 CONTINUE
      JUMP=0
      INDEX=2
      RETURN
  140 CONTINUE
      STORE(IST)=ALPHA
      STORE(IST+1)=-1.
      IST=IST+2
      INDEX=3
      RETURN
  154 CONTINUE
  155 CONTINUE
      IF(.NOT. DGL3FG) GO TO 188
      WRITE(6,6000)
11000 FORMAT(4E16.8)
      WRITE(6,11000) ZETI(1,1),ZETI(2,1),ZETI(3,1),ZETI(4,1)
      WRITE(6,11000) ZETI(1,2),ZETI(2,2),ZETI(3,2),ZETI(4,2)
      WRITE(6,11000) ZETI(1,3),ZETI(2,3),ZETI(3,3),ZETI(4,3)
      WRITE(6,11000) THETAI(1,1),THETAI(1,2),THETAI(1,3),THETAI(1,4)
      WRITE(6,11000) THETAI(2,1),THETAI(2,2),THETAI(2,3),THETAI(2,4)
      WRITE(6,11000) THETAI(3,1),THETAI(3,2),THETAI(3,3),THETAI(3,4)
      WRITE(6,11000) THETAI(4,1),THETAI(4,2),THETAI(4,3),THETAI(4,4)
      WRITE(6,11000) PTHRI(1,1),PTHRI(1,2),PTHRI(1,3),PTHRI(1,4)
      WRITE(6,11000) PTHRI(2,1),PTHRI(2,2),PTHRI(2,3),PTHRI(2,4)
      WRITE(6,11000) PTHRI(3,1),PTHRI(3,2),PTHRI(3,3),PTHRI(3,4)
      WRITE(6,11000) PTHRI(4,1),PTHRI(4,2),PTHRI(4,3),PTHRI(4,4)
      WRITE(6,11000) RADI(1,1),RADI(2,1),RADI(3,1),RADI(4,1)
      WRITE(6,11000H RADI(1,2),RADI(2,2),RADI(3,2),RADI(4,2)
      WRITE(6,11000) RADI(1,3),RADI(2,3),RADI(3,3),RADI(4,3)
      WRITE(6,11000) CX,CY,X1,Y1
      WRITE(6,11000) VAL,XDEL,BASEX,APPROX
      WRITE(6,18000) MKSFLG,HESGFG,PTALFG,PTTBFG,PTINFG,DGL0FG,DGL1FG,
     2DGL2FG,DGL3FG,CALFLG,SSIFLG,LSIFLG,LINEFG,PTOUFG,PTGDFG,PTCOFG,
     3CONTFG,SYMFLG
18000 FORMAT(19L4)
      WRITE(6,12000) NUMRUN,NUMFLI,NEXTS,N(1,1,1),N(2,1,1),N(3,1,1)
12000 FORMAT(3I5,3F10.5)
      WRITE(6,11000) XLM1(1,1),XLM2(1,1),XLM1(2,1),XLM2(2,2),XLM1(3,1),
     2XLM1(4,1),XLM2(4,1),STO(1),SL(1),TP(1,1),TY(1,1)
      WRITE(6,6000)
  188 CONTINUE
      IF(IPHIL.NE.500) RETURN
      IF(.NOT.CANCEL) RETURN
      STOP
      END
$IBFTC CALPWM  DECK
      SUBROUTINE CALPLT
      COMMON/RYNME/RYNAM(10)
      COMMON/CAREA/AREA
      COMMON/FACTR/AXMAX,AXMIN,AYMAX,AYMIN,DELX,DELY,AINCHX,AINCHY,
     1  SETPLT
      COMMON/WM/XULHC,YULHC,XSF,XZIN,YHT
      COMMON/ARP/ARNV(20,25)
      COMMON/LOGFG2/DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,
     1   PTGDFG,PTCOFG,PTALFG
      COMMON/GDPAR/CX,CY,X1,Y1,NX,NY
      COMMON/JOE2/NEFFLG,AREAFG
      COMMON/JOE1/WECPFG,CANCEL
      COMMON/TROUBL/FLIGHT(22,20)
```

```
      DIMENSION PNOIS(3)
      LOGICAL DGL0FG,DGL1FG,DGL2FG,DGL3FG,PTINFG,PTTBFG,PTOUFG,PTGDFG,
     1   PTCOFG,PTALFG
      LOGICAL NEFFLG,AREAFG
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/EXTS/NEXT,JEXT(200),JRW(200),IFT(200),ALT(200),CANGLE(200),
     1  THRUST(200),CURV(200),MFL,MXX,MXSG
      COMMON/JOE/SYMFLG,REFLIN,XCOORD(2),YCOORD(2),TITLE(12)
      COMMON/ARPLOT/PRHARR(30,2,10),IMXARR(10),XNEV(10)
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/XDATA/ZETI(10,100),THETAI(10,100),PTHRI(10,100),
     1  RADI(10,100),NSG(100),NFLGHT
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),XLM1(3,10),
     X XLM2(3,10),STO(10),SL(10)
      DIMENSION SCA(4),XREF(30),YREF(30),INX(10),NEV(10)
      LOGICAL WECPFG,CANCEL
      LOGICAL SYMFLG
      LOGICAL SETPLT
      REAL N
      DATA PNOIS/6HNE VAL,6HWECPNL,6HN E F /
      READ(5,910) VAL1,VAL2,VAL3
  910 FORMAT(3F10.1)
C TITLE OF PLOT ON ITS SIDE
      CALL SYMBOL(0.,10.,.5,TITLE,-90.,21)
      CALL PLOT(6.,0.,-3)
      INCP=1
      IF(WECPFG) INCP=2
      IF(NEFFLG) INCP=3
      CONS1=.3048006096
      IF(.NOT. MKSFLG) GO TO 1030
      CX=CX*CONS1
      CY=CY*CONS1
      X1=X1*CONS1
      Y1=Y1*CONS1
 1030 CONTINUE
      IF(.NOT. MKSFLG) GO TO 1020
      DO 1010 J=1,NRW
      DO 1010 I=1,3
      XLM1(I,J)=XLM1(I,J)*CONS1
      XLM2(I,J)=XLM2(I,J)*CONS1
 1010 CONTINUE
 1020 CONTINUE
      IF(NFLGHT .GT. 10) GO TO 211
      DO 1000 J=1,20
      DO 1000 I=1,22
 1000 FLIGHT(I,J)=0.
      IIXT=0
      KI=0
      DO 500 J=1,NRW
      L=NFT(J)                                                    35200
      DO 500 I=1,L
      KI=KI+1
      FLIGHT(1,KI)=TP(I,J)
      FLIGHT(2,KI)=W(I,J)
      FLIGHT(3,KI)=PT(I,J)
      FLIGHT(4,KI)=TY(I,J)
      FLIGHT(5,KI)=N(1,I,J)
      NXT=NSG(KI)-2
```

```
      FLIGHT(6,KI)=NXT
      IF(NXT.EQ.0)GO TO 500
      INDEX=IIXT
      IIXT=IIXT+NXT
      DO 550 K=1,NXT
      IAD=(K-1)*4
      KX=INDEX+K                                                    35550
      FLIGHT(IAD+7,KI)=ALT(KX)
      FLIGHT(IAD+8,KI)=CANGLE(KX)*57.2958
      FLIGHT(IAD+9,KI)=THRUST(KX)
      FLIGHT(IAD+10,KI)=CURV(KX)
  550 CONTINUE
  500 CONTINUE
      IREF=0
      IF(SYMFLG)IREF=1
C
C        DRAW THE DATA FOR THE PLOT.
C
      NUM=0
      DO 210 I=1,NRW
  210 NUM=NUM+NFT(I)
      CALL HEAD(NUM)
  211 CONTINUE
      AINCH=AINCHX
      YINCH=AINCHY
      XD=DELX
      YD=DELY
      IF(SETPLT)GO TO 600
C
C
C        GET ADJUSTED SCALE VALUES FOR VERTICAL AXIS
C                SCA(3)=ADJUSTED MINIMUM
C                SCA(4)=INCREMENT ALONG AXIS
C
      XNX=NX-1
      XNY=NY-1
      T1=X1+XNX*CX
      T2=Y1+XNY*CY
      AYMIN=Y1
      AYMAX=T2
      IF(T2 .GT. Y1) GO TO 20
      AYMIN=T2
      AYMAX=Y1
   20 CONTINUE
      YULHC=AYMAX
      SCA(1)=AYMAX
      SCA(2)=AYMIN
      CALL SCALE(SCA,10.,2,1)
      YM=SCA(3)
      YD=SCA(4)
      AXMIN=X1
      AXMAX=T1
      IF(T1 .GT. X1) GO TO 10
      AXMIN=T1
      AXMAX=X1
   10 CONTINUE
      XULHC=AXMIN
      SCA(1)=AXMAX
      SCA(2)=AXMIN
```

```
      CALL SCALE(SCA,10.,2,1)
C        COMPUTE THE NUMBER OF INCHES FOR X-AXIS USING
C        THE Y-AXIS INCREMENT.IF LESS THAN 20,USE THE
C        Y-INCREMENT ON THE X-AXIS(AND Y-AXIS)
C
      SCMX=SCA(3)+10.*SCA(4)
      NINCH=(SCMX-SCA(3))/YD+.9
      IF(NINCH.LE.20)GO TO 40
C
C        IF THE NUMBER OF INCHES IS GREATER THAN 20,
C        SCALE THE X-AXIS TO 20 INCHES AND USE THIS
C        INCREMENT FOR THE Y-AXIS.
C
      CALL SCALE(SCA,20.,2,1)
      NINCH=20
      YD=SCA(4)
   40 IF(NINCH.LT.10)NINCH=10
      AINCH=NINCH
C
C        COMPUTE THE ZERO POINT ON THE Y-AXIS,IF ANY,
C        AND START THE X-AXIS THERE.
C
      YHT=-YM/YD
      IF(YHT.GT.10..OR.YHT.LT.0.)YHT=0.
C
C        DRAW THE AXES.
C                Y-AXIS = 10 INCHES LONG
C                X-AXIS = BETWEEN 10 AND 20 INCHES LONG
C
      IF(.NOT.MKSFLG)CALL AXIS(-.5,0.,35HLATERAL DISTANCE(THOUSANDS OF F
     XEET) ,35,10.,90.,YM/1000.,YD/1000.)
      IF(MKSFLG)CALL AXIS(-.5,0.,37HLATERAL DISTANCE(THOUSANDS OF METERS
     X) ,37,10.,90.,YM/1000.,YD/1000.)
      IF(.NOT.MKSFLG)CALL AXIS(0.,YHT,27HDISTANCE(THOUSANDS OF FEET) ,
     X -27,AINCH,0.,SCA(3)/1000.,YD/1000.)
      IF(MKSFLG)CALL AXIS(0.,YHT,29HDISTANCE(THOUSANDS OF METERS) ,
     X -29,AINCH,0.,SCA(3)/1000.,YD/1000.)
C
C
      XM=SCA(3)
      XD=YD
      XSF=XD
      XSF=1./XSF
  600 IF(.NOT.SETPLT) GO TO 650
      XNX=NX-1
      XNY=NY-1
      T1=X1+XNX*CX
      T2=Y1+XNY*CY
      XULHC=X1
      IF(T1 .LT. X1) XULHC=T1
      YULHC=Y1
      IF(T2 .GT. Y1) YULHC=T2
      XM=AXMIN
      IF(.NOT. MKSFLG)CALL AXIS2(-.5,0.,AYMAX,AYMIN,YD,YINCH,
     X 35HLATERAL DISTANCE(THOUSANDS OF FEET) ,35,-1,DELNY)
      IF(MKSFLG)CALL AXIS2(-.5,0.,AYMAX,AYMIN,YD,YINCH,
     X 37HLATERAL DISTANCE(THOUSANDS OF METERS) ,37,-1,DELNY)
      YHT=-AYMIN/DELNY
      IF(YHT.GT.YINCH .OR. YHT .LT. 0.) YHT=0.
```

```
      IF(.NOT.MKSFLG)CALL AXIS2(0.,YHT,AXMAX,AXMIN,XD,-AINCH,
     X 27HDISTANCE(THOUSANDS OF FEET) ,27,-1,DELNX)
      IF(MKSFLG)CALL AXIS2(0.,YHT,AXMAX,AXMIN,XD,-AINCH,
     X 29HDISTANCE(THOUSANDS OF METERS) ,29,-1,DELNX)
      YD=DELNY*1000.
      YM=AYMIN*1000.
      XM=AXMIN*1000.
      XD=DELNX*1000.
      XSF=1./XD
  650 CONTINUE
      DO 111 I=1,NRW
      X1=(XLM1(1,I)-XM)/XD
      Y1=(XLM1(2,I)-YM)/YD
      X2=(XLM2(1,I)-XM)/XD
      Y2=(XLM2(2,I)-YM)/YD
  111 CALL RUNWAY(X1,Y1,X2,Y2)
      CALL RWYLEG(AINCH)
      TEMP1=AINCH-4.5
      XUB=XM+AINCH*XD
      XZIN=-AINCH*XM/(XUB-XM)
      CALL NOMEN(XM,YM,XD,YD)
      WRITE(6,8010)
      WRITE(6,8020) XM,XUB,XSF,YM,YD,YHT,XZIN,AINCH,SCMX,T1,T2,XULHC,
     1  YULHC
      WRITE(6,8030) NINCH
 8000 CONTINUE
      IF(VAL1 .EQ. 0.) GO TO 900
      IF(MKSFLG) GO TO 8075
      CALL SYMBOL(11.,9.7,.08,15HAREA IN SQ. FT.,0.,15)
      GO TO 8076
 8075 CALL SYMBOL(11.,9.7,.08,15HAREA IN SQ. M. ,0.,15)
 8076 CONTINUE
      CALL SYMBOL(9.2,9.5,    .15,PNOIS(INCP),0.,6)
      CALL NUMBER(10.3,9.5,.15,VAL1,0.,-1)
      CALL AUTCNT(VAL1)
      CALL NUMBER(11.,9.5,.15,AREA,0.,-1)
      CALL SYMBOL(TEMP1,.25,.2,TITLE,0.,23)
      CALL SYMBOL(TEMP1,0.,.1,43HNUMBER OF FLIGHTS      NUMBER OF OPERAT
     1IONS,0.,43)
      TEMP2=TEMP1+1.85
      XNFLG=NFLGHT
      CALL NUMBER(TEMP2,0.,.1,XNFLG,0.,-1)
      TOTOP=0.
      DO 8200 I=1,3
      DO 8200 J=1,20
      DO 8200 K=1,10
 8200 TOTOP=TOTOP+N(I,J,K)
      TEMP2=TEMP1+4.45
      CALL NUMBER(TEMP2,0.,.1,TOTOP,0.,-1)
      IF(VAL2 .EQ. 0.) GO TO 900
      CALL SYMBOL(+36.,1.,.2,45HCALCOMP OPERATOR--PLEASE CHANGE PEN COLO
     1R NOW,45.,45)
      CALL SYMBOL(+36.,.7,.2,45H**********************************************
     1*****,45.,45)
      CALL SYMBOL(+36., 8.,1.,9HTHANK YOU,-45.,9)
      CALL SYMBOL(9.2,9.2,.15,PNOIS(INCP),0.,6)
      CALL NUMBER(10.3,9.2,.15,VAL2,0.,-1)
      CALL AUTCNT(VAL2)
      CALL NUMBER(11.,9.2,.15,AREA,0.,-1)
```

```
      IF(VAL3 .EQ. 0.) GO TO 900
      CALL SYMBOL(+38.,1.,.2,28HMAY I IMPOSE UPON YOU AGAIN ,45.,28)
      CALL SYMBOL(+38.,.7,.2,21H*****MANY THANKS*****,45.,21)
      CALL SYMBOL(9.2,8.9,.15,PNOIS(INCP),0.,6)
      CALL NUMBER(10.3,8.9,.15,VAL3,0.,-1)
      CALL AUTCNT,VAL3)
      CALL NUMBER(11.,8.9,.15,AREA,0.,-1)
  900 CONTINUE
C         END THE PLOT.
C
      CALL PLOT(ABS(AINCH)+12.,0.,-3)
      CALL SYMBOL(0.,0.,.5,TITLE,90.,21)
      CALL PLOT(6.,0.,-3)
      RETURN
 8010 FORMAT(12H1FROM CALPLT)
 8020 FORMAT(55H XM,XUB,XSF,YM,YD,YHT,XZIN,AINCH,SCMX,T1,T2,XULHC,YULHC/
     1(8E15.5))
 8030 FORMAT(6H NINCH/I10)
      END
$IBFTC RWLGX    DECK
      SUBROUTINE RWYLEG(AINCH)
C  THIS SUBROUTINE PLOTS THE RUNWAY LEGEND
      COMMON/AA/TP(20,10),PT(20,10),W(20,10),TY(20,10),
     1  XLM1(3,10),XLM2(3,10),STO(10),SL(10)
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      COMMON/RYNME/RYNAM(10)
      CALL PLOT(AINCH+4.,2.,-3)
      CALL PLOT(2.,2.,3)
      CALL PLOT(-2.,2.,2)
      CALL PLOT(-2.,-2.,2)
      CALL PLOT(2.,-2.,2)
      CALL PLOT(2.,2.,2)
      CALL SYMBOL(-1.63,1.70,.25,13HRUNWAY LEGEND,0.,13)
      DO 1111 I=1,NRW
      SX1=XLM1(1,I)/5000.
      SY1=XLM1(2,I)/5000.
      SX2=XLM2(1,I)/5000.
      SY2=XLM2(2,I)/5000.
      CALL RUNWAY(SX1,SY1,SX2,SY2)
      SLOPE=(SY2-SY1)/(SX2-SX1)
      ANG=ATAN(SLOPE)*180./3.14159
      IF(SLOPE .GT. 0. .AND. SY2 .GT. SY1) GO TO 1000
      CALL SYMBOL(SX1+.125,SY1+.125,.063,RYNAM(I),ANG,3)
      GO TO 1111
 1000 CALL SYMBOL(SX1-.2,SY1-.2,.063,RYNAM(I),ANG,3)
 1111 CONTINUE
      CALL PLOT(-AINCH-4.,-2.,-3)
      RETURN
      END
$IBFTC INIP     DECK
      SUBROUTINE INIPLT
      COMMON/FACTR/AXMAX,AXMIN,AYMAX,AYMIN,DELX,DELY,AINCHX,AINCHY,
     1  SETPLT
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      LOGICAL SETPLT
      DIMENSION DATA(1024)
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      CALL PLOTS(DATA,1024)
      IF(SETPLT) GO TO 10
```

```
      SIZE=11.
      IF(LSIFLG)SIZE=30.
      CALL FACTOR(SIZE/11.)
      GO TO 20
   10 SIZE=10.
      IF(LSIFLG) SIZE=30.
      CALL FACTOR(SIZE/10.)
   20 CONTINUE
      CALL PLOT(0.,-11.,-3)
      CALL PLOT(0.,.5,-3)
      RETURN
      ENTRY FINPLT
      CALL PLOT(0.,0.,999)
      RETURN
      END
$IBFTC NOMX    DECK
      SUBROUTINE NOMEN(XM,YM,XD,YD)
      RETURN
      END
$IBFTC REF     DECK
      SUBROUTINE REFLEC(X1,Y1,X2,Y2,X0,Y0,XN,YN)
      RETURN
      END
$IBFTC RUN     DECK
      SUBROUTINE RUNWAY(X1,Y1,X2,Y2)
      COMMON/IANG/SINAN,COSAN,XW,YW
      XW=0.
      YW=0.
      DELX=.075
      DELY=.05
      CALL PLOT(X1,Y1,-3)
      XX2=SQRT((Y2-Y1)**2+(X2-X1)**2)
      SINAN=(Y2-Y1)/XX2
      COSAN=(X2-X1)/XX2
      CALL PLOTA(0.,DELY,3)
      CALL PLOTA(0.,-DELY,2)
      CALL PLOTA(XX2,DELY,3)
      CALL PLOTA(XX2,-DELY,2)
      YY=DELX
      DO 10 I=1,5
      YY=YY-DELX/3.
      CALL PLOTA(0.,YY,3)
   10 CALL PLOTA(XX2,YY,2)
      CALL PLOT(-X1,-Y1,-3)
      RETURN
      END
$IBFTC DLIN    DECK
      SUBROUTINE DLINE(X,Y,K,XM,DELX,YM,DELY,NC,HT)
      COMMON/IANG/STH,CTH,POINTX,POINTY
      DIMENSION X(1),Y(1)
      IF(NC.EQ.1)GO TO 14
      IGO=NC-1
      H23=2.*HT/3.
      H13=HT/3.
      H12=HT/2.
      H56=5.*HT/6.
      H16=HT/6.
      I=2
      POINTX=(X(1)-XM)/DELX
```

```
      POINTY=(Y(1)-YM)/DELY
    1 DIFFX=(X(I)-XM)/DELX-POINTX
      DIFFY=(Y(I)-YM)/DELY-POINTY
      DIS=SQRT(DIFFX*DIFFX+DIFFY*DIFFY)
      IF(DIS.GT.HT)GO TO 2
      I=I+1
      IF(I.GT.K)RETURN
      GO TO 1
    2 THETA=ATAN2(DIFFY,DIFFX)
      CTH=COS(THETA)
      STH=SIN(THETA)
      CALL PLOT(POINTX,POINTY,3)
      GO TO (10,11,12,13),IGO
   10 CALL PLOTA(POINTX+H23,POINTY,2)
      GO TO 15
   11 CALL PLOTA(POINTX+H13,POINTY,2)
      CALL PLOTA(POINTX+H12,POINTY,3)
      CALL PLOTA(POINTX+H56,POINTY,2)
      GO TO 15
   12 CALL PLOTA(POINTX+H12,POINTY,2)
      CALL PLOTA(POINTX+H23,POINTY,3)
      CALL PLOTA(POINTX+H56,POINTY,2)
      GO TO 15
   13 CALL PLOTA(POINTX+H16,POINTY,2)
      CALL PLOTA(POINTX+H13,POINTY,3)
      CALL PLOTA(POINTX+H12,POINTY,2)
      CALL PLOTA(POINTX+H23,POINTY,3)
      CALL PLOTA(POINTX+H56,POINTY,2)
      GO TO 15
   14 CALL ALINE(X,Y,K,XM,DELX,YM,DELY)
      RETURN
   15 POINTX=POINTX+CTH*HT
      POINTY=POINTY+STH*HT
      GO TO 1
      END
$IBFTC PTA      DECK
      SUBROUTINE PLOTA(X,Y,I)
      COMMON/IANG/SINAN,COSAN,XW,YW
      XX=X-XW
      YY=Y-YW
      XP=XX*COSAN+YY*SINAN+XW
      YP=XX*SINAN-YY*COSAN+YW
      CALL PLOT(XP,YP,I)
      RETURN
      END
$IBFTC LNE      DECK
      SUBROUTINE ALINE(X,Y,N,XM,DELX,YM,DELY)
      DIMENSION X(1),Y(1)
      CALL PLOT((X(1)-XM)/DELX,(Y(1)-YM)/DELY,3)
      DO 10 I=2,N
   10 CALL PLOT((X(I)-XM)/DELX,(Y(I)-YM)/DELY,2)
      RETURN
      END
$IBFTC AXE2     DECK
      SUBROUTINE AXIS2(X0,Y0,AMAX,AMIN,DELX,AINCH,BCD,NCH,NDEC,DELN)
      DIMENSION BCD(1)
      HT=.12
      CINCH=ABS(AINCH)
      IF(AINCH.LT.0.)GO TO 5
```

```
      W2=1.
      W1=0.
      GO TO 10
    5 W1=1.
      W2=0.
   10 CALL PLOT(X0,Y0,3)
      NUM=(AMAX-AMIN)/DELX+1.
      ANC=CINCH/FLOAT(NUM-1)
      DELN=DELX/ANC
      ANUM=AMIN-DELX
      X=0.
      Y=0.
      XM=0.
      DO 40 I=1,NUM
      ANUM=ANUM+DELX
      II=0
   25 IF(ABS(ANUM)/10.**II.LT.1.)GO TO 20
      II=II+1
      GO TO 25
   20 IF(ANUM.LT.0.)II=II+1
      IMORE=NDEC+1
      II=II+IMORE
      CENTER=FLOAT(II)*HT
      IF(W2.EQ.0.)CENTER=CENTER/2.
      OFF=HT/2.
      XC=X-CENTER-.15*W2
      IF(XC.LT.XM)XM=XC
      YC=Y-(HT+.15)*W1-OFF*W2
      CALL PLOT(X0+X,Y0+Y,2)
      CALL PLOT(X0+X+.1*W2,Y0+Y+.1*W1,3)
      CALL PLOT(X0+X-.1*W2,Y0+Y-.1*W1,2)
      CALL NUMBER(X0+XC,Y0+YC,HT,ANUM,0.,NDEC)
      CALL PLOT(X0+X,Y0+Y,3)
      X=X+ANC*W1
      Y=Y+ANC*W2
   40 CONTINUE
      BST=(CINCH-FLOAT(NCH)*HT)/2.
      XXC=(X0+BST)*W1+(X0+XM-OFF)*W2
      YYC=(Y0+YC-HT-OFF)*W1+(Y0+BST)*W2
      CALL SYMBOL(XXC,YYC,.12,BCD,90.*W2,NCH)
      RETURN
      END
$IBFTC HD       DECK
      SUBROUTINE HEAD(NUM)
      COMMON/JOE2/NEFFLG,AREAFG
      COMMON/TROUBL/FLIGHT(22,20)
      COMMON/JOE/SYMFLG,REFLIN,XCOORD(2),YCOORD(2),TITLE(12)
      COMMON/JOE1/WECPFG,CANCEL
      COMMON/LOGFG1/MKSFLG,CALFLG,SSIFLG,LSIFLG
      COMMON/NN/NRW,N(3,20,10),NFT(10)
      DIMENSION FTKG(2),LBME(2),FTMD(2)
      DIMENSION PLANE(3,13)
      LOGICAL NEFFLG,AREAFG
      LOGICAL SYMFLG
      LOGICAL LL
      LOGICAL CANCEL,WECPFG
      LOGICAL MKSFLG,CALFLG,SSIFLG,LSIFLG
      REAL LBME
      REAL N
```

```
      DATA FTKG,LBME,FTMD/4H(FT),4H(M) ,4H(LB),4H(KG),6H FT/DG,6H M/DG /
      DATA PLANE(1,1)/18H4 ENG. HBPR TF       /,
     X     PLANE(1,2)/18H3 ENG. HBPR TF       /,
     X     PLANE(1,3)/18H4 ENG. LBPR TF       /,
     X     PLANE(1,4)/18H4 ENG. TURBOJET      /,
     X     PLANE(1,5)/18H3 ENG. LBPR TF       /
      DATA PLANE(1,6)/18H2 ENG. LBPR TF       /,
     X     PLANE(1,7)/18H2 ENG. LBPR FANS     /,
     X     PLANE(1,8)/18H4 ENG. EX. JETS      /,
     X     PLANE(1,9)/18H2 ENG. LBPR FANS     /,
     X     PLANE(1,10)/18H2 ENG. EX. JETS      /,
     X     PLANE(1,11)/18H2 ENG. PROP.         /,
     X     PLANE(1,12)/18H4 ENG. PROP.         /,
     X     PLANE(1,13)/18HSST                 /
      FLEN=2.5
      X=0.
      AMAX=FLIGHT(6,1)
      IF(NUM.EQ.1) GO TO 5
      DO 6 I=2,NUM
      IF(FLIGHT(6,I).GT.AMAX) AMAX=FLIGHT(6,I)
    6 CONTINUE
    5 XCTR=(3.5+FLOAT(NUM)*FLEN)/2.
      AMX=AMAX
      IF(AMAX .GE. 2.) AMX=2.
      YTOP=(AMX*2.+3.2+2.1)
      YTOP=YTOP+(10.-YTOP)/2.
      INDEX=1
      IF(.NOT.MKSFLG)GO TO 41
      INDEX=2
      DO 40 I=1,NUM
      CALL LBTOKG(FLIGHT(2,I),1)
      IND=IFIX(FLIGHT(6,I))
      IF(IND.EQ.0)GO TO 40
      DO 400 K=1,IND
      JI=4*(K-1)
      CALL FTTOM(FLIGHT(JI+10,I),1)
  400 CONTINUE
   40 CONTINUE
   41 DO 42 I=1,NUM
      IND=FLIGHT(6,I)
      IF(IND.EQ.0)GO TO 42
      DO 420 K=1,IND
      JI=4*(K-1)
      LL=FLIGHT(JI+10,I).EQ.0.
      IF(LL.AND.MKSFLG)CALL FTTOM(FLIGHT(JI+7,I),1)
      IF(.NOT.LL)FLIGHT(JI+7,I)=FLIGHT(JI+7,I)*57.2958
  420 CONTINUE
   42 CONTINUE
      IF(WECPFG)CALL SYMBOL(XCTR-1.3,YTOP-.2,.2,13HWECPNL VALUES ,0.,13)
      IF(.NOT.WECPFG)CALL SYMBOL(XCTR-2.1,YTOP-.2,.2,21HNOISE EXPOSURE V
     XALUES,0.,21)
      CALL SYMBOL(XCTR-2.3,YTOP-.6,.2,TITLE,0.,23)
      CALL SYMBOL(XCTR-3.2,YTOP-1.1,.2,
     1 32HRUNWAYS AND GRID POINTS AS SHOWN,0.,32)
      CALL SYMBOL(XCTR-1.7,YTOP-2.3,.2,17HFLIGHT STATISTICS,0.,17)
      YTOP=YTOP-2.3
      DELY=.2
      SAVE=YTOP-2.*DELY
      YTOP=YTOP-2.*DELY
```

```
      CALL SYMBOL(X,YTOP-4.*DELY,.15,15HTAKEOFF/LANDING,0.,15)
      CALL SYMBOL(X,YTOP-6.*DELY,.15,15HAIRCRAFT WEIGHT    ,0.,15)
      CALL SYMBOL(X+2.25,YTOP-6.*DELY,.15,LBME(INDEX),0.,4)
      CALL SYMBOL(X,YTOP-8.*DELY,.15,14HPERCENT THRUST,0.,14)
      CALL SYMBOL(X,YTOP-10.*DELY,.15,13HAIRCRAFT TYPE,0.,13)
      CALL SYMBOL(X,YTOP-12.*DELY,.15,17HNO. OF OPERATIONS,0.,17)
      CALL SYMBOL(X,YTOP-14.*DELY,.15,17HNO. OF EXTENSIONS,0.,17)
      IF(AMAX.LT.1.) GO TO 50
      MAX=AMAX
      DO 15 K=1,MAX
      IF(K.EQ.2.OR.K.EQ.4)YTOP=YTOP-2.
      CALL SYMBOL(X+.3,YTOP-16.*DELY,.15,9HEXTENSION,0.,9)
      IF(K.EQ.3)YTOP=YTOP+2.
      IF(K.GT.2)GO TO 15
      CALL SYMBOL(X,YTOP-18.*DELY,.15,16HPROJECTED LENGTH    ,0.,16)
      CALL SYMBOL(X+2.4,YTOP-18.*DELY,.15,FTMD(INDEX),0.,6)
      CALL SYMBOL(X,YTOP-20.*DELY,.15,20HELEVATION ANGLE(DEG),0.,20)
      CALL SYMBOL(X,YTOP-22.*DELY,.15,14HPERCENT THRUST,0.,14)
      CALL SYMBOL(X,YTOP-24.*DELY,.15, 6HRADIUS    ,0., 6)
      CALL SYMBOL(X+1.,YTOP-24.*DELY,.15,FTKG(INDEX),0.,4)
   15 CONTINUE
   50 X=X+3.5-FLEN
      JJ=1
      KK=0
      DO 800 I=1,NUM
      Y=SAVE-4.*DELY
      X=X+FLEN
      IF(KK .LT. NFT(JJ)) GO TO 55
      JJ=JJ+1
      KK=0
   55 CONTINUE
      KK=KK+1
      IF(KK.EQ.1)CALL SYMBOL( X,Y+4.*DELY,.15,6HRUNWAY,0.,6)
      IF(KK.EQ.1)CALL NUMBER(X+1.1,Y+4.*DELY,.15,FLOAT(JJ),0.,-1)
      CALL SYMBOL(X,Y+2.*DELY,.15,6HFLIGHT,0.,6)
      CALL NUMBER(X+1.1,Y+2.*DELY,.15,FLOAT(KK),0.,-1)
      IF(FLIGHT(1,I).EQ.1.) GO TO 60
      CALL SYMBOL(X,Y,.15,7HTAKEOFF,0.,7)
      GO TO 65
   60 CALL SYMBOL(X,Y,.15,7HLANDING,0.,7)
   65 K=6+INT(FLIGHT(6,I))*4
      XTEMP=X
      DO 80 J=2,K
      NK=-1
      IF(J .EQ. 16 .OR. J .EQ. 20) NK=1
      IF(J.EQ.8.OR.J.EQ.12)NK=1
      Y=Y-2.*DELY
      IF(J.EQ.5.AND.WECPFG)GO TO 67
      IF(J .EQ. 5 .AND. NEFFLG) GO TO 670
      IF(J.EQ.4)GO TO 70
      IF(J.EQ.7.OR.J.EQ.11) Y=Y-2.*DELY
      IF(J.EQ.5)NK=3
      IF(J.EQ.7)SAY7=Y
      IF(J.EQ.11)SAY11=Y
      IF(J.EQ.15)Y=SAY7
      IF(J.EQ.15)X=X+1.
      IF(J.EQ.19)Y=SAY11
      KKK=(J-6)/4+1
      IF(J.EQ.7.OR.J.EQ.11.OR.J.EQ.15.OR.J.EQ.19)
```

```
      X  CALL NUMBER(X+.4,Y+2.*DELY,.15,FLOAT(KKK),0.,-1)
         CALL NUMBER(X,Y,.15,FLIGHT(J,I),0.,NK)
         GO TO 80
C WECPNL OPERATIONS
   67 CALL SYMBOL(X,Y+.13,.1,3HDAY,0.,3)
         CALL NUMBER(X+.4,Y+.13,.1,N(1,KK,JJ),0.,3)
         CALL SYMBOL(X,Y,.1,3HEVN,0.,3)
         CALL NUMBER(X+.4,Y,.1,N(2,KK,JJ),0.,3)
         CALL SYMBOL(X,Y-.13,.1,3HNGT,0.,3)
         CALL NUMBER(X+.4,Y-.13,.1,N(3,KK,JJ),0.,3)
         GO TO 80
C NEF OPERATIONS
  670 CALL SYMBOL(X,Y+.07,.12,3HDAY,0.,3)
         CALL NUMBER(X+.5,Y+.07,.12,N(1,KK,JJ),0.,3)
         CALL SYMBOL(X,Y-.13,.12,3HNGT,0.,3)
         CALL NUMBER(X+.5,Y-.13,.12,N(3,KK,JJ),0.,3)
         GO TO 80
   70 IND=FLIGHT(J,I)
         CALL SYMBOL(X,Y,.15,PLANE(1,IND),0.,18)
   80 CONTINUE
         X=XTEMP
  800 CONTINUE
         CALL PLOT(XCTR*2.+2.,0.,-3)
         RETURN
         END
$IBFTC ACXX     DECK
         SUBROUTINE AUTCNT(VAL)
         DIMENSION V1(3),V2(3),V3(3),P1(3),P2(3)
         DIMENSION C(4,3),COEF(8,4)
         COMMON/ARP/ZZ(20,25)
         COMMON/GDPAR/CX,CY,XX1,YY1,NS,MS
         COMMON/WM/X1S,Y1S,SF,XO,YO
         COMMON/WARP/Z(27,22)
         COMMON/CAREA/ARFA
         AREA=0.
         DX=ABS(CX)
         DY=ABS(CY)
         M=MS+2
         N=NS+2
         X1=X1S-DX
         Y1=Y1S+DY
         DO 3 I=1,NS
         DO 3 J=1,MS
         ZTEMP=ZZ(I,J)
         IP1=I+1
         JP1=J+1
3        Z(JP1,IP1)=ZTEMP
         DO 6 J=3,N
         L=J-1
         ZTEMP=2.*Z(2,L)-Z(3,L)
         Z(1,L)=ZTEMP
         ZTEMP=2.*Z(M-1,L)-Z(M-2,L)
6        Z(M,L)=ZTEMP
         DO 10 I=3,M
         L=I-1
         ZTEMP=2.*Z(L,2)-Z(L,3)
         Z(L,1)=ZTEMP
         ZTEMP=2.*Z(L,N-1)-Z(L,N-2)
10       Z(L,N)=ZTEMP
```

```
      Z(1,1)=(2.*(Z(2,1)+Z(1,2)+Z(2,2))
     X        -(Z(3,1)+Z(1,3)+Z(3,3)))/3.0
      ZTEMP=(2.*(Z(1,N-1)+Z(2,N)+Z(2,N-1))
     X        -(Z(1,N-2)+Z(3,N)+Z(3,N-2)))/3.0
      Z(1,N)=ZTEMP
      ZTEMP=(2.*(Z(M,2)+Z(M-1,2)+Z(M-1,1))
     X        -(Z(M,3)+Z(M-2,3)+Z(M-2,1)))/3.0
      Z(M,1)=ZTEMP
      ZTEMP=(2.*(Z(M,N-1)+Z(M-1,N-1)+Z(M-1,N))
     X        -(Z(M,N-2)+Z(M-2,N-2)+Z(M-2,N)))/3.0
      Z(M,N)=ZTEMP
      ACC=.1
      NFINE=ABS(DX*SF/ACC)+1.0
      MFINE=ABS(DY*SF/ACC)+1.0
      DO 9 I=2,M
      DO 9 JQ=2,N
      J=JQ
      J14=MOD(I,2)
      IF(J14.NE.0) J=N+2-JQ
      IF(NFINE.LE.2.OR.MFINE.LE.2) GO TO 30
      IF((I.GT.2.AND.I.LT.M).AND.(J.GT.2.AND.J.LT.N)) GO TO 8
      GO TO 9
30    DO 4 L=1,4
      JM=J+MOD(L/2,2)-1
      IM=I+MOD((L+3)/2,2)-1
      C(L,1)=X1+DX*FLOAT(JM-1)
      C(L,2)=Y1-DY*FLOAT(IM-1)
4     C(L,3)=Z(IM,JM)
      CALL CRECT(C,VAL)
      GO TO 9
8     CALL CUBICS(I,J,COEF)
      DO 7 II=2,MFINE
      DO 7 JK=2,NFINE
      JJ=JK
      J24=MOD(II,2)
      IF(J24.NE.0) JJ=NFINE+2-JK
      DO 5 L=1,4
      JM=JJ+MOD(L/2,2)-1
      IM=II+MOD((L+3)/2,2)-1
      C(L,1)=FLOAT(JM-1)/FLOAT(NFINE-1)-1.
      C(L,2)=FLOAT(IM-1)/FLOAT(MFINE-1)-1.
      CALL INTPT(COEF,C(L,1),C(L,2),C(L,3))
      TCC=C(L,1)*DX+X1-DX*FLOAT(J-1)
      C(L,1)=TCC
      TCC=-C(L,2)*DY+Y1-DY*FLOAT(I-1)
5     C(L,2)=TCC
      CALL CRECT(C,VAL)
7     CONTINUE
9     CONTINUE
      DO 24 I=1,MS
24    WRITE(6,25)(ZZ(J,I),J=1,NS)
25    FORMAT(1X,10F7.1/1X,10F7.1)
      WRITE(6,26)CX,CY,XX1,YY1,NS,MS
26    FORMAT(1H1,2E17.5///1X,2E17.5////1X,2I10)
      WRITE(6,27)X1S,Y1S,SF,XO,YO
27    FORMAT(////1X,5E17.5)
      DO 28 I=1,M
28    WRITE(6,25)(Z(I,J),J=1,N)
      RETURN
```

```
      END
$IBFTC CRECX   DECK
      SUBROUTINE CRECT(C,VAL)
      DIMENSION C(4,3),V1(3),V2(3),V3(3),P1(3),P2(3)
      COMMON/ARP/ZZ(20,25)
      COMMON/GDPAR/CX,CY,XX1,YY1,N,M
      COMMON/WM/X1,Y1,SF,XO,YO
      COMMON/WARP/Z(27,22)
      DX=ABS(CX)
      DY=ABS(CY)
      V2(1)=(C(1,1)+C(2,1))/2.0
      V2(2)=(C(2,2)+C(3,2))/2.0
      V2(3)=(C(1,3)+C(2,3)+C(3,3)+C(4,3))/4.0
      DO 81 K=5,8
      IL=MOD((K-1),4)+1
      IT=MOD((K-2),4)+1
      V1(1)=C(IL,1)
      V1(2)=C(IL,2)
      V1(3)=C(IL,3)
      V3(1)=C(IT,1)
      V3(2)=C(IT,2)
      V3(3)=C(IT,3)
      CALL STRIKE(V1,V2,V3,VAL,P1,P2,IER)
      IF(IER.NE.2. GO TO 81
      U1=P1(1)*SF+XO
      V11=P1(2)*SF+YO
      U2=P2(1)*SF+XO
      V22=P2(2)*SF+YO
      CALL PLOT(U1,V11,3)
      CALL PLOT(U2,V22,2)
81    CONTINUE
      RETURN
      END
$IBFTC CUBX    DECK
      SUBROUTINE CUBICS(I,J,COEF)
      DIMENSION COEF(8,4)
      COMMON/WARP/Z(27,22)
      DO 3 IEQ=1,4
      IB=I+IEQ-3
      COEF(IEQ,1)=Z(IB,J)
      CF=(Z(IB,J-2)-6.*Z(IB,J-1)+3.*Z(IB,J)
     X    +2.*Z(IB,J+1))/6.
      COEF(IEQ,2)=CF
      CF=(Z(IB,J-1)+Z(IB,J+1))/2.0-Z(IB,J)
      COEF(IEQ,3)=CF
      CF=(-Z(IB,J-2)+3.*Z(IB,J-1)-3.*Z(IB,J)
     X   +Z(IB,J+1))/6.0
3     COEF(IEQ,4)=CF
      DO 4 IEQ=5,8
      JB=J+IEQ-7
      COEF(IEQ,1)=Z(I,JB)
      CF=(Z(I-2,JB)-6.*Z(I-1,JB)+3.*Z(I,JB)
     X   +2.*Z(I+1,JB))/6.0
      COEF(IEQ,2)=CF
      CF=(Z(I-1,JB)+Z(I+1,JB))/2.0-Z(I,JB)
      COEF(IEQ,3)=CF
      CF=(-Z(I-2,JB)+3.*Z(I-1,JB)-3.*Z(I,JB)
     X   +Z(I+1,JB))/6.0
4     COEF(IEQ,4)=CF
```

```
      RETURN
      END
$IBFTC ITPTX   DECK
      SUBROUTINE INTPT(COEF,XREL,YREL,HT)
      DIMENSION COEF(8,4),ZH(4)
      DO 30 IEQ=1,4
      IF(XREL .NE. 0.) GO TO 2
      ZH(IEQ)=COEF(IEQ,1)
      GO TO 30
    2 ZH(IEQ)=0.
      DO 3 J=1,4
      ZTEMP=ZH(IEQ)+COEF(IEQ,J)*XREL**(J-1)
3     ZH(IEQ)=ZTEMP
   30 CONTINUE
      ZY=ZH(3)
     X      +YREL*(ZH(1)-6.*ZH(2)+3.*ZH(3)+2.*ZH(4))/6.
     X      +YREL**2*((ZH(2)+ZH(4))/2.-ZH(3))
     X      +YREL**3*(-ZH(1)+3.*ZH(2)-3.*ZH(3)+ZH(4))/6.
      DO 50 IEQ=5,8
      I=IEQ-4
      IF(YREL .NE. 0.) GO TO 4
      ZH(I)=COEF(IEQ,1)
      GO TO 50
    4 ZH(I)=0.
      DO 5 J=1,4
      ZTEMP=ZH(I)+COEF(IEQ,J)*YREL**(J-1)
5     ZH(I)=ZTEMP
   50 CONTINUE
      ZX=ZH(3)
     X      +XREL*(ZH(1)-6.*ZH(2)+3.*ZH(3)+2.*ZH(4))/6.
     X      +XREL**2*((ZH(2)+ZH(4))/2.-ZH(3))
     X      +XREL**3*(-ZH(1)+3.*ZH(2)-3.*ZH(3)+ZH(4))/6.
      HT=(ZX+ZY)/2.
      RETURN
      END
$IBFTC STRKX   DECK
      SUBROUTINE STRIKE(V1,V2,V3,H,P1,P2,IER)
      COMMON/CAREA/AREA
      DIMENSION V1(1),V2(1),V3(1),P1(1),P2(1),Q(3,3)
      ARINC=ABS((V3(1)-V1(1))*(V2(2)-V1(2))-(V3(2)-V1(2))*(V2(1)-V1(1)))
     1/2.
      DO 6 I=1,3
      P1(I)=0.
      P2(I)=0.
      T1=V1(I)
      Q(I,1)=T1
      T1=V2(I)
      Q(I,2)=T1
      T1=V3(I)
6     Q(I,3)=T1
      DO 7 I=1,3
      DO 1 J=I,3
      IF(Q(3,I)-Q(3,J))1,1,2
    2 DO 8 L=1,3
      A=Q(L,I)
      T1=Q(L,J)
      Q(L,I)=T1
      Q(L,J)=A
    8 CONTINUE
```

```
    1 CONTINUE
    7 CONTINUE
      IER=0
      IF(Q(3,3).LT.H.OR.Q(3,1).GT.H)GO TO 10
      IF(Q(3,1).EQ.H.AND.Q(3,2).EQ.H.AND.Q(3,3).EQ.H) GO TO 20
      A=(H-Q(3,1))/(Q(3,1)-Q(3,3))
      P1(1)=(Q(1,1)-Q(1,3))*A+Q(1,1)
      P1(2)=(Q(2,1)-Q(2,3))*A+Q(2,1)
      P1(3)=H
      IF(Q(3,1).NE.H) GO TO 3
      IF(Q(3,2).NE.H)GO TO 4
      P2(1)=Q(1,2)
      P2(2)=Q(2,2)
      P2(3)=Q(3,2)
      IER=2
      GO TO 20
    3 IF(Q(3,2).GT.H)GO TO 5
      A=(H-Q(3,2))/(Q(3,2)-Q(3,3))
      P2(1)=(Q(1,2)-Q(1,3))*A+Q(1,2)
      P2(2)=(Q(2,2)-Q(2,3))*A+Q(2,2)
      P2(3)=H
      IER=2
      GO TO 20
    4 IER=1
      P2(1)=P1(1)
      P2(2)=P1(2)
      P2(3)=H
      RETURN
    5 A=(H-Q(3,1))/(Q(3,1)-Q(3,2))
      P2(1)=(Q(1,1)-Q(1,2))*A+Q(1,1)
      P2(2)=(Q(2,1)-Q(2,2))*A+Q(2,1)
      P2(3)=H
      IER=2
 20   AREA=AREA+ARINC/2.
      RETURN
 10   IF(Q(3,1).GT.H)AREA=AREA+ARINC
      RETURN
      END
$STOP
'     END OF FILE CARD
```