NASA CONTRACTOR REPORT

NASA CR-1912

DEVELOPMENT AND APPLICATION OF A METHOD FOR PREDICTING ROTOR FREE WAKE POSITIONS AND RESULTING ROTOR BLADE AIR LOADS

Volume II - Program Listings

by S. Gene Sadler

Prepared by
ROCHESTER APPLIED SCIENCE ASSOCIATES, INC.
Rochester, N. Y. 14618
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • DECEMBER 1971

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-1912		

4. Title and Subtitle	5. Report Date
DEVELOPMENT AND APPLICATION OF A METHOD FOR PREDICTING ROTOR FREE WAKE POSITIONS AND RESULTING ROTOR BLADE AIR LOADS – VOLUME II - PROGRAM LISTINGS	December 1971
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
S. Gene Sadler	69-11
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
Rochester Applied Science Associates, Inc. 140 Allens Creek Road Rochester, New York 14618	NAS1-8448
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

VOLUME I - MODEL AND RESULTS is contained in NASA CR-1911

16. Abstract

Computer program listings are presented for two separate programs - the wake geometry and blade loads and response programs. These listings correspond to the calculations discussed in Volume I.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Rotor wake prediction Blade response prediction Free wake program Rotor wake flow	Unclassified - Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	149	$3.00

*For sale by the National Technical Information Service, Springfield, Virginia 22151

CONTENTS

CONTENTS

DEVELOPMENT AND APPLICATION OF A METHOD FOR PREDICTING ROTOR FREE WAKE POSITIONS AND RESULTING ROTOR BLADE AIR LOADS

VOLUME II - PROGRAM LISTINGS*

By S. Gene Sadler
Rochester Applied Science Associates, Inc.

SUMMARY

Computer program listings are presented for two separate programs, the wake geometry and the blade loads and response programs. These listings correspond to the calculations discussed in Volume I.

INTRODUCTION

Four steps are necessary in obtaining blade loads and response results including the effects of free wake distortions by using the programs discussed in this report. (1) Preliminary calculations (or measured data) are used to define rotor system trim parameters and flight conditions; and definitions of model parameter and program control variables are necessary before program operation may begin. (2) A wake geometry calculation is then made to obtain wake-induced velocity influence coefficients for use in the blade loads calculation. (3) Blade natural frequencies and normal modes are computed for use in calculating blade response. (Steps 2 and 3 are independent, and their order unimportant.) (4) Given the wake program input for use in blade loads calculations and the frequency program input for use in blade response calculations, the blade loads and response program is then used to calculate blade airloads and dynamic response. Response variables calculated include blade shears, moments, and angular and linear deflections (and their rates) as computed from the appropriate normal mode quantities and generalized coordinate magnitudes.

The blade frequency program listing is not contained in this report since it was not developed under this contract and is a proprietary program.

Program input and output is in English units.

---

*VOLUME I - MODEL AND RESULTS is contained in NASA Contractor Report CR-1911.

```
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION AO(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION NPTS(60)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RMOD(5)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VLL(16)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION VXX(01,01)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
      DIMENSION ZSTOR(800)
C
      COMMON /ITRG/ ITRGX
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
C
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /WK1A/ PSIR,DPSIK,PSIK,DELTA,RREF,MUCDS,MUSDS,THTAX,THTAY
      COMMON /WK1B/ XROT,YROT,ZROT,TCOR,ALFAT,ALFA1,ALFA2
```

```
      COMMON /WK1C/ PSI,CCLA,DIR
      COMMON /WK2A/ MSET,GAMMK,INDX,JAC,INDXL,SGMAZ,SIGMZ,SGMBL
      COMMON /SUBIC/ R,C,DTWOPI
      COMMON /SUBIE/ NAS
      COMMON /CONVGA/ EPSG,NWKRQ
      COMMON /AL1BDA/ ABK
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /OUTIN/ IN,OUT
      COMMON /WKCONT/ NWKPD
      COMMON /STEPXA/ WKPT,WW,IOUT,NOTTP1,KAT,NBC
      COMMON /WK4A/ VXX
      COMMON /MODWK1/ GAMMAM(54,16),RM(1),AM(05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /MODWK3/ AFM(4,10),BFM(4,4)
      COMMON /OUTDI/ NNTV,NEXPWK,NSIGRW,NMODR,NMODC
      COMMON /OUTDII/ NWKCLM
      COMMON /WK2GAM/ GAMFAC
      COMMON /VLIMIT/ VLIM(5),VMLIM(16)
      COMMON /ELNTHS/ ELL(16)
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /SUBIB/ KX,LSQ,DSQ,DTMP,LTMP,ATMP,BTMP
      COMMON /SUBID/ I,IP1,IM1
      COMMON /CONVGB/ SGMA1,INDXG
      COMMON /WK2C/ SIGBL
      COMMON /CONVGC/ GAMMAG
      COMMON /APXLDB/ LOADN
      COMMON /DART1/ SGRATO
      COMMON /VLNTHS/ NALIM,VLL
      COMMON /WKQ/ NUMXYZ
      COMMON /ZCNTRL/ NZS
      COMMON /ZSS/ ZSTOR
C
C
      DATA BLANK/1H /
      DATA NWKX,NWKY,NWKZ/3HWKX,3HWKY,3HWKZ/
C
  907 FORMAT(16X,3F8.8)
  906 FORMAT (29X,I5)
   16 FORMAT (20A4)
  901 FORMAT (19X,2X,8X,E10.8)
 9875 FORMAT (1X,8G16.7)
 8882 FORMAT (1H1,47X,37HFREE ROTOR WAKE GEOMETRY CALCULATIONS///
     11H ,26X,20A4/
     21H ,26X,20A4/
     31H ,26X,20A4//
     41H ,58X,16HBLADE PROPERTIES//
     51H ,9X,F8.3,30H  ADVANCE RATIO, DIMENSIONLESS,
     630X,F8.5,27H  MAXIMUM CONVERGENCE ERROR/
     71H ,9X,F8.3,31H  AIR MASS DENSITY, LB-SEC2/FT4,
     8   29X,F8.3,30H  REFERENCE ROTOR LENGTH, FEET/
```

```
     21H ,9X,F8.3,25H  BLADE ROOT RADIUS, FEET,
     1   35X,F8.3,35H  ROTATIONAL RATE OF ROTOR, RAD/SEC/
     61H ,9X,F8.3,26H  FORWARD VELOCITY, FT/SEC,
     5   34X,F8.3,30H  SLOPE OF LIFT CURVE,  1/RAD2/
     91H ,77X,F8.3,42H  VORTEX CORE RADIUS FACTOR, DIMENSIONLESS//)
 8883 FORMAT(
     11H ,6X,5HROTOR,63X,5HROTOR/
     21H ,3X,11HONE      TWO,57X,11HONE      TWO//
     31H ,F8.3,1X,F8.3,40H  AMPLITUDE OF LATERAL CYCLIC PITCH, RAD,
     411X,F8.3,1X,F8.3,44H  LATERAL SHAFT TILT ANGLE, POS TO PORT, RAD/
     51H ,F8.3,1X,F8.3,45H  AMPLITUDE OF LONGITUDINAL CYCLIC PITCH, RAD,
     66X, F8.3,1X,F8.3,41H  LONG FIRST HARMONIC FLAPPING ANGLE, RAD/
     71H ,F8.3,1X,F8.3,36H  ANGLE OF ATTACK AT BLADE ROOT, RAD,15X,F8.3,
     81X, F8.3,44H  LONG SHAFT TILT ANGLE, POS AFT FR VRT, RAD/
     91H ,F8.3,1X,F8.3,25H  BLADE CONING ANGLE, RAD,26X,F8.3,
     8    1X,F8.3,40H  MASS MOMENT OF INERTIA, FT-LB-SEC2/RAD )
 8889 FORMAT (
     21H ,F8.3,1X,F8.3,36H  BLADE INBOARD AIRFOIL RADIUS, FEET,15X,F8.3,
     11X, F8.3,43H  OFFSET OF HINGE FM CNTR OF ROTATION, FEET/
     91H ,F8.3,1X,F8.3,24H  BLADE MASS, LB-SEC2/FT,27X,F8.3,
     31X, F8.3,28H  ROTOR REFERENCE ANGLE, RAD/
     41H ,F8.3,1X,F8.3,24H  BLADE TWIST ANGLE, RAD,27X,F8.3,
     3    1X,F8.3,18H  ROTOR THRUST, LB/
     61H ,F8.3,1X,F8.3,13H  CHORD, FEET,
     538X,F8.3,1X,F8.3,31H  SHAFT TILT, POSITIVE AFT, RAD/
     81H ,F8.3,1X,F8.3,45H  DIRECTION OF ROTOR, POS IS COUNTERCLOCKWISE,
     76X,F8.3,1X,F8.3,29H  SPRING STIFFNESS, FT-LB/RAD)
 8887 FORMAT(
     3 1H ,F8.3,1X,F8.3,44H  EXTRAPOLATED ANGLE OF ATTACK AT SHAFT, RAD,
     7 7X,F8.3,1X,F8.3,31H  X-AXIS ROTOR COORDINATE, FEET/
     1       1H ,F8.3,1X,F8.3,38H  FEET FROM HINGE TO BLADE MASS CENTER,
     913X,F8.3,1X,F8.3,31H  Y-AXIS ROTOR COORDINATE, FEET/
     11H ,F8.3,1X,F8.3,44H  LATERAL FIRST HARMONIC FLAPPING ANGLE, RAD,
     2     7X,F8.3,1X,F8.3,31H  Z-AXIS ROTOR COORDINATE, FEET//// )
C
 8884 FORMAT(
     11H ,34HCOORDINATE TRANSFORMATION MATRICES,69X,
     225HPROGRAM CONTROL CONSTANTS//
     31H ,12X,9HROTOR ONE/
     41H ,99X,I3,17H  AZIMUTHAL STEPS/
     51H ,F7.3,2(6X,F7.3)/
     61H ,F7.3,2(6X,F7.3),66X,I3,18H  BLADES PER ROTOR/
     71H ,F7.3,2(6X,F7.3)/
     81H ,99X,I3,24H  INPUT CONTROL CONSTANT/
     91H ,12X, 9HROTOR TWO/
     11H ,99X,I3,22H  REV OF WAKE RETAINED/
     21H ,F7.3,2(6X,F7.3)/
     31H ,F7.3,2(6X,F7.3),66X,I3,8H  ROTORS/
     41H ,F7.3,2(6X,F7.3)/
     51H ,99X,I3,24H  TRAILED VORTICES/BLADE//)
```

```
 8885 FORMAT(1H ,5HRCAPS//
     11H ,10HROTOR ONE ,1X,9(1X,F7.4,1H ),1X,F7.4)
 8886 FORMAT(1H ,10HROTOR TWO ,1X,9(1X,F7.4,1H ),1X,F7.4//)
C
C
C
      IN=5
      OUT=6
      IOUT=7
      INTP1=4
      NOTTP1=8
      INTP2=4
      NOTTP2=4
C
C
C
      DO 2 I=1,90
    2 DI(I)=BLANK
C
C
C
C        READ INPUT.
C
      READ (5,16) NPTS
      READ (IN,906) NBC,NWKRQ,WW,NUWKPT,NTVM,NANRM,NA,NIB,NTV,NREV,NROT
      READ (IN,906) WKPT,ITRGX
      READ (IN,906) NALIM
      READ (IN,906) NUMXYZ
      READ (IN,901) SGRATO
      READ (IN,901) VOOMR,ABK,OM,V,RHO
      NTV1=NTV-1
      READ (IN,901)  (ALFA1(I),ALFA2(I),CHORD(I),ALFAS(I),DELTA(I),
     1THTAY(I),THTAX(I),PSIR(I),RZERO(I),DIR(I),I=1,NROT)
      READ (IN,901)((RCAP(I,J),J=1,NTV),I=1,NROT),RREF,CLA,EPSG
      READ (IN,901) (XROT(I),YROT(I),ZROT(I),I=1,NROT)
      NTVMP2=NTV*NROT
      READ(IN,901)  ((AFM(I,J),  J=1,NTVMP2),  I=1,NTVM)
      READ (IN,901) ((BFM(I,J),J=1,NTV1),I=1,NTVM)
      READ (IN,901) VLIM,VMLIM
      READ (5,901) GAMFAC
      READ (IN,901) ELIM
      IF (NUMXYZ.GT.0) READ (IN,901) (ZSTOR(I),I=1,NUMXYZ)
C
C        ITRGX = UPPER LIMIT ON NUMBER OF ITERATIONS FOR GAMMAS
C        ABR = VORTEX CORE RADIUS CONSTANT
C        SUBSCRIPT(M) = INCREMENTED NUMBER OF ROTORS
C
C
C
C        DEFINE CONSTANTS.
```

```
C
      INDXL=0
      PI=3.141593
      KAT=0
      TWOPI=2.*PI
      DTWOPI=1./TWOPI
      SGRATO=DTWOPI/SGRATO
      DPSI=TWOPI/NA
      DPSIK=TWOPI/NIB
      OMSQ=OM*OM
      MU=V/(OM*RREF)
      MUDP=MU*DPSI
C
C        READ ANGLES IN RADIANS OR CONVERT DEGREES TO RADIANS BEFORE
C             NEXT STEP.
C
      CY=COS(THTAY(1))
      SY=SIN(THTAY(1))
      CX=COS(THTAX(1))
      SX=SIN(THTAX(1))
C
C
C
C        COMPUTE COORDINATE  TRANSFORMATION MATRIX FOR USE IN LOCATION
C             OF MTH ROTOR.
C
C
      TCOR(1,1)=CY
      TCOR(2,1)=SY*SX
      TCOR(3,1)=-SY*CX
      TCOR(1,2)=0.
      TCOR(2,2)=CX
      TCOR(3,2)=SX
      TCOR(1,3)=SY
      TCOR(2,3)=-SX*CY
      TCOR(3,3)=CY*CX
      MBETR=3
C
C        COMPUTE TIP PATH ANGLE FROM APPROXIMATE EQUATIONS OF MOTION OF
C             RIGID, SPRING-HINGED BLADE.
C
      DO 15 M=1,NROT
      RO=RZERO(M)
      CHORD(M)=CHORD(M)/RREF
      PSI(M)=0.
      ALPHA1=ALFA1(M)
      ALPHA2=ALFA2(M)
      AS=ALFAS(M)
      F=CHORD(M)
      D=DELTA(M)
```

```
      NBETC=M
      NCALB=0
      IF (NCALB.EQ.1) CALL BETAS
      IF (NCALB.EQ.0)
     1READ (5,901) BETA(1,M),BETA(2,M),BETA(3,M),AO(M),AR(M)
      ALFAT(M)=ALFAS(M)-BETA(3,M)
      MUSDP(M)=MUDP*SIN(ALFAT(M))
      MUCDP(M)=MUDP*COS(ALFAT(M))
      MUSDS(M)=MUDP*SIN(ALFAS(M))
      MUCDS(M)=MUDP*COS(ALFAS(M))
   15 CCLA(M)=.5*CHORD(M)*CLA
      SINB3=SIN(BETA(3,1))
      COSB3=COS(BETA(3,1))
C
C         PRINT OUT INPUT AND CONTROL CONSTANTS
C
      WRITE (OUT,8882)NPTS,MU,EPSG,RHO,RREF,RO,OM,V,CLA,ABK
      WRITE (OUT,8883) ALFA1,THTAY,ALFA2,BETA(3,1),BETA(3,2),ALPHAO,
     1 THTAX,BETA(1,1),BETA(1,2),IO
      WRITE (OUT,8889) RZERO,DELTA,MB,PSIR,ALPHAR,LX,CHORD,ALFAS,DIR,KXX
      WRITE (OUT,8887) AO,XROT,XX,YROT,BETA(2,1),BETA(2,2),ZROT
C
      WRITE (OUT,8884) NA,TCOR(1,1),TCOR(1,2),TCOR(1,3),TCOR(2,1),
     1 TCOR(2,2),TCOR(2,3),NIB,TCOR(3,1),TCOR(3,2),TCOR(3,3),WKPT,
     2 NREV,T(1,1),T(1,2),T(1,3),T(2,1),T(2,2),T(2,3),NROT,T(3,1),
     3 T(3,2),T(3,3),NTV
      WRITE (OUT,8885) (RCAP(1,IX),IX=1,NTV)
      IF (NROT.EQ.2)
     1WRITE (OUT,8886) (RCAP(2,IX),IX=1,NTV)
      WRITE(6,30)
   30 FORMAT(4HOAFM/)
      WRITE (OUT,9875)((AFM(I,J),J=1,NTVMP2),I=1,NTVM)
      WRITE(6,31)
   31 FORMAT(4HOBFM/)
      WRITE (OUT,9875)((BFM(I,J),J=1,NTV1),I=1,NTVM)
      WRITE(6,32)
   32 FORMAT(33HOVLIM(NTV),VMLIM(NTVM*NIB),GAMFAC/)
      WRITE (OUT,9875) VLIM,VMLIM,GAMFAC
      WRITE(OUT,33)NALIM,ELIM
   33 FORMAT(12HONALIM  ELIM/2X,I3,G16.7)
      IF (NUMXYZ.GT.0) WRITE (OUT,9875) (ZSTOR(I),I=1,NUMXYZ)
C
C
C         DIMENSIONALIZATION FACTORS
C         LOAD RHO*OM*OM*R**3
C
      R11=RREF*RREF*RREF
      DFLOD=RHO*OMSQ *R11
C
C         CONSTANTS USED TO CONTROL PROGRAM.
```

```
C
      NJ=NIB*NROT*NTV
      NAR=NA*NREV
      NGJ=NIB*NTV1
      NGJR=NGJ*NROT
      NIBV=NIB*NTV
      NIBRV=NIBV*NROT
      NIBM=NIB*NTVM
      NBRV1=NIBRV+1
      NAS=2
      NLP1=NROT
      NLP2=NIB
      NLP3=NTV
      NWKPD=0
      NPER=1
      NR=NTV1*NROT
      NANR=NA*NR
      NIBNA=NA/NIB
      NIBRVM=NTVM*NIB*NROT
      NWSTRE=2
      NWR=NA*NREV
      NNTV=20
      NEXPWK=5
      NSIGRW=16
      NWKRW=5
      NWKCLM=20
      NMODR=54
      NMODC=16
      NNROT=2
      NNTVM=10
      NTVMX=4
C
C        COMPUTE LENGTHS FOR USE IN MODIFIED WAKE
C
      N1=NIB*NTVM
      KK=0
      DO 10 M=1,NROT
      CALL MODRM (NTV,NTVM,RCAP,RMOD,AFM,NNROT,NNTVM,NTVMX,M,NWKRW)
      DO 5 JJ=1,NTVM
      KK=KK+1
      ELL(KK)=RMOD(JJ)*DPSI/RREF
    5 VLL(KK)=ELIM*ELL(KK)
      N3=(M-1)*N1
      N2=N1+N3
      N3=NTVM+1+N3
      DO 10 JJ=N3,N2
      KK=KK+1
      ELL(KK)=ELL(KK-NTVM)
   10 VLL(KK)=ELIM*ELL(KK)
      WRITE(6,34)
```

```
   34 FORMAT(12HOVLL(NIBRVM)/)
      WRITE (OUT,9875) (VLL(I),I=1,NIBRVM)
C
C        DEFINE BLADE POSITIONS AND CIRCULATIONS FOR WKPT.GE.3, CARD REA
C             WKPT=2, TAPE READ FOR WKPT=1
C
      IF (NWKRQ.EQ.0) GO TO 20
      NWKCL=1
      NWKLST=0
      DO 13 I=1,NUWKPT
      NWKLST=NWKLST+1
      IF (NWKLST.LE.NWKRW) GO TO 14
      NWKLST=1
      NWKCL=NWKCL+1
   14 READ (IN,907) WKX(NWKLST,NWKCL),WKY(NWKLST,NWKCL),WKZ(NWKLST,
     1NWKCL)
   13 CONTINUE
      IF (NWKCL.EQ.1) NWKRW=NWKLST
      IF (NWKLST.EQ.NWKRW.OR.NWKCL.EQ.1) GO TO 200
      NXX=NWKLST+1
      DO 22 I=NXX,NWKRW
      WKX(I,NWKCL)=0.
      WKY(I,NWKCL)=0.
   22 WKZ(I,NWKCL)=0.
      CALL MPRECT (NWKX,WKX,NWKRW,NWKCL,NWKRW,NWKCLM)
      CALL MPRECT (NWKY,WKY,NWKRW,NWKCL,NWKRW,NWKCLM)
      CALL MPRECT (NWKZ,WKZ,NWKRW,NWKCL,NWKRW,NWKCLM)
  200 DO 18 I=1,NWKRW
      DO 18 J=1,NWKCL
      VXX(I,J)=0.
   18 VI(I,J)=0.
   20 NAS1=0
      NW=1
      II=1
C
C        READ FOR STARTING DATA ON TAPE
C
C
C        PERFORM A STEP
C
 9888 WKPT=3
      CALL WK1
      STOP
      END
```

```
      SUBROUTINE WK1
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
      DIMENSION ZSTOR(800)
C
C
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /WKQ/ NUMXYZ
      COMMON /ZCNTRL/ NZS
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
C
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /WK1A/ PSIR,DPSIK,PSIK,DELTA,RREF,MUCDS,MUSDS,THTAX,THTAY
      COMMON /WK1B/ XROT,YROT,ZROT,TCOR,ALFAT,ALFA1,ALFA2
      COMMON /WK1C/ PSI,CCLA,DIR
```

```
      COMMON /SUBIC/ R,C,DTWOPI
      COMMON /APXLDA/ RBAR
      COMMON /ZSS/ ZSTOR
C
 9876 FORMAT (1HO,/8(1X,G14.7))
C
C        COMPUTATIONS FOR REFERENCE ROTOR
C
C
C        COMPUTE BLADE POINTS FOR II=1, WAKE POINTS FOR II=2, FOR EACH
C             BLADE.
C
      OUT=6
      NZS=0
   21 II=II+1
      J=0
      DO 28 K=1,NIB
      PSIK=(K-1)*DPSIK
      WORK1=PSI(1)+PSIK
      SINPK=SIN(WORK1)
      COSPK=COS(WORK1)
      BK=BETA(1,1)+BETA(2,1)*SINPK+BETA(3,1)*COSPK
      COSBK=COS(BK)
      SINBK=SIN(BK)
      DO 28 I=1,NTV
      J=J+1
      NZS=NZS+1
      RSCRP=(RCAP(1,I)-DELTA(1))/RREF
      RSMAL=DELTA(1)/RREF+RSCRP*COSBK
      RSMLL(1,J)=RSMAL
      X(II,J)=RSMAL*COSPK
      Y(II,J)=RSMAL*SINPK
      Z(II,J)=RSCRP*SINBK
   25 IF (NUMXYZ.GT.0) Z(II,J)=ZSTOR(NZS)
      IF (II.EQ.1) GO TO 28
   27 X(2,J)=X(2,J)+MUCDS(1)
      Z(2,J)=Z(2,J)+MUSDS(1)
      IF (NUMXYZ.GT.0) Z(2,J)=ZSTOR(NZS)
   28 CONTINUE
C
C        COMPUTATION FOR ADDITIONAL ROTORS
C        ANGLES MUST BE IN RADIANS, EITHER READ RADIANS OR CONVERTED
C             FROM DEGREES.  T(I,J) IS COORDINATE TRANSFORMATION MATRIX
C             FOR MTH ROTOR.
C
   29 IF (NROT.LE.1) GO TO 50
   30 DO 39 M=2,NROT
      CY=COS(THTAY(M))
      CX=COS(THTAX(M))
      SY=SIN(THTAY(M))
```

```
      SX=SIN(THTAX(M))
      T(1,1)=CY
      T(2,1)=0.
      T(3,1)=SY
      T(1,2)=SY*SX
      T(2,2)=CX
      T(3,2)=-SX*CY
      T(1,3)=-SY*CX
      T(2,3)=SX
      T(3,3)=CY*CX
      DO 39 K=1,NIB
      PSIK=PSIR(M)+(K-1)*DPSIK
      COSPK=COS(PSI(M)+PSIK)
      SINPK=SIN(PSI(M)+PSIK)
      BK=BETA(1,M)+BETA(2,M)*SINPK+BETA(3,M)*COSPK
      COSBK=COS(BK)
      SINBK=SIN(BK)
      DO 39 I=1,NTV
      J=J+1
      NZS=NZS+1
      RSCRP=(RCAP(M,I)-DELTA(M))/RREF
      RSMAL=DELTA(M)/RREF+RSCRP*COSBK
      RSMLL(2,J)=RSMAL
      C(1)=RSMAL*COSPK
      C(2)=RSMAL*SINPK
      C(3)=RSCRP*SINBK
C
C        COMPUTE POSITION OF MTH ROTOR IN COORDINATE SYSTEM OF ROTOR 1
C
      TR(1)=XROT(M)
      TR(2)=YROT(M)
      TR(3)=ZROT(M)
      DO 32 L=1,3
      DO 32 JJ=1,3
   32 TR(L)=TR(L)+T(L,JJ)*C(JJ)
      X(II,J)=0.
      Y(II,J)=0.
      Z(II,J)=0.
C
C              COMPUTE BLADE POINTS FOR II=1, WAKE POINTS FOR II=2, FOR
C              EACH BLADE.
C
      DO 34 L=1,3
      X(II,J)=X(II,J)+TCOR(1,L)*TR(L)
      Y(II,J)=Y(II,J)+TCOR(2,L)*TR(L)
      Z(II,J)=Z(II,J)+TCOR(3,L)*TR(L)
      IF (NUMXYZ.GT.0) Z(II,J)=ZSTOR(NZS)
   34 CONTINUE
      IF (II.LE.1) GO TO 39
   38 X(2,J)=X(2,J)+MUCDS(M)
```

```
      Z(2,J)=Z(2,J)+MUSDS(M)
      IF (NUMXYZ.GT.0) Z(02,J)=ZSTOR(NZS)
   39 CONTINUE
C
C        DEFINE CIRRCULATIONS BY CALCULATION OR READ FROM CARDS, TAPE.
C
   50 J=0
   51 JJ=0
      IF (NUMXYZ.GT.0) WRITE (6,9876)  (Z(II,JH),JH=1,NIBRV)
      DO 60 M=1,NROT
      MUALT=MU*ALFAT(M)
      DO 60 K=1,NIB
      PSIK=(K-1)*DPSIK+PSIR(M)+PSI(M)
      SINPK=SIN(PSIK)
      COSPK=COS(PSIK)
      BK=BETA(1,M)+BETA(2,M)*SINPK+BETA(3,M)*COSPK
      COSBK=COS(BK)
      ALFBR=AO(M)+ALFA1(M)*SINPK+ALFA2(M)*COSPK
      MUSPK=MU*SINPK
      DO 60 I=1,NTV1
      JJ=JJ+1
      RSMAL=(DELTA(M)+(RCAP(M,I)-DELTA(M))*COSBK)/RREF
      RBAR(JJ)=RSMAL+.5*(RCAP(M,I+1)-RCAP(M,I))*COSBK/RREF
      ALFBR=ALFBR-AR(M)*RBAR(JJ)
      GAMMA(II,JJ)=CCLA(M)*(ALFBR*(RBAR(JJ)+MUSPK*DIR(M))+MUALT)
   60 CONTINUE
      WRITE (6,9876) (GAMMA(1,JX),JX=1,NGJR)
C
C        ENTER TYPICAL AZIMUTHAL STEP COMPUTATION
C
   70 IF (II.GT.1) GO TO 72
      CALL WK2
      CALL CONVG
      CALL AL1BD2
      CALL APXLD
      CALL WK3
      CALL STEPX
      GO TO 21
   72 II=0
      NW=NW+1
      WRITE (6,9875) NW
 9875 FORMAT (1HO,I4,4H WK1)
      DO 73 M=1,NROT
   73 PSI(M)=PSI(M)+DPSI*DIR(M)
      GO TO 21
      END
```

```
      SUBROUTINE MATINV
      DIMENSION A(16,16)
      DIMENSION B(16,1)
      DIMENSION INDEX(16,3)
      COMMON /BETA2/A,B,N,M,DETERM
      EQUIVALENCE                                    (AMAX,T,SWAP)
C        INITIALIZATION
   10 DETERM=1.
   15 DO 20 J=1,N
   20 INDEX(J,3)=0.
   30 DO 550 I=1,N
C        SEARCH FOR PIVOT ELEMENT
   40 AMAX=0.
   45 DO 105 J=1,N
      IF(INDEX(J,3)-1)60,105,60
   60 DO 100 K=1,N
      IF (INDEX(K,3)-1) 80,100,715
   80 IF (AMAX-ABS(A(J,K))) 85,100,100
   85 IROW=J
   90 ICOLUM=K
      AMAX=ABS(A(J,K))
  100 CONTINUE
  105 CONTINUE
      INDEX(ICOLUM,3)=INDEX(ICOLUM,3)+1
  260 INDEX(I,1)=IROW
  270 INDEX(I,2)=ICOLUM
  130 IF (IROW-ICOLUM) 140,310,140
  140 DETERM=-DETERM
  150 DO 200 L=1,N
  160 SWAP=A(IROW,L)
  170 A(IROW,L)=A(ICOLUM,L)
  200 A(ICOLUM,L)=SWAP
      IF (M) 310,310,210
  210 DO 250 L=1,M
  220 SWAP=B(IROW,L)
  230 B(IROW,L)=B(ICOLUM,L)
  250 B(ICOLUM,L)=SWAP
C        DIVIDE PIVOT ROW BY PIVOT ELEMENT
  310 PIVOT=A(ICOLUM,ICOLUM)
      DETERM=DETERM*PIVOT
  330 A(ICOLUM,ICOLUM)=1.
  340 DO 350 L=1,N
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT
  355 IF (M) 380,380,360
  360 DO 370 L=1,M
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT
C        REDUCE NON-PIVOT ROWS
  380 DO 550 L1=1,N
  390 IF (L1-ICOLUM) 400,550,400
  400 T=A(L1,ICOLUM)
```

```
420 A(L1,ICOLUM)=0.
430 DO 450 L=1,N
450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T
455 IF (M) 550,550,460
460 DO 500 L=1,M
500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T
550 CONTINUE
715 ID=2
740 RETURN
    END
```

```
      SUBROUTINE MPRECT(NI,R,L,M,LD,MD)
      REAL R(LD,MD)
      J1=0
      J2=0
      JSEC=0
    1 J1=J2+1
      J2=J1+7
      IF(J2.LE.M)GO TO 2
      J2=M
    2 JSEC=JSEC+1
      WRITE(6,3)NI,JSEC
    3 FORMAT (1HO,A4,1X,6HMATRIX,8X,7HSECTION,I3)
      WRITE(6,4)(I,I=J1,J2)
    4 FORMAT (2X,3HROW,I13,7I14/)
      DO 5 I=1,L
    5 WRITE(6,6)I,(R(I,K),K=J1,J2)
    6 FORMAT(I4,4X,8F14.4)
      IF(J2.LT.M)GO TO 1
      RETURN
      END
```

```
      SUBROUTINE MODRM (NTV,NTVM,RCAP,RMOD,AFM,NNROT,NNTVM,NTVMX,M,NWKR)
      DIMENSION RCAP(NNROT,NWKR ),RMOD(NWKR ),AFM(NTVMX,NNTVM)
      DO 10 LM=1,NTVM
      RMOD(LM)=0.
      DO 10 L=1,NTV
      N=L+NTV*(M-1)
   10 RMOD(LM)=AFM(LM,N)*RCAP(M,L)+RMOD(LM)
      RETURN
      END
```

```
      SUBROUTINE BETAS
C     THE INPUTS TO THIS SUBROUTINE ARE COMMON TO OTHER SUBROUTINES
      REAL MU,MU2,MU4,MBDXB,MB,M00,L,M01,M02,MBB,K1M10,K2M10,M11,M12,
     1M13,M20,M20C,M20D,M115,IO,K
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION BETA(3,2)
      DIMENSION D3(09)
      DIMENSION D33(016,016)
      DIMENSION XBETA(016)
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,R,C,AS,ALPHA1,ALPHA2,RO
C        MBETR = NUMBER OF BETA ROWS
C        NBETC = NUMBER OF BETA COLUMNS
      COMMON /BETA2/D33,XBETA,N,M,DETERM
C        COMMON.BETA2 CONTAINS THOSE VARIABLES USED BY MATINV
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      EQUIVALENCE (D33(1,1),D3(1))
C     BLADE RIGID BODY CALCULATIONS FOR BLADE WAKE AND ESTIMATED LOADS
C     THE RIGID BODY BLADE FLAPPING MOTIONS ARE GIVEN BY BETA(OMEGA*T),
C        WHERE BETA(OMEGA*T)=BETA(1)+BETA(2)*SIN(OMEGA*T)+BETA3*COS(
C        OMEGA*T) OR SINCE PSI=OMEGA*T...
C        BETA(PSI)=BETA(1)+BETA(2)*SIN(PSI)+BETA(3)*SIN(PSI)
C        PSI=OMEGA*T=0 AT THE X-AXIS
C     THE BETA(1), BETA(2), BETA(3) ARE DEFINED BY THE EQUATION WHERE
C        THE MATRIX OF COEFFICIENTS POST-MULTIPLIED BY THE BETA COLUMN
C        VECTOR = THE COLUMN VECTOR(M00,M01,M02)
C     ALPHA = BLADE SECTION ANGLE OF ATTACK, RADIANS
C           = AO+AR*R+ALPHA1*SIN(OMT)+ALPHA2*COS(OMT)
C     ALPHAO = GEOMETRIC ANGLE OF ATTACK AT BLADE ROOT, RADIANS
C     ALPHAR = TOTAL DECREASE IN ANGLE OF ATTACK TOTAL BLADE TWIST
C        ANGLE, RADIANS
C     ALPHA1 = AMPLITUDE OF LATERAL CYCLIC PITCH, RADIANS
C     ALPHA2 = AMPLITUDE OF LONGITUDINAL CYCLIC PITCH, RADIANS
C     AS = SHAFT TILT, POSITIVE AFT, RADIANS
C     C = CHORD, FEET
C     D = OFFSET OF HINGE FROM CENTER OF ROTATION, FEET
C     IO = MASS MOMENT OF INERTIA OF BLADE ABOUT THE BLADE HINGE, FT-LB
C        -SEC2/RAD
C     K = SPRING STIFFNESS, FT-LB/RAD
C     L = LIFT OR TRUST OF ROTOR
C     MB = BLADE MASS, LB-SEC2/FT
C     OMEGA = OM = ROTATIONAL RATE OF ROTOR, RAD/SEC
C     R = ROTOR RADIUS, FEET
C     RHO = AIR MASS DENSITY, LB-SEC2/FT4
C     RO = BLADE ROOT RADIUS, FEET
C     V= FORWARD VELOCITY, FT/SEC
C     XB = DISTANCE FROM HINGE TO BLADE MASS CENTER, FEET
      READ    (5,1) K,IO,MB,XB,ALPHAO,ALPHAR,L,R
    1 FORMAT (29X,E10.8)
      RX=R-RO
```

```
      AO(NBETC)=ALPHAO+ALPHAR*RO/RX
      AR(NBETC)=ALPHAR/RX
      VOM=V*OM
      VV=V*V
      RORO=RO*RO
      RORORO=RORO*RO
      RO4=RORORO*RO
      RR=R*R
      RRR=RR*R
      RRRR=RRR*R
      A1AS=ALPHA1+AS
      MU2=MU*MU
      MU4=MU2*MU2
      PIROCL=PI*RHO*C
      R1=PIROCL*(R-RO)
      R2=PIROCL*(RR-RORO)/2.
      R3=PIROCL*(RRR-RORORO)/3.
      R4=PIROCL*(RRRR-RO4)/4.
      R5=PIROCL*(RRRR*R-RO4*RO)/5.
      R2DR1=R2-D*R1
      R3DR2=R3-D*R2
      R4DR3=R4-D*R3
      MBDXB=MB*D*XB
      C2=C*.5
      CT=L/(PI*RHO*OMSQ*RRRR)
      WI= SQRT(CT*CT+MU4)
      IF (WI.LT.MU2) STOP
      WI=OM*R*(.5*SQRT(WI-MU2))
      RX=VV*.5
      A1=AO(NBETC)
      A2=AR(NBETC)
      M00=R2DR1*RX*A1    +R3DR2*(VOM*A1AS-OM*WI-A2*RX   )+R4DR3*OMSQ*A1-R5
     1*OMSQ*A2
      M01=R2DR1*(.75*VV*ALPHA1-V*WI+VV*AS)+R3DR2*VOM*A1*2.+R4DR3*(OMSQ*
     1ALPHA1-2.*OM*A2*V)
      M02=(R2DR1*VV*.25+R4DR3*OMSQ)*ALPHA2
      MBB=MBDXB-R3DR2*C2
      K1M10=K+OMSQ*(IO+MBB)
      K2M10=K1M10-OMSQ*IO
      M11=R2DR1*VOM*C2
      M12=-R3DR2*VOM
      M13=-R2DR1*RX
      M20=-R4DR3*OM
      RX=M13*.5
      RX1=M20*OM
      M20C=RX1+RX
      M20D=RX1-RX
      M115=M11*.5
      D33(1,1)=K1M10
      D33(2,1)=-M11
```

```
      D33(3,1)=-M12
      D33(1,2)=-M115
      D33(2,2)=K2M10
      D33(3,2)=-M20C
      D33(1,3)=0.
      D33(2,3)=M20D
      D33(3,3)=K2M10
      XBETA(1)=M00
      XBETA(2)=M01
      XBETA(3)=M02
      N=3
      M=1
      CALL MATINV
      DO 2 I=1,MBETR
    2 BETA(I,NBETC)=XBETA(I)
      RETURN
      END
```

```
      SUBROUTINE WK2
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
C
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /WK1C/ PSI,CCLA,DIR
      COMMON /WK2A/ MSET,GAMMK,INDX,JAC,INDXL,SGMAZ,SIGMZ,SGMBL
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /SUBIB/ KX,LSQ,DSQ,DTMP,LTMP,ATMP,BTMP
      COMMON /SUBIC/ R,C,DTWOPI
      COMMON /SUBID/ I,IP1,IM1
      COMMON /CONVGB/ SGMA1,INDXG
```

```
      COMMON /WK2C/ SIGBL
      COMMON /MODWK1/ GAMMAM(54,16),RM(1),AM(05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /SUBIE/ NAS
      COMMON /OUTDI/ NNTV,NEXPWK,NSIGRW,NMODR,NMODC
      COMMON /WK2GAM/ GAMFAC
      COMMON /DART1/ SGRATO
C
C
      DATA INX1,INX2,INX3,INX4,INX5,INX6,INX7/
     14H   X,4H   Y,4H   Z,4H  VX,4H  VY,4H  VZ,4HSIGA/
C
 9876 FORMAT (1HO,I14/(8(1X,G14.7)))
C
C
C        NEW COORDINATES FOR P1J AND LOAD COMPONENTS OF GAMMA(1,J) ARE
C             NOW COMPUTED, WAKE AND SELF-INDUCED VELOCITY COMPONENTS
C             OF GAMMA(1,J) ARE TO BE COMPUTED
C
C        DEFINITION OF VZJ(RBARJ,PSII)
C
C        VZS AND SIGMAS ARE DEFINED AS FOLLOWS.  THE EFFECT OF THE WAKE
C             FROM ONE BLADE IS COMPUTED, ONE ROW OF SHED VORTICES AND
C             THE TRAILING VORTICES JUST AHEAD OF THEM TAKEN PER TIME.
C             THE DISTANCES FROM THE POINT WHERE THE INDUCED VELOCITY IS
C             COMPUTED AND THE VORTEX ELEMENT IS LOCATED, AND THE
C             ORIENTATION OF THE VORTICES ARE SUCH THAT ESSENTIALLY
C             THE SAME FORMULA CAN BE USED TO COMPUTE THE EFFECT OF
C             BOTH SHED AND TRAILING VORTEX ELEMENTS.  XA, XB, XC, ARE
C             THE X COORDINATES OF THE POINT AT WHICH THE INDUCED
C             VELOCITY IS COMPUTED, AND THE END POINTS OF THE VORTEX
C             ELEMENT UNDER CONSIDERATION.
C
C
      J=0
      M=1
      IF (NW.GT.2) GO TO 83
      DO 81 JX=1,NIBRV
   81 A(1,JX)=CHORD(M)
      DO 82 JX=1,NGJR
   82 B(2,JX)=CHORD(M)
   83 DO 140 M=1,NROT
      NPSI(M)=(PSI(M)/DPSI)+.5
C
C     T44 TESTS TO SEE IF VAR IS IN RANGE
C
      CALL T44(NPSI(M),NA)
      NSET = NR * (NPSI(M)  )+(M-1) * NTV1
      DO 140 K=1,NIB
```

```
      T45 = (K-1)*NA/NIB
      MSET = NSET + NR*T45
      CALL T44(MSET,NANR)
      JKL=(K-1)*NTV+(M-1)*NTV*NIB
      DO 140 L=1,NTV1
      MSET=MSET+1
      J=J+1
      INDXG(J)=MSET
      JKL=JKL+1
      JP1=JKL+1
      LP1=L+1
C
C     INITIALIZE SIGBL FOR BLADE LOADS
C
      DO 84 IND=1,NANR
84    SIGBL(IND) = 0.0
      JAC = 0
C
C        A IS THE POINT AT WHICH INDUCED VELOCITIES ARE TO BE COMPUTED.
C
      XA=.5*(X(1,JKL)+X(1,JP1))
      YA=.5*(Y(1,JKL)+Y(1,JP1))
      ZA=.5*(Z(1,JKL)+Z(1,JP1))
      VZ(1,J)=0.
      JSIG=0
      MODWK=0
      N1=NIBRV
      N2=NTV
      KX=M*K*L
   91 DO 138 JA=1,N1,N2
      JSIGT=1+(JA-1)*NTV1/NTV
      JAC = JAC +1
C
C        COMPUTE R FOR CURRENT BLADE
C
      IF (MODWK.EQ.0) GO TO 86
      JB=JB+1
      IF (JB.GT.NTVM) JB=1
      JAC=(JA-1)/NTVM+1
      XB=XM(NANRM,JA)
      YB=YM(NANRM,JA)
      ZB=ZM(NANRM,JA)
      WORK1=XB-XA
      WORK2=YB-YA
      WORK3=ZB-ZA
      RM(1)=SQRT(WORK1*WORK1+WORK2*WORK2+WORK3*WORK3)
      GO TO 137
   86 DO 87 JL=1,NTV
      JK=JL+JA-1
      XB=X(1,JK)
```

```
      YB=Y(1,JK)
      ZB=Z(1,JK)
      WORK1=XA-XB
      WORK2=YA-YB
      WORK3=ZA-ZB
      RT=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
   87 R(JL)=SQRT(RT)
C
C        INITIALIZE TEMPORARY STORAGE LOCATIONS.
C
   88 DO 89 NN=1,NTV
      LTMP(NN)=0.
      ATMP(NN)=CHORD(M)
      DTMP(NN)=0.
   89 BTMP(NN)=CHORD(M)
      NWM1=NW-1
  137 SGMAZ=0.
      INDX = NPSI(M) + (JAC-1)*NA/NIB+1
C
C        LOCATE POINTS B AND C.
C
      IF (MODWK.EQ.0) CALL SUBI
      IF (MODWK.EQ.1) CALL SUBII(NPER,NA,INDX,NR,JAC,NIB,NROT,NTV1,VZ,
     1 JA,J,JB)
  138 CONTINUE
      MODWK=MODWK+1
      N1=NIBRVM
      N2=1
      JB=0
      IF (MODWK.EQ.1.AND.NAS.GT.NANRM) GO TO 91
      IF ( NPER.NE.4) GO TO 139
      SXG=SIGBL(MSET)*SGRATO
      SXX=ABS(SXG)
      DO 7447 IND=1,NANR
      SIGBL(IND)=SIGBL(IND)*DTWOPI
      SXY=ABS(SIGBL(IND))
      IF ((SXY.GT.SXX).AND.IND.NE.MSET) SIGBL(IND)=SIGBL(IND)/SXY*SXX
 7447 CONTINUE
      WRITE (4) MSET,(SIGBL(IND),IND=1,NANR)
      WRITE (6,9876) MSET,(SIGBL(IND),IND=1,NANR)
C
C        LNTH(1,J) AND A(1,J), DNTH(2,JSIG) AND B(2,JSIG) ARE NOT
C             COMPUTED UNTIL STATEMENTS 150 THRU 166.
C        DNTH(1,JSIG) AND B(1,JSIG) ARE NOT YET NEEDED, AS THEY INVOLVE
C             THE VORTICES AT THE BLADE AND KNOWN LENGTHS BETWEEN THE
C             R(J)S.
C        COMPUTE GAMMA, INDUCED WAKE VELOCITY, EXCEPT GAMMA(1,J) AND
C             LOAD COMPONENTS.
C
  139 WORKX=ABS(VZ(1,J))
```

```
      PX=GAMFAC
      IF (WORKX.GT.PX) VZ(1,J)=VZ(1,J)/WORKX*GAMFAC
  140 GAMMK(1,J)=VZ(1,J)*CCLA(M)/TWOPI+GAMMA(1,J)
      CALL MPRECT (INX7,SGMA1,NGJR,NGJR,NSIGRW,NSIGRW)
      WRITE (6,9876) NPER,(GAMMK(1,JX),JX=1,NGJR)
      RETURN
      END
```

```
      SUBROUTINE T44 (A,NA)
      INTEGER   A
1     IF(A.LT.0.0) GO TO 2
      IF(A.GE.NA) GO TO 3
      RETURN
2     A=A + NA
      GO TO 1
3     A = A - NA
      GO TO 1
      END
```

```
      SUBROUTINE SUBI
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION GAMMK(1,016)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
C
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /WK1C/ PSI,CCLA,DIR
      COMMON /WK2A/ MSET,GAMMK,INDX,JAC,INDXL,SGMAZ,SIGMZ,SGMBL
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /WK2C/ SIGBL
      COMMON /SUBIB/ KX,LSQ,DSQ,DTMP,LTMP,ATMP,BTMP
      COMMON /SUBIC/ R,C,DTWOPI
      COMMON /SUBID/ I,IP1,IM1
```

```
      COMMON /SUBIE/ NAS
      COMMON /CONVGB/ SGMA1,INDXG
C
      DO 135 I=1,NWM1
      IP1=I+1
      XC=X(IP1,JA)
      YC=Y(IP1,JA)
      ZC=Z(IP1,JA)
      XB=X(I,JA)
      YB=Y(I,JA)
      ZB=Z(I,JA)
      NN=1
      N=1
      SIGN=1.
      JJ=JA
      JSIG=JSIGT-1
      ASSIGN 100 TO NCNTR
      WORK1=XA-XC
      WORK2=YA-YC
      WORK3=ZA-ZC
      RT=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
C
C        COMPUTE SQUARE ROOT OF RT AT THE APPROPRIATE STEP.
C
      IF(NPER.NE.4) GO TO 90
85    IGN = 0
      IF(INDX-I.GE.NA) IGN = -1
      IF(INDX-I.LT.0) IGN = 1
      INDX = INDX + NA * IGN
      IF ( IGN.NE.0) GO TO 85
      INDXL= (INDX-I) * NR + ((JAC-1)/NIB) * (NR/NROT)
C
   90 RS=SQRT(RT)
      GO TO NCNTR, (100,114,115,117,118)
C
C        COMPUTE NV*G, TRAILING VORTEX CONTRIBUTION, OR ETA*H, SHED
C             VORTEX CONTRIBUTION, L2 OR D2, AND ADD TO PREVIOUS CON-
C             TRIBUTIONS FROM QUADRILATERAL FOR GAMMA(I,JSIG).
C
  100 RPR=RS+R(NN)
      DORL=(XB-XC)**2+(YB-YC)**2+(ZB-ZC)**2
      ASSIGN 94 TO IORGT
      WORK1=R(NN)
      VTEST=RS*RS+WORK1*WORK1-DORL
      IF (VTEST.GT.0) GO TO 101
      WORK2=RS-WORK1
      WORK2=WORK2*WORK2
      WORK3=RS+WORK1
      WORK3=WORK3*WORK3
      VTEST=(WORK3-DORL )*(DORL -WORK2)/(4.*DORL )
```

```
      IF (I.NE.1) GO TO 92
      WORK1=CHORD(M)
      GO TO 93
   92 IF (SIGN.GT.0) WORK1=A(I-1,JJ)
      IF (SIGN.LT.0) WORK1=B(I  ,JSIG+1)
   93 WORK1=WORK1*WORK1
      IF (VTEST.GT.WORK1) GO TO 101
      HORG=0.
      WORK5=SQRT(DORL)*WORK1
      IF(WORK5.NE.0.) HORG=1./WORK5
      ASSIGN 95 TO IORGT
      GO TO 103
  101 HORG=0.
      WORK5=R(NN)*RS*(RPR*RPR-DORL)
      IF(WORK5.NE.0.) HORG=SIGN*RPR/WORK5
  103 EORNZ=(XA-XC)*(YC-YB)-(YA-YC)*(XC-XB)
      EHNGZ=EORNZ*HORG
      GO TO IORGT, (94,95)
   95 SIGM=1.
      IF (EHNGZ.LT.0.) SIGM=-1.
      JX=JSIG
      IF (N.LE.2) JX=JSIG+1
      WORK1=GAMMA(I,JX)
      WORK2=EHNGZ*WORK1
      IF (ABS(WORK2).GT.1.) EHNGZ=SIGM/WORK1
   94 SGMAZ=EHNGZ+SGMAZ
   96 IF (N-2) 102,104,106
C
C        STORE R, L2, COMPUTE B FOR SHED VORTEX CONTRIBUTION.
C
  102 R(NN)=RS
      LLNTH(NN)=DORL
      JJ=JJ+1
      INDXL= INDXL+ 1
      XB=X(IP1,JJ)
      YB=Y(IP1,JJ)
      ZB=Z(IP1,JJ)
      WORK1=XA-XB
      WORK2=YA-YB
      WORK3=ZA-ZB
      RT=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
      N=2
      SIGN=-1.
      GO TO 90
C
C        STORE D2, COMPUTE C FOR TRAILED VORTEX CONTRIBUTION.
C
  104 C(3)=EHNGZ
      JSIG=JSIG+1
      XC=X(I,JJ)
```

```
      YC=Y(I,JJ)
      ZC=Z(I,JJ)
      N=3
      DLNTH(NN  )=DORL
      NN=NN+1
      SIGN=1.
      GO TO 100
C
C        REDEFINE C.
C
  106 XC=XB
      YC=YB
      ZC=ZB
      IF (I-1) 107,107,108
C
C        GAMMA(1,J)S ARE UNKNOWN.  IF I=1, STORE SIGMA(J,JSIG).
C        THIS IS SIGMA(1,PSII,J,JSIG)
C
  107 SGMA1(J,JSIG)=SGMAZ*CCLA(M)*DTWOPI
      SGMBL = SGMAZ
      GO TO 109
C
C        GAMMA(1,J) ARE KNOWN FOR I.GT.1.    COMPUTE VZ(1,J).
C
  108 VZ(1,J)=VZ(1,J)+(SGMAZ-SIGMZ(NN-1))*GAMMA(I,JSIG)
      IF( NPER.NE.4) GO TO 109
      SGMBL = SGMAZ - SIGMZ(NN-1)
      SGMA2(I,JSIG) = SGMBL
C
C        CONVENIENT LOCATION TO COMPUTE SIGMA(J,PSII,I,JJ)
C        POSSIBLE ADDED CODING
C
  109 SIGMZ(NN-1)=C(3)
      IF (NPER.NE.4) GO TO 112
      SIGBL(INDXL)= SGMBL + SIGBL(INDXL)
C
C        IF THE ROW IS NOT COMPLETED, STORE SGMAZ AND CONTINUE WITH VZ
C             COMPUTATION.
C
  112 IF (NN.GT.NTV1) GO TO 111
  110 SGMAZ=-EHNGZ
      GO TO 102
C
C        COMPUTE NEW VORTEX CORE RADII FOR I.GT.1, OTHERWISE STORE L2
C             AND D2.
C
  111 R(NN)=RS
      SGMAZ=0.
      LLNTH(NN)=DORL
      IF (KX.GT.1) GO TO 135
```

```
      IF (I.LE.1) GO TO 122
      IM1=I-1
      NN=1
C
C        N IS REPLACING J FROM STATEMENTS 111-138.
C
      N=JA
      JSIG=JSIGT
  113 RT=LLNTH(NN)
      RS=SQRT(RT)
C
C        STORE L.
C
  114 LLNTH(NN)=RS
      RT=LNTH(I-1,N)/RS
      RS=SQRT(RT)
C
C        COMPUTE TRAILING VORTEX CORE RADII.
C
  115 AT(NN)=RS*A(I-1,N)
      IF (NN.GT.NTV1) GO TO 119
C
C        THERE IS NO B(NTV) SO DO NOT COMPUTE IT, OTHERWISE COMPUTE B
C             TRAILING VORTEX CORE RADII.
C
  116 RT=DLNTH(NN)
      RS=SQRT(RT)
  117 RT=DNTH(I   ,JSIG)/RS
      DLNTH(NN)=RS
      RS=SQRT(RT)
  118 BT(NN)=RS*B(I   ,JSIG)
      N=N+1
      JSIG=JSIG+1
      NN=NN+1
      GO TO 113
C
C        MAKE PERMANENT STORAGE OF L AND D, A AND B.
C
  119 IM1=I-1
      N=JA-1
      DO 120 NN=1,NTV
      N=N+1
      LNTH(IM1,N)=LTMP(NN)
      LTMP(NN)=LLNTH(NN)
      A(I-1,N)=ATMP(NN)
      ATMP(NN)=AT(NN)
  120 CONTINUE
      JSIG=JSIGT-1
      DO 121 NN=1,NTV1
      JSIG=JSIG+1
```

```
      DNTH(I,JSIG)=DTMP(NN)
      DTMP(NN)=DLNTH(NN)
      B(I,JSIG)=BTMP(NN)
      BTMP(NN)=BT(NN)
  121 CONTINUE
      GO TO 135
C
C        STORE L**2, D**2 FOR I=1, ALL J, FOR DEFINITION OF A AND B
C             AFTER CIRCULATIONS ARE COMPUTED
C
  122 JNTV=JA+NTV1
      NN=0
      DO 124 JJ=JA,JNTV
      NN=NN+1
      LSQ(JJ)=LLNTH(NN)
  124 CONTINUE
      JSIGI=JSIG-NTV1+1
      NN=0
      DO 126 JJ=JSIGI,JSIG
      NN=NN+1
      DSQ(JJ)=DLNTH(NN)
  126 CONTINUE
  135 CONTINUE
      IF (KX.GT.1) RETURN
      N=JSIGT-1
      DO 139 NN=1,NTV1
      N=N+1
      DNTH(NW,N)=DTMP(NN)
      B(NW,N)=BTMP(NN)
  139 CONTINUE
      N=JA-1
      DO 141 NN=1,NTV
      N=N+1
      LNTH(NW-1,N)=LTMP(NN)
      A(NW-1,N)=ATMP(NN)
      IF (NW.EQ.2.AND.NAS.EQ.1) A(NW,N)=CHORD(M)
  141 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBII (NPER,NA,INDX,NR,JAC,NIB,NROT,NTV1,VZ,JA,J,JB)
C
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION VLL(16)
      DIMENSION VZ(05,20)
C
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBV
      COMMON /MODWK1/ GAMMA (54,16),R (1),A (05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),X (54,16),Y (54,16),Z (54,16)
      COMMON /MODWK3/ AFM(4,10),BFM(4,4)
      COMMON /VLNTHS/ NALIM,VLL
      COMMON /WK2C/ SIGBL(144)
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
C
      EL=VLL(JA)
      TWOEL=2.*EL
      RSQ=R(1)*R(1)
      NSW=-1
      NWSTM1=NWSTRE-1
      I=NANRM-1
   70 I=I+1
      IP1=I+1
      XC=X(IP1,JA)
      YC=Y(IP1,JA)
      ZC=Z(IP1,JA)
      WORK1=XA-XC
      WORK2=YA-YC
      WORK3=ZA-ZC
      RT=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
      IF(RT.GT.TWOEL) GO TO 86
      IF (RSQ.LT.EL) GO TO 84
      IF (RT.LT.EL) GO TO 84
      NSW=0
      GO TO 134
   86 I=I+NALIM
      NSW=1
      GO TO 134
   84 IF (NSW.LT.1) GO TO 87
      I=I-NALIM
      NSW=-1
      GO TO 135
C
C        COMPUTE SQUARE ROOT OF RT AT THE APPROPRIATE STEP.
   87 XB=X(I,JA)
      YB=Y(I,JA)
      ZB=Z(I,JA)
```

```
      NSW=-1
      NN=1
      N=1
      SIGN=1.
      JJ=JA
      IF(NPER.NE.4) GO TO 90
85    IGN = 0
      IF(INDX-I.GE.NA) IGN = -1
      IF(INDX-I.LT.0) IGN = 1
      INDX = INDX + NA * IGN
      IF ( IGN.NE.0) GO TO 85
      INDXL= (INDX-I) * NR + ((JAC-1)/NIB) * (NR/NROT)
C
   90 RS=SQRT(RT)
      GO TO 100
C
C        COMPUTE NV*G, TRAILING VORTEX CONTRIBUTION, OR ETA*H, SHED
C             VORTEX CONTRIBUTION, L2 OR D2, AND ADD TO PREVIOUS CON-
C             TRIBUTIONS FROM QUADRILATERAL FOR GAMMA(I,JA).
C
  100 RPR=RS+R(NN)
      DORL=(XB-XC)**2+(YB-YC)**2+(ZB-ZC)**2
      ASSIGN 94 TO IORGT
      WORK1=R(NN)
      VTEST=RS*RS+WORK1*WORK1-DORL
      IF (VTEST.GT.0) GO TO 101
      WORK2=RS-WORK1
      WORK2=WORK2*WORK2
      WORK3=RS+WORK1
      WORK3=WORK3*WORK3
      VTEST=(WORK3-DORL )*(DORL -WORK2)/(4.*DORL )
   92 IF (SIGN.GT.0) WORK1=A(NANRM,JJ)
   93 WORK1=WORK1*WORK1
      IF (VTEST.GT.WORK1) GO TO 101
      HORG=0.
      WORK5=SQRT(DORL)*WORK1
      IF(WORK5.NE.0.) HORG=1./WORK5
      ASSIGN 95 TO IORGT
      GO TO 103
 101  HORG=0.
      WORK5=R(NN)*RS*(RPR*RPR-DORL)
      IF(WORK5.NE.0.) HORG=SIGN*RPR/WORK5
  103 EORNZ=(XA-XC)*(YC-YB)-(YA-YC)*(XC-XB)
      EHNGZ=EORNZ*HORG
      GO TO IORGT, (94,95)
   95 SIGM=1.
      IF (EHNGZ.LT.0.) SIGM=-1.
      WORK1=GAMMA(I,JA)
      WORK2=EHNGZ*WORK1
      IF (ABS(WORK2).GT.1.) EHNGZ=SIGM/WORK1
```

```
   94 SGMAZ=EHNGZ
  102 R(NN)=RS
C
C         GAMMA(1,J)S ARE UNKNOWN.  IF I=1, STORE SIGMA(J,JSIG).
C         THIS IS SIGMA(1,PSII,J,JSIG)
C
C
C         GAMMA(1,J) ARE KNOWN FOR I.GT.1.    COMPUTE VZ(1,J).
C
  108 VZ(1,J)=VZ(1,J)+(SGMAZ              )*GAMMA(I,JA)
      IF( NPER.NE.4) GO TO 111
      SGMBL = SGMAZ
C
C         CONVENIENT LOCATION TO COMPUTE SIGMA(J,PSII,I,JJ)
C
      DO 107 JZ=1,NTV1
      MODINX=INDXL+JZ
  107 SIGBL(MODINX)=-SGMBL*BFM(JB,JZ)+SIGBL(MODINX)
C
C         IF THE ROW IS NOT COMPLETED, STORE SGMAZ AND CONTINUE WITH VZ
C              COMPUTATION.
C
  111 R(NN)=RS
  134 RSQ=RT
  135 IF (I.LT.NANRM) GO TO 140
      IF (I.LT.NWSTM1) GO TO 70
  140 RETURN
      END
```

```
      SUBROUTINE CONVG
C
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
```

```
      DIMENSION PSI(2)
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
C
      COMMON /ITRG/ ITRGX
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA2/ XSIMQ,ZAP,  NGJJ,M1,DETERM
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /CONVGA/ EPSG,NWKRQ
      COMMON /CONVGB/ SGMA1,INDXG
      COMMON /CONVGC/ GAMMAG
      COMMON /WK2A/ MSET,GAMMK,INDX,JAC,INDXL,SGMAZ,SIGMZ,SGMBL
```

```
      COMMON /WK2C/ SIGBL
      COMMON /OUTIN/ IN,OUT
      COMMON /SUBIE/ NAS
      COMMON /STEPXA/ WKPT,WW,IOUT,NOTTP1,KAT,NBC
C
 9876 FORMAT (1H0,/8(1X,G14.7))
  950 FORMAT (13H ITG DIVERGES,2I5,2E14.7)
  951 FORMAT (I5,E12.4,31H GAMMAS HAVE CONVERGED   MSET= ,I10,/)
C
C
C
C        COMPUTE NEW GAMMA(1,J) USING ITERATION SCHEME ON AN EQ OF THE
C             FORM G=GL+GV+C*SUM((SIG*G))
      NGJJ=NGJR
      N=1
      M1=0
      ITR=0
  142 GDI=0.
      G=0.
      ITR=ITR+1
      DO 145 J=1,NGJR
      GDIF=GAMMA(1,J)
      XK=0.
      DO 144 K=1,JSIG
  144 XK=SGMA1(J,K)*GAMMA(1,K)+XK
      GAMMA(1,J)=(GAMMK(1,J)+XK-SGMA1(J,J)*GAMMA(1,J))/(1.-SGMA1(J,J))
      GDI=(GDIF-GAMMA(1,J))**2+GDI
  145 G=G+GAMMA(1,J)**2
      GTEST=GDI /G
      IF (GTEST.LE.EPSG) GO TO 150
C
C        IF GAMMAS CONVERGE, CONTINUE, OTHERWISE ITERATE AGAIN UNLESS
C             ITR EXCEEDS UPPERLIMIT.
C
  146 IF (ITR.LT.ITRGX) GO TO 142
  148 WRITE (OUT,950) ITR,ITRGX,GTEST,EPSG
      IF (M1.EQ.1) STOP
      DO 152 J=1,NGJR
      DO 152 K=1,JSIG
  152 XSIMQ(J,K)=-SGMA1(J,K)
      DO 153 J=1,NGJR
      ZAP(J)=GAMMK(1,J)
  153 XSIMQ(J,J)=1.+XSIMQ(J,J)
      ITR=0
      M1=1
      CALL MATINV
      DO 147 J=1,NGJR
  147 GAMMA(1,J)=ZAP(J)
      GO TO 142
C
```

```
C         GAMMAS HAVE CONVERGED
C
  150 WRITE (OUT,951) ITR, GTEST,MSET
      WRITE (6,9876)  XK,((GAMMA(IX,JX),IX=1,N),JX=1,NGJR)
C
      IF (NPER.EQ.4) GO TO 157
      RETURN
  157 DO 164 J=1,NGJR
      M=INDXG(J)
  164 GAMMAG(M)=GAMMA(1,J)
      IF (NAS.EQ.WW+(NA/NIB)-1) WRITE (NOTTP1) (GAMMAG(J),J=1,NANR)
      IF (NAS .EQ. WW+(NA/NIB)-1)  WRITE (6,9876) (GAMMAG(J),J=1,NANR)
      RETURN
      END
```

```
      SUBROUTINE AL1BD2
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
C
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /ALIBDA/ ABK
      COMMON /SUBIB/ KX,LSQ,DSQ,DTMP,LTMP,ATMP,BTMP
C
C        PERFORM FLOW PERIODICITY CHECK.  IF NOT PERIODIC CHECK TIME
C             LIMIT (GO TO 446).  IF PERIODIC CHOOSE ONE OR MORE OF (A)
C             COMPUTE SIGS FOR BLADE LOADS ON BASIS OF SMALLER
C             SPACING, (B) COMPUTE FLOW FIELD, NOT AT VORTEX END
```

```
C              POINTS, AND (C) OTHER.
C
C
C
C         COMPUTE A(1,J), B(2,J), L(1,J), AND D(2,J)
C
      JJ=0
      JSIG=0
      NN=0
  151 JJ=JJ+1
      RT=LSQ(JJ)
      RS=SQRT(RT)
      LNTH(1,JJ)=RS
      NN=NN+1
      IF (NN.GT.1) GO TO 158
      JSIG=JSIG+1
  154 RT=ABS(GAMMA(1,JSIG))
      IF (JJ.LT.NIBRV) GO TO 155
      RS=SQRT(RT)
      GO TO 162
  155 RS=SQRT(RT)
C
C         VORTEX CORE RADIUS IS CONSTANT TIMES CIRCULATION**.5
C
  156 A(1,JJ)=RS*ABK
      GO TO 151
  158 IF (NN-NTV) 160,159,162
  159 NN=0
      GO TO 154
  160 JSIG=JSIG+1
      RT=ABS(GAMMA(1,JSIG)-GAMMA(1,JSIG-1))
      RS=SQRT(RT)
      GO TO 156
  162 A(1,JJ)=RS*ABK
      JJ=0
  163 JJ=JJ+1
      RT=DSQ(JJ)
      RS=SQRT(RT)
      DNTH(2,JJ)=RS
      RT=ABS(GAMMA(1,JJ)-GAMMA(2,JJ))
      RS=SQRT(RT)
      B(2,JJ)=RS*ABK
      IF (JJ.LT.NGJR) GO TO 163
      RETURN
      END
```

```
      SUBROUTINE APXLD
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
C
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QS.
      COMMON /APXLDA/ RBAR
      COMMON /APXLDB/ LOADN
      COMMON /WK1A/ PSIR,DPSIK,PSIK,DELTA,RREF,MUCDS,MUSDS,THTAX,THTAY
      COMMON /WK1C/ PSI,CCLA,DIR
C
C        DEFINE APPROXIMATE BLADE LOADS
C        NONDIMENSIONAL FORM=(RHO*U*GAMMA)/(RHO*OM*ON*R*R*R)
C
  170 DO 180 M=1,NROT
      DO 180 K=1,NIB
      PSIK=PSIR(M)+(K-1)*DPSIK
```

```
      MUSPK=MU*SIN(PSI(M)+PSIK)
      JJ=0
      DO 180 J=1,NTV1
      JJ=JJ+1
      LOADN(JJ)=(RBAR(JJ)+MUSPK*DIR(M))*GAMMA(1,JJ)
  180 CONTINUE
C
C        LOADN IS NON-DIMENSIONAL
C        LOADD(JJ)=LOADN(JJ)*DFLOD
C        WRITE EITHER LOADN OR LOADD AFTER COMPUTATION, BUT DO NOT USE
C             BOTH.  LOADN(JJ) AND VZ(J) COULD BE EQUIVALENCED.
C
      RETURN
      END
```

```
      SUBROUTINE WK3
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VXX(1,1)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
C
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /CONVGA/ EPSG,NWKRQ
      COMMON /WKCONT/ NWKPD
      COMMON /STEPXA/ WKPT,WW,IOUT,NOTTP1,KAT,NBC
      COMMON /WK4A/ VXX
      COMMON /OUTDI/ NNTV,NEXPWK,NSIGRW,NMODR,NMODC
      COMMON /MODWK1/ GAMMAM(54,16),RM(1),AM(05,16),VXM(54,16),
```

```
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /OUTDII/ NWKCLM
C
      DATA INX1,INX2,INX3,INX4,INX5,INX6,INX7/
     14H   X,4H   Y,4H   Z,4H  VX,4H  VY,4H  VZ,4HSIGA/
      DATA NWKX,NWKY,NWKZ/3HWKX,3HWKY,3HWKZ/
C
C        LOADN(JJ) COULD BE LOADD(JJ)
C
      IF (NWKRQ.EQ.1.AND.NPER.EQ.4) GO TO 192
      GO TO 189
  192 NLP1=1
      NLP2=1
      NLP3=NWKCL
      NLP4=NWKRW
      CALL T3A
      DO 186 JX=1,NWKCL
      DO 186 IX=1,NWKRW
      IF (IX.GT.NWKLST.AND.JX.EQ.NWKCL) GO TO 191
      VXX(IX,JX)=-VZ(IX,JX)*COSB3+VX(IX,JX)*SINB3
  186 VI(IX,JX)=VXX(IX,JX)+VI(IX,JX)
  191 CONTINUE
      CALL MPRECT (NWKY,VXX,NWKRW,NWKCL,NWKRW,NWKCLM)
      NAS1=NAS1+1
      IF (NAS1.LT.NIBNA) GO TO 188
C
C        COMPUTE WAKE FLOWS
C
      VONA=NIB/(VOOMR*NA)
      DO 187 JX=1,NWKCL
      DO 187 IX=1,NWKRW
      IF (IX.GT.NWKLST.AND.JX.EQ.NWKCL) GO TO 190
  187 VI(IX,JX)=VONA*VI(IX,JX)
  190 CONTINUE
      CALL MPRECT (NWKX,VI,NWKRW,NWKCL,NWKRW,NWKCLM)
      KAT=1
  188 NLP1=NROT
      NLP2=NIB
      NLP3=NTV
C
C
C
C        SET WAKE AZIMUTHAL INDEX LIMIT
C
  189 NLP4=NW
      NPER=1
      CALL T3A
      NPER=NWKPD
      IF (NWKPD.NE.4) GO TO 377
```

```
      CALL MPRECT (INX4,VX,NW,NIBRV,NEXPWK,NNTV)
      CALL MPRECT (INX5,VY,NW,NIBRV,NEXPWK,NNTV)
      CALL MPRECT (INX6,VZ,NW,NIBRV,NEXPWK,NNTV)
      CALL MPRECT (INX1   ,X,NW,NIBRV,NEXPWK,NNTV)
      CALL MPRECT (INX2   ,Y,NW,NIBRV,NEXPWK,NNTV)
      CALL MPRECT (INX3   ,Z,NW,NIBRV,NEXPWK,NNTV)
      CALL MPRECT (INX4,VXM,NWSTRE,NIBRVM,NMODR,NMODC)
      CALL MPRECT (INX5,VYM,NWSTRE,NIBRVM,NMODR,NMODC)
      CALL MPRECT (INX6,VZM,NWSTRE,NIBRVM,NMODR,NMODC)
      CALL MPRECT (INX1,XM,NWSTRE,NIBRVM,NMODR,NMODC)
      CALL MPRECT (INX2,YM,NWSTRE,NIBRVM,NMODR,NMODC)
      CALL MPRECT (INX3,ZM,NWSTRE,NIBRVM,NMODR,NMODC)
  377 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE T3A
      INTEGER Z2
      INTEGER OUT,WKPT,CNTR
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LLTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMK(1,016)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION SGMA1(016,016)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
```

```
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZROT(02)
C
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /MODWK1/ GAMMAM(54,16),RM(1),AM(05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /SUBIE/ NAS
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /OVT3A/ Z2,II
      COMMON /OVT3B/ IROW,JAKM,K,L
      COMMON /OVT3F/ JPASS
C
C
C
C        COMPUTE INDUCED VELOCITIES AT ALL VORTEX ELEMENT END POINTS,
C             AND INCLUDING BLADE VORTEX POINTS. INCLUDE SELF-INDUCED
C             EFFECTS DUE TO BLADE AND NEIGHBORING VORTICES (BASED ON
C             APPROXIMATE CURVATURE FROM AN ARC DETERMINED FROM 3-POINT
C             CIRCULAR ARC CURVE FITTING.
C
C
C        INITIALIZE CONTROL CONSTANTS FOR WAKE INDUCED VELOCITY
C             CALCULATIONS.
C
      N1=NLP1
      N2=NLP2
      N3=NLP3
      N4=NLP4
      N5=2
      N6=1
      IF (NPER.EQ.4) N5=1
      DO 600 Z2=1,N5
```

```
      IF (NAS.LE.NANRM.AND.Z2.EQ.2) GO TO 600
  200 JJ=0
      J=0
C
C        J AND K HAVE REDEFINED FOR K>2
C
      IF (NPER.EQ.4) GO TO 201
      IF (Z2.EQ.1) GO TO 201
      N3=NTVM
      N4=NWSTRE-1
      IF (NWSTRE.LT.NWR) N4=NWSTRE
      N6=NANRM+1
  201 DO 500 M=1,N1
      DO 500 K=1,N2
      JAKM=(K-1)*NTV+1+NTV*NIB*(M-1)
      DO 500 L=1,N3
      J=J+1
      DO 500 II=N6,N4
      IF (Z2.EQ.2) GO TO 212
      IF (NPER.EQ.4) GO TO 211
      XA=X(II,J)
      YA=Y(II,J)
      ZA=Z(II,J)
      GO TO 220
  211 IF (II.GT.NWKLST.AND.J.EQ.NWKCL) RETURN
      JAKM=0
      XA=WKX(II,J)
      YA=WKY(II,J)
      ZA=WKZ(II,J)
      GO TO 220
  212 XA=XM(II,J)
      YA=YM(II,J)
      ZA=ZM(II,J)
      VXM(II,J)=0.
      VYM(II,J)=0.
      VZM(II,J)=0.
      GO TO 221
C
C        POINT A IS POINT AT WHICH VELOCITIES ARE TO BE COMPUTED.
C
  220 VX(II,J)=0.
      VY(II,J)=0.
      VZ(II,J)=0.
C        INITIALIZE VELOCITY COMPONENTS, INDICES, ETC.
  221 JSIG=0
      IROW=NPER
      JPASS=J
      CALL T3AB
      IF (NAS.GT.NANRM) CALL T3ASP(VX,VY,VZ)
      IF (Z2.EQ.1) CALL ADVXYZ (VX(II,J),VY(II,J),VZ(II,J),TWOPI)
```

```
      IF (Z2.EQ.2) CALL ADVXYZ(VXM(II,J),VYM(II,J),VZM(II,J),TWOPI)
  500 CONTINUE
  600 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ADVXYZ(VX,VY,VZ,TWOPI)
      VX=VX/TWOPI
      VY=VY/TWOPI
      VZ=VZ/TWOPI
      WORK1=ABS(VX)
      WORK2=ABS(VY)
      WORK3=ABS(VZ)
      WORK4=AMAX1(WORK1,WORK2,WORK3)
      IF (WORK4.LE..1) RETURN
      VX=(VX/WORK4)*.1
      VY=(VY/WORK4)*.1
      VZ=(VZ/WORK4)*.1
      RETURN
      END
```

```
      SUBROUTINE T3AB
      INTEGER Z2
      INTEGER OUT,WKPT,CNTR
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMK(1,016)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION SGMA1(016,016)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
```

```
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZROT(02)
C
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /MODWK1/ GAMMAM(54,16),RM(1),AM(05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /VLIMIT/ VLIM(5),VMLIM(16)
      COMMON /OVT3A/ Z2,II
      COMMON /OVT3B/ IROW,JAKM,K,L
C
      QSX=0.
      QSY=0.
      QSZ=0.
      DO 400 JA=1,NIBRV,NTV
      QX=0.
      QY=0.
      QZ=0.
      JL=JA
      JSIGT=1+((JA-1)*NTV1)/NTV
C        IROW CONTROLS BRANCHING TO SPECIAL COMUTATIONS REQUIRED FOR
C             SELF-INDUCED VELOCITY CALCULATIONS, AND AVOIDANCE OF
C             CALCULATIONS BY STANDARD EQUATIONS.  IROW=1, POINT A IS ON
C             CURRENT ROW, SELF-INDUCED VELOCITIES ARE LINEAR, IROW=2,
C             POINT A IS ON NEXT ROW, IROW=3, POINT A IS NOT ON CURRENT
C             BLADES WAKE, BUT HAS NOT BEEN ACCOUNTED FOR, IROW=4,
C             POINT A HAS BEEN ACCOUNTED FOR, NO FURTHER CHECKS NEED BE
C             MADE UNTIL POINT A IS REDEFINED.
      IF (IROW.GT.3) GO TO 213
  212 IROW=2
  213 CONTINUE
      DO 390 I=1,NWM1
      JSIG=JSIGT
```

```
      SGMAX=0.
      SGMAY=0.
      SGMAZ=0.
      NN=0
      JL=JA
C        IF POINT A IS NOT ON CURRENT BLADES WAKE DO STANDARD
C             CALCULATION, OTHERWIZE TEST FOR NECESSITY OF SELF-INDUCED
C             VELOCITY COMPUTATIONS.
C        I=1, START ON NEW BLADES WAKE
  214 IF (I.GT.1) GO TO 280
C        CHECK TO SEE IF POINT A IS ON CURRENT BLADES WAKE
  215 IF (JA-JAKM) 218,216,219
C        II=I=1 AUTOMATICALLY REQUIRES SPECIALIZED SELF-INDUCED VELOCITY
C             CALCULATIONS.
  216 IF (II.LE.1) GO TO 222
  217 IROW=2
      GO TO 270
  218 IROW=3
      GO TO 270
  219 IROW=4
      GO TO 270
C        II=I=1, AND A ARE ON CURRENT BLADES WAKE
C        COMPUTE R(JL)S FOR FUTURE USE
  222 JL=J
C
C        COMPUTE ADDITIONAL SELF-INDUCED QSZ COMPONENT DUE TO BLADE OR
C             COMPUTE SHED VORTEX COMPONENT AT END OF WAKE.
C        STATEMENTS TO BE ADDED
C
      CALL TEST5
      DO 224 JX=1,NTV
  224 R(JX)=ABS(RCAP(M,L)-RCAP(M,JX))
      IF (NW.LE.2) GO TO 260
C        COMPUTE SELF-INDUCED VELOCITY FROM TRAILING VORTEX ONLY CON-
C             TRIBUTION.
      XB=X(2,JL)
      YB=Y(2,JL)
      ZB=Z(2,JL)
      JSIG=JSIGT+J-JA-1
      IF (JSIG.EQ.JSIGT-1) JSIG=JSIG+1
      XC=X(3,JL)
      YC=Y(3,JL)
      ZC=Z(3,JL)
      IP1=I+1
      SIGN=1.
      GO TO 231
C        II=NW AND A ARE ON CURRENT BLADES WAKE
  229 IP1=I-1
      IF (IP1.EQ.0) GO TO 321
      XC=X(IP1,JL)
```

```
      YC=Y(IP1,JL)
      ZC=Z(IP1,JL)
      SIGN=-1.
  231 MX=(YA-YB)*(ZB-ZC)-(YB-YC)*(ZA-ZB)
      MY=(ZA-ZB)*(XB-XC)-(ZB-ZC)*(XA-XB)
      MZ=(XA-XB)*(YB-YC)-(XB-XC)*(YA-YB)
      RT=MX*MX+MY*MY+MZ*MZ
      IF (RT.EQ.0.) RT=1.
      RS=SQRT(RT)
      MXYZ=RS
      DELSQ=(XA-XC)**2+(YA-YC)**2+(ZA-ZC)**2
      WORK1=LNTH(I,JL)*LNTH(IP1,JL)
      WORK1=WORK1*WORK1
      WORK2=LNTH(I,JL)*LNTH(I,JL)
      WORK3=LNTH(IP1,JL)*LNTH(IP1,JL)
      WORK2=WORK2+WORK3-DELSQ
      WORK2=WORK2*WORK2
      RT=4.*WORK1-WORK2
      IF (RT.LE.0.) GO TO 247
      RT=DELSQ/RT
      RS=SQRT(RT)
      RSCRP=2.*LNTH(I,JL)*LNTH(IP1,JL)*RS
      RT=LNTH(I,JL)*LNTH(I,JL)
      RT=RSCRP*RSCRP-RT
      IF (RT.LE.0.) GO TO 247
      RS=SQRT(RT)
      WORK1=LNTH(I,JL)*LNTH(I,JL)
      WORK2=LNTH(IP1,JL)*LNTH(IP1,JL)
C        CHOOSE DEFINITION OF FSMAL ACCORDING TO L(I,J)**2.GE.DELSQ
C             +L(IP1,J)**2
      IF (WORK1-DELSQ-WORK2.GT.0) GO TO 238
  236 FSMAL=(RSCRP-RS)/LNTH(I,JL)
      GO TO 240
  238 FSMAL=(RSCRP+RS)/LNTH(I,JL)
  240 RT=4.*FSMAL*RSCRP/A(I,JL)
C        CHOOSE AND USE PROPER CIRCULATION VALUE
      IF (JL-JA) 243,244,242
  242 IF (JL-JA-NTV1) 246,245,243
  243 STOP
  244 FSCRP=GAMMA(I,JSIG)*(ALOG(RT)+.25)/(2.*RSCRP*MXYZ)
      GO TO 248
  245 FSCRP=-GAMMA(I,JSIG)*(ALOG(RT)+.25)/(2.*RSCRP*MXYZ)
      GO TO 248
  246 FSCRP=(GAMMA(I,JSIG+1)-GAMMA(I,JSIG))*(ALOG(RT)+.25)/(2.*RSCRP*
     1MXYZ)
      GO TO 248
  247 FSCRP=0.
  248 FSCRP=FSCRP*SIGN
  249 IF (II.GE.NW) GO TO 322
      QSX=MX*FSCRP
```

```
      QSY=MY*FSCRP
      QSZ=MZ*FSCRP
  263 CONTINUE
  260 DO 262 NX=1,NTV1
C        INITIALIZE SIGMS
      SIGMX(NX)=0.
      SIGMY(NX)=0.
  262 SIGMZ(NX)=0.
      IROW=1
C        AVOID COMPUTATION EHNGZ AT JL=J
      IF (JL.EQ.JA) GO TO 265
      JSIG=JSIGT
      JL=JA
      GO TO 360
  265 R(1)=LNTH(1,JL)
      NN=1
      IP1=I+1
      GO TO 357
  270 JL=JA
      NN=0
  272 XB=X(1,JA)
      YB=Y(1,JA)
      ZB=Z(1,JA)
      RT=(XB-XA)**2+(YB-YA)**2+(ZB-ZA)**2
      RS=SQRT(RT)
      R(1)=RS
  275 JL=JL+1
      NN=NN+1
      NX=JSIGT+NN-1
      XC=XB
      YC=YB
      ZC=ZB
      XB=X(1,JL)
      YB=Y(1,JL)
      ZB=Z(1,JL)
      RT=(XB-XA)**2+(YB-YA)**2+(ZB-ZA)**2
      RS=SQRT(RT)
      RPR=RS+R(NN)
      DORL=(XB-XC)**2+(YB-YC)**2+(ZB-ZC)**2
      WORK1=R(NN)
      VTEST=RS*RS+WORK1*WORK1-DORL
      IF (VTEST.GT.0) GO TO 276
      WORK2=RS-WORK1
      WORK2=WORK2*WORK2
      WORK3=RS+WORK1
      WORK3=WORK3*WORK3
      VTEST=(WORK3-DORL )*(DORL -WORK2)/(4.*DORL )
      WORK1=CHORD(M)
      WORK1=WORK1*WORK1
      IF (VTEST.GT.WORK1) GO TO 276
```

```
      HORG=0.
      WORK5=SQRT(DORL)*WORK1
      IF(WORK5.NE.0.) HORG=1./WORK5
      GO TO 277
  276 HORG=0.
      WORK5=R(NN)*RS*(RPR*RPR-DORL)
      IF(WORK5.NE.0.) HORG=RPR/WORK5
  277 EORNX=(XA-XC)*(YC-YB)-(YA-YC)*(XC-XB)
      EORNY=(ZA-ZC)*(XC-XB)-(XA-XC)*(ZC-ZB)
      EORNZ=(YA-YC)*(ZC-ZB)-(ZA-ZC)*(YC-YB)
  288 WORK1=GAMMA(1,NX)
      WORK2=ABS(EORNX*WORK1*HORG)
      WORK3=ABS(EORNY*WORK1*HORG)
      WORK4=ABS(EORNZ*WORK1*HORG)
      WORK1=AMAX1(WORK2,WORK3,WORK4)
      IF (WORK1.LE.VLIM(NN)) GO TO 287
      SIGMX(NN)=EORNX/WORK1*HORG*VLIM(NN)
      SIGMY(NN)=EORNY/WORK1*HORG*VLIM(NN)
      SIGMZ(NN)=EORNZ/WORK1*HORG*VLIM(NN)
      GO TO 286
  287 SIGMX(NN)=EORNX*HORG
      SIGMY(NN)=EORNY*HORG
      SIGMZ(NN)=EORNZ*HORG
  286 R(NN+1)=RS
      IF (NN.LT.NTV1) GO TO 275
C        CONTINUE COMPUTING BLADE CONTRIBUTIONS UNTIL BLADE IS COMPLETED
C          THEN GO TO NEXT ROW.
  278 JL=JA
      IF (IROW-2) 350,280,360
C        DO SPECIAL CALCULATIONS ONLY IF POINT A IS ON CURRENT BLADES
C          WAKE.
  280 IF(JA-JAKM) 281,282,359
  281 IROW=3
      GO TO 360
C        IF POINT A IS NOT ON OR JUST DOWN THE WAKE FROM B OR C, DO
C          STANDARD CALCULATION.
  282 IF (II-I-1) 350,283,360
C        IF POINT A IS NOT BEHIND POINT JL, DO STANDARD CALCULATION.
  283 IF (JL.NE.J) GO TO 360
C        COMPUTE SELF-INDUCED VELOCITIES FOR I.GT.1
C        COMPUTE SELF-INDUCED QS FOR TRAILING VORTICES
  284 JK=0
      N=JL
      XB=X(I,JL)
      YB=Y(I,JL)
      ZB=Z(I,JL)
      IF (II.GE.NW) GO TO 229
      LTMP(1)=LNTH(I,JL)
      LTMP(2)=LNTH(II,JL)
      IP1=I+2
```

```
      XC=X(IP1,JL)
      YC=Y(IP1,JL)
      ZC=Z(IP1,JL)
      IP1=I+1
      ATMP(1)=A(I,JL)
  285 MX=(YB-YA)*(ZA-ZC)-(YA-YC)*(ZB-ZA)
      MY=(ZB-ZA)*(XA-XC)-(ZA-ZC)*(XB-XA)
      MZ=(XB-XA)*(YA-YC)-(XA-XC)*(YB-YA)
      RT=MX*MX+MY*MY+MZ*MZ
      IF (RT.EQ.0.) RT=1.
      RS=SQRT(RT)
      MXYZ=RS
      DELSQ=(XB-XC)**2+(YB-YC)**2+(ZB-ZC)**2
      WORK3=    (4.*(LTMP(1)*LTMP(2))**2-(LTMP(1)**2+LTMP(2)**2-DELSQ)**
     12)
      IF (WORK3.LE.0) WORK3=5.4E-70
      RT=DELSQ/WORK3
      RS=SQRT(RT)
      RSCRP=2.*LTMP(1)*LTMP(2)*RS
  289 RT=RSCRP**2-LTMP(1)**2
      JK=JK+1
      IF (RT.LT.0.) RT=0.
      RS=SQRT(RT)
C        COMPUTE FSMAL DEPENDING ON L(1)**2.GE.DELSQ)L(2)**2.
      IF (LTMP(1)**2-DELSQ-LTMP(2)**2.GT.0) GO TO 294
  292 FSMAL=(RSCRP-RS)/LTMP(1)
      GO TO 296
  294 FSMAL=(RSCRP+RS)/LTMP(1)
  296 RT=4.*FSMAL*RSCRP/ATMP(1)
C        COMPUTE TRAILED (JK=1,2) OF SHED (JK=3,4) CONTRIBUTION TO
C          SELF-INDUCED VELOCITY.
      GO TO (297,310,346,348), JK
  297 IF (JL-JA.LE.0) GO TO 300
  298 IF (JL-JA-NTV1) 304,302,343
  300 IF (RT.EQ.0.) GO TO 307
      FSCRP=GAMMA(I,JSIG)*(ALOG(RT)+.25)
      GO TO 308
  302 IF (RT.EQ.0.) GO TO 307
      FSCRP=-GAMMA(I,JSIG)*(ALOG(RT)+.25)
      GO TO 308
  304 IF (RT.EQ.0.) GO TO 307
      FSCRP=(GAMMA(I,JSIG+1)-GAMMA(I,JSIG))*(ALOG(RT)+.25)
      GO TO 308
C        COMPUTE CONTRIBUTION FROM NEXT TRAILED VORTEX ELEMENT.
  307 FSCRP=0.
  308 ATMP(1)=A(II,JL)
      LTMP(3)=LTMP(1)
      LTMP(1)=LTMP(2)
      LTMP(2)=LTMP(3)
      GO TO 289
```

```
  310 IF (JL.LE.JA) GO TO 316
  314 IF (JL-JA-NTV1) 320,318,343
  316 IF (RT.EQ.0.) GO TO 322
      FSCRP=(FSCRP+GAMMA(II,JSIG)*(ALOG(RT)+.25))/(2.*RSCRP*MXYZ)
      GO TO 322
  318 IF (RT.EQ.0.) GO TO 322
      FSCRP=(FSCRP-GAMMA(II,JSIG)*(ALOG(RT)+.25))/(2.*RSCRP*MXYZ)
      GO TO 322
  320 IF (RT.EQ.0.) GO TO 322
      FSCRP=(FSCRP+(GAMMA(II,JSIG+1)-GAMMA(II,JSIG))*(ALOG(RT)+.25))/(
     12.*RSCRP*MXYZ)
C         DEFINE TRAILED VORTEX SELF-INDUCED VELOCITY COMPONENT.
C         CHECK SIGN OF FSCRP
  322 QSX=MX*FSCRP
      QSY=MY*FSCRP
      QSZ=MZ*FSCRP
  321 IF (JL-JA) 323,323,340
C         COMPUTE NG WITH CNLY ONE VORTEX INCLUDED, JSIG
C         COMPUTE FIRST CONTRIBUTION FROM SHED VORTICITY.
  323 JLM1=JL
      NN=1
      JL=JL+1
      JSIG1=JSIG+1
      JLP1=JL+1
      JK=0
      RT=1.
  324 ATMP(1)=   B(II,JSIG)
      XB=X(II,JL)
      YB=Y(II,JL)
      ZB=Z(II,JL)
      XC=X(II,JLP1)
      YC=Y(II,JLP1)
      ZC=Z(II,JLP1)
      LTMP(1)=DNTH(II,JSIG)
      LTMP(2)=DNTH(II,JSIG1)
  325 MX=((YA-YB)*(ZB-ZC)-(YB-YC)*(ZA-ZB))*RT
      MY=((ZA-ZB)*(XB-XC)-(ZB-ZC)*(XA-XB))*RT
      MZ=((XA-XB)*(YB-YC)-(XB-XC)*(YA-YB))*RT
      RT=MX*MX+MY*MY+MZ*MZ
      IF (RT.EQ.0.) RT=1.
      RS=SQRT(RT)
  326 MXYZ=RS
      DELSQ=(XA-XC)**2+(YA-YC)**2+(ZA-ZC)**2
      RT=      (4.*(LTMP(1)*LTMP(2))**2-(LTMP(1)**2+LTMP(2)**2-
     1DELSQ)**2)
      IF (RT.LE.0) GO TO 337
      RT=DELSQ/RT
      RS=SQRT(RT)
      RSCRP=2.*LTMP(1)*LTMP(2)*RS
      RT=RSCRP**2-LTMP(1)**2
```

```
      IF (RT.LE.0.) GO TO 337
      RS=SQRT(RT)
      IF (LTMP(1)**2-DELSQ-LTMP(2)**2.GT.0) GO TO 334
  332 FSMAL=(RSCRP-RS)/LTMP(1)
      GO TO 336
  334 FSMAL=(RSCRP+RS)/LTMP(2)
  336 RT=4.*FSMAL*RSCRP/ATMP(1)
      IF (II.GE.NW) GO TO 338
      FSMAL=GAMMA(II,JSIG)-GAMMA(I,JSIG)
      GO TO 339
  338 FSMAL=-GAMMA(I,JSIG)
  339 IF (RT.LE.0.) GO TO 337
      FSCRP=FSMAL*(ALOG(RT)+.25)/(2.*RSCRP*MXYZ)
      QSX=QSX+MX*FSCRP
      QSY=QSY+MY*FSCRP
      QSZ=QSZ+MZ*FSCRP
  337 IF (JL.EQ.JA+1) GO TO 343
      JL=JL+1
  340 IF (JL-JA-NTV1.LT.0) GO TO 344
  341 JL=JA+NTV1-1
      JLP1=JL-1
      RT=-1.
      JK=JK+1
      NN=NTV1
      JSIG1=JSIG-1
      GO TO (324,343,324,343), JK
C        CHECK ON VALIDITY OF THIS TRANSFER WHEN JL=JA+NTV1
  343 IROW=1
      QX=QX+QSX
      QY=QY+QSY
      QZ=QZ+QSZ
      IF (N.LE.JA) GO TO 342
C        COMPUTE INDUCED VELOCITY FROM PREVIOUS TRAILING AND SHED VORTEX
  345 R(NN)=DNTH(II,JSIG-1)
      QX=QX+(SGMAX-SIGMX(NN))*GAMMA(I,JSIG-1)
      QY=QY+(SGMAY-SIGMY(NN))*GAMMA(I,JSIG-1)
      QZ=QZ+(SGMAZ-SIGMZ(NN))*GAMMA(I,JSIG-1)
      SIGMX(NN)=0.
      SIGMY(NN)=0.
      SIGMZ(NN)=0.
      IF (NN.GE.NTV1) GO TO 3420
      C(1)=0.
      C(2)=0.
      C(3)=0.
 3420 XB=X(II,JL)
      YB=Y(II,JL)
      ZB=Z(II,JL)
      SGMAX=0.
      SGMAY=0.
      SGMAZ=0.
```

```
      NN=NN+1
  342 RS=DNTH(II,JSIG)
      IP1=I+1
      IF (N-JA-NTV1.LT.0) GO TO 379
      IF (N-JA-NTV1.EQ.0) GO TO 380
      WRITE (6,3444)
 3444 FORMAT (9H0342 HALT)
      STOP
C        COMPUTE SELF-INDUCED VELOCITY FOR POINT BETWEEN SHED VORTICES
  344 JLP1=JL+1
      XC=X(II,JLP1)
      YC=Y(II,JLP1)
      ZC=Z(II,JLP1)
      LTMP(2)=DNTH(II,JSIG+1)
      JLP1=JL-1
      XB=X(II,JLP1)
      YB=Y(II,JLP1)
      ZB=Z(II,JLP1)
      ATMP(1)=B(II,JSIG)
      LTMP(1)=DNTH(II,JSIG)
      JK=2
      GO TO 285
  346 IF (II.GE.NW) GO TO 3451
      FSMAL=GAMMA(II,JSIG)-GAMMA(I,JSIG)
      GO TO 347
 3451 FSMAL=-GAMMA(I,JSIG)
  347 FSCRP=0.
      IF (RT.LE.0.) GO TO 3471
      FSCRP=FSMAL*(ALOG(RT)+.25)
 3471 LTMP(3)=LTMP(1)
      LTMP(1)=LTMP(2)
      LTMP(2)=LTMP(3)
      JSIG=JSIG+1
      ATMP(1)=B(II,JSIG)
      GO TO 289
  348 IF (II.GE.NW) GO TO 3491
      FSMAL=GAMMA(II,JSIG)-GAMMA(I,JSIG)
      GO TO 349
 3491 FSMAL=-GAMMA(I,JSIG)
  349 FSCRP=0.
      IF (RT.LE.0.) GO TO 3492
      FSCRP=(FSCRP+FSMAL*(ALOG(RT)+.25))/(2.*RSCRP*MXYZ)
 3492 QSX=MX*FSCRP+QSX
      QSY=MY*FSCRP+QSY
      QSZ=MZ*FSCRP+QSZ
      JL=JL+1
      R(NN+2)=DNTH(II,JSIG)
C        COMPUTE COMPONENT OF INDUCED VELOCITY FROM PREVIOUS TRAILING
C              VORTEX
      GO TO 343
```

```
C         CALAULATE USUAL INDUCED VELOCITY COMPONENTS UNLESS JL=J,II=I
C         AVOIDS RECALCULATION OF TRAILED VORTEX SELF-INDUCED VELOCITY
C           COMPONENT.
  350 IF (JL.NE.J) GO TO 360
  351 SIGMX(1)=0.
      SIGMY(1)=0.
      SIGMZ(1)=0.
      R(1)=LNTH(I,JL)
      IP1=I+1
      NN=1
      GO TO 357
  354 XB=X(IP1,JL)
      YB=Y(IP1,JL)
      ZB=Z(IP1,JL)
      RT=(XA-XB)**2+(YA-YB)**2+(ZA-ZB)**2
      DORL=DNTH(IP1,JSIG)**2
      N=4
      SIGN=-1.
      GO TO 365
C         RETURNS TO 370
  355 SIGMX(NN)=EHNGX
      SIGMY(NN)=EHNGY
      SIGMZ(NN)=EHNGZ
      QX=QX+SGMAX*GAMMA(I,JSIG)
      QY=QY+SGMAY*GAMMA(I,JSIG)
      QZ=QZ+SGMAZ*GAMMA(I,JSIG)
      XK=35.
      NN=NN+1
      IF (NN.EQ.NTV) GO TO 390
  356 JSIG=JSIG+1
      SIGMX(NN)=0.
      SIGMY(NN)=0.
      SIGMZ(NN)=0.
  357 DORL=DNTH(IP1,JSIG)**2
      XC=X(IP1,JL)
      YC=Y(IP1,JL)
      ZC=Z(IP1,JL)
      R(NN)=LNTH(I,JL)
      JL=JL+1
      XB=X(IP1,JL)
      YB=Y(IP1,JL)
      ZB=Z(IP1,JL)
      RT=(XA-XB)**2+(YA-YB)**2+(ZA-ZB)**2
      N=2
      IROW=4
      SIGN=1.
      GO TO 365
  359 IROW=4
C         BEGIN STANDARD TYPE ROW INDUCED VELOCITY CALCULATION.
  360 SGMAX=0.
```

```
      SGMAY=0.
      SGMAZ=0.
      JL=JA
      XB=X(I,JL)
      JSIG=JSIGT
      YB=Y(I,JL)
      ZB=Z(I,JL)
      IP1=I+1
      XC=X(IP1,JL)
      YC=Y(IP1,JL)
      ZC=Z(IP1,JL)
      NN=1
      RT=(XA-XC)**2+(YA-YC)**2+(ZA-ZC)**2
      DORL=LNTH(I,JL)**2
      N=1
      SIGN=1.
  365 RS=SQRT(RT)
C     IF (CNTR.EQ.2) GO TO 372
  370 RPR=RS+R(NN)
C        TEST FOR END OF VORTEX ELEMENT (POINT A) INSIDE CURRENT VORTEX
C          ELEMENT CORE.  DEFINE HORG DIFFERENTLY ONLY IF A IS INSIDE
C          CORE OF B-C ELEMENT.
      WORK1=R(NN)
      VTEST=RS*RS+WORK1*WORK1-DORL
      IF (VTEST.GT.0) GO TO 371
      WORK3=RS+WORK1
      WORK3=WORK3*WORK3
      WORK2=RS-WORK1
      WORK2=WORK2*WORK2
      VTEST=(WORK3-DORL )*(DORL -WORK2)/(4.*DORL )
      IF (SIGN.EQ.0) STOP
      WORK1=CHORD(M)
      WORK1=WORK1*WORK1
      IF (VTEST.GT.WORK1) GO TO 371
      HORG=0.
      WORK5=SQRT(DORL)*WORK1
      IF(WORK5.NE.0.) HORG=1./WORK5
      GO TO 377
  371 HORG=0.
      WORK5=R(NN)*RS*(RPR*RPR-DORL)
      IF(WORK5.NE.0.) HORG=SIGN*RPR/WORK5
  377 EORNZ=(XA-XC)*(YC-YB)-(YA-YC)*(XC-XB)
      EORNY=(ZA-ZC)*(XC-XB)-(XA-XC)*(ZC-ZB)
      EORNX=(YA-YC)*(ZC-ZB)-(ZA-ZC)*(YC-YB)
      EHNGX=EORNX*HORG
      EHNGY=EORNY*HORG
      EHNGZ=EORNZ*HORG
  381 WORK1=GAMMA(I,JSIG)
      WORK2=ABS(EHNGX*WORK1)
      WORK3=ABS(EHNGY*WORK1)
```

```
      WORK4=ABS(EHNGZ*WORK1)
      WORK1=AMAX1(WORK2,WORK3,WORK4)
      IF (WORK1.LE.VLIM(NN)) GO TO 383
      EHNGX=EHNGX/WORK1*VLIM(NN)
      EHNGY=EHNGY/WORK1*VLIM(NN)
      EHNGZ=EHNGZ/WORK1*VLIM(NN)
  383 SGMAX=SGMAX+EHNGX
      SGMAY=SGMAY+EHNGY
      SGMAZ=SGMAZ+EHNGZ
  385 GO TO (372,378,380,355), N
  372 R(NN)=RS
      JL=JL+1
      IF (IROW.GT.2) GO TO 376
  373 IF (II-I-1) 374,375,376
  374 IF (JL.EQ.J) GO TO 354
      GO TO 376
  375 IF (JL.EQ.J) GO TO 284
  376 IP1=I+1
      XB=X(IP1,JL)
      YB=Y(IP1,JL)
      ZB=Z(IP1,JL)
      RT=(XA-XB)**2+(YA-YB)**2+(ZA-ZB)**2
      DORL=DNTH(IP1,JSIG)**2
      N=2
      SIGN=-1.
      GO TO 365
C        RETURN TO 370
  378 C(1)=EHNGX
      C(2)=EHNGY
      C(3)=EHNGZ
  379 XC=X(I,JL)
      YC=Y(I,JL)
      ZC=Z(I,JL)
      N=3
      DORL=LNTH(I,JL)**2
      SIGN=1.
      NN=NN+1
      GO TO 370
  380 NX=NN-1
      QX=QX+(SGMAX-SIGMX(NX))*GAMMA(I,JSIG)
      QY=QY+(SGMAY-SIGMY(NX))*GAMMA(I,JSIG)
      QZ=QZ+(SGMAZ-SIGMZ(NX))*GAMMA(I,JSIG)
      SIGMX(NX)=C(1)
      SIGMY(NX)=C(2)
      SIGMZ(NX)=C(3)
  382 IF (NN.EQ.NTV) GO TO 390
C        CHECK NN INCREMENTING IN SPECIAL AREAS
  384 SGMAX=-EHNGX
      SGMAY=-EHNGY
      SGMAZ=-EHNGZ
```

```
      JSIG=JSIG+1
      XC=XB
      YC=YB
      ZC=ZB
      GO TO 372
  390 R(NN)=RS
C         ADD CURRENT BLADES WAKE EFFECTS TO INDUCED VELOCITY AT A.
      IF (Z2.EQ.2) GO TO 394
  393 VX(II,J)=VX(II,J)+QX
      VY(II,J)=VY(II,J)+QY
      VZ(II,J)=VZ(II,J)+QZ
      GO TO 395
  394 VXM(II,J)=VXM(II,J)+QX
      VYM(II,J)=VYM(II,J)+QY
      VZM(II,J)=VZM(II,J)+QZ
  395 DO 392 NN=1,NTV1
      SIGMX(NN)=0.
      SIGMY(NN)=0.
  392 SIGMZ(NN)=0.
      C(1)=0.
      C(2)=C.
      C(3)=0.
  400 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TEST5
      REAL LBIG,LSMAL,LBSLS,LBSLSQ
      DIMENSION C(02)
      DIMENSION GAMMA(005,016)
      DIMENSION LSMAL(05)
      DIMENSION RCAP(02,05)
      DIMENSION RSMAL(02,20)
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,C,RSMAL,RCAP,GAMMA,QSZ
      GAMSUM=0.
      C3=3.*C(M)
      C34SQ=C3/4.
      C34SQ=C34SQ*C34SQ
      RJDPSI=RSMAL(M,JL)*DPSI
      RJPSSQ=RJDPSI*RJDPSI
      TERM1=+1./RJDPSI
      DO 1 I=1,NTV1
    1 LSMAL(I)=ABS(RCAP(M,I+1)-RCAP(M,I))
      JX=JL
    6 IF (JX.LE.NTV) GO TO 7
      JX=JX-NTV
      GO TO 6
    7 JJ=0
      NN=JSIG+NTV1-1
      DO 5 I=JSIG,NN
      JJ=JJ+1
      IF (JJ.LT.JX) GO TO 2
      KX=JX
      N=JJ
      GO TO 3
    2 KX=JJ
      N=JX-1
    3 LBIG=0.
      DO 4 K=KX,N
    4 LBIG=LSMAL(K)+LBIG
      LBSLS=LBIG-LSMAL(JJ)
      LBSLSQ=LBSLS*LBSLS
      C3LI42=C3/(4.*LBIG)
      C3LI42=C3LI42*C3LI42
      RJDPLI=RJDPSI/LBIG
      RJDPLI=RJDPLI*RJDPLI
    5 GAMSUM=GAMMA(1,I)*ALOG(
     2((1.+SQRT(1.+RJDPLI))/
     3(1.+SQRT(1.+C3LI42)))*
     4((LBSLS+SQRT(LBSLSQ+C34SQ))/
     5(LBSLS+SQRT(LBSLSQ+RJPSSQ))))+GAMSUM
      TERM1=TERM1*GAMSUM
      QSZ=QSZ+TERM1
      RETURN
      END
```

```
      SUBROUTINE T3ASP(VX,VY,VZ)
      INTEGER Z2
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS
C
      DIMENSION VLL(16)
      DIMENSION VX(05,020)
      DIMENSION VY(05,020)
      DIMENSION VZ(05,020)
C
C
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /MODWK1/ GAMMA (54,16),R (1),A (05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),X (54,16),Y (54,16),Z (54,16)
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /VLIMIT/ VLIM(5),VMLIM(16)
      COMMON /ELNTHS/ ELL(16)
      COMMON /OVT3A/ Z2,II
      COMMON /OVT3F/ J
      COMMON /VLNTHS/ NALIM,VLL
C
C        COMPUTE INDUCED VELOCITIES AT ALL VORTEX ELEMENT END POINTS,
C             AND INCLUDING BLADE VORTEX POINTS. INCLUDE SELF-INDUCED
C             EFFECTS DUE TO BLADE AND NEIGHBORING VORTICES (BASED ON
C             APPROXIMATE CURVATURE FROM AN ARC DETERMINED FROM 3-POINT
C             CIRCULAR ARC CURVE FITTING.
C
C
C        INITIALIZE CONTROL CONSTANTS FOR WAKE INDUCED VELOCITY
C             CALCULATIONS.
C
      IF (Z2.EQ.2) NSW1=1
      NN=1
      NWSTM1=NWSTRE-1
      DO 400 JA=1,NIBRVM
      EEL=VLL(JA)
      TWOEL=EEL+EEL
      NSW=-1
      EL=ELL(JA)
      NANRX=NANRM
      JL=JA
      JSIG=JA
      QX=0.
      QY=0.
      QZ=0.
      WORK1=XA-X(NANRM,JL)
      WORK2=YA-Y(NANRM,JL)
      WORK3=ZA-Z(NANRM,JL)
      RSQ=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
      R(1)=SQRT(RSQ)
```

```
      IF (R(1).NE.0) GO TO 5
      NANRX=NANRM+1
      IF (NANRX.GT.NWSTM1) GO TO 400
      WORK1=XA-X(NANRX,JL)
      WORK2=YA-Y(NANRX,JL)
      WORK3=ZA-Z(NANRX,JL)
      R(1)=SQRT(WORK1*WORK1+WORK2*WORK2+WORK3*WORK3)
    5 I=NANRX-1
      XB=X(NANRX,JL)
      YB=Y(NANRX,JL)
      ZB=Z(NANRX,JL)
   10 I=I+1
      IF (NWSTRE.GT.II.AND.I.EQ.II.AND.JA.EQ.J) GO TO 390
C        BEGIN STANDARD TYPE ROW INDUCED VELOCITY CALCULATION.
      IP1=I+1
      XC=X(IP1,JL)
      YC=Y(IP1,JL)
      ZC=Z(IP1,JL)
      NRETN=0
      IF ((II.EQ.I.OR.II-1.EQ.I).AND.JA.EQ.J.AND.Z2.EQ.2)
     1 CALL SELFIN (I,II,JA,QX,QY,QZ,NRETN)
      IF (NRETN.EQ.1) GO TO 390
      WORK1=XA-XC
      WORK2=YA-YC
      WORK3=ZA-ZC
      RT=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
      IF(RT.GT.TWOEL) GO TO 362
      IF (RSQ.LT.EEL) GO TO 364
      IF (RT.LT.EEL) GO TO 364
      NSW=0
      GO TO 389
  362 NSW=1
      I=I+NALIM
      GO TO 389
  364 IF (NSW.LT.1) GO TO 363
      I=I-NALIM
      NSW=-1
      GO TO 390
  363 WORK1=XB-XC
      WORK2=YB-YC
      WORK3=ZB-ZC
      DORL=WORK1*WORK1+WORK2*WORK2+WORK3*WORK3
      NSW=-2
  365 RS=SQRT(RT)
  370 RPR=RS+R(NN)
C        TEST FOR END OF VORTEX ELEMENT (POINT A) INSIDE CURRENT VORTEX
C          ELEMENT CORE.  DEFINE HORG DIFFERENTLY ONLY IF A IS INSIDE
C          CORE OF B-C ELEMENT.
      WORK1=R(NN)
      VTEST=RS*RS+WORK1*WORK1-DORL
```

```
      IF (VTEST.GT.0) GO TO 371
      WORK3=RS+WORK1
      WORK3=WORK3*WORK3
      WORK2=RS-WORK1
      WORK2=WORK2*WORK2
      VTEST=(WORK3-DORL )*(DORL -WORK2)/(4.*DORL )
      WORK1=A(NANRM,JA)
      WORK1=WORK1*WORK1
      IF (VTEST.GT.WORK1) GO TO 371
      HORG=0.
      WORK5=SQRT(DORL)*WORK1
      IF(WORK5.NE.0.) HORG=1./WORK5
      GO TO 377
  371 HORG=0.
      WORK5=R(NN)*RS*(RPR*RPR-DORL)
      IF(WORK5.NE.0.) HORG=RPR/WORK5
  377 EORNZ=(XA-XC)*(YC-YB)-(YA-YC)*(XC-XB)
      EORNY=(ZA-ZC)*(XC-XB)-(XA-XC)*(ZC-ZB)
      EORNX=(YA-YC)*(ZC-ZB)-(ZA-ZC)*(YC-YB)
      EHNGX=EORNX*HORG
      EHNGY=EORNY*HORG
      EHNGZ=EORNZ*HORG
  381 WORK1=GAMMA(I,JSIG)
      WORK2=ABS(EHNGX*WORK1)
      WORK3=ABS(EHNGY*WORK1)
      WORK4=ABS(EHNGZ*WORK1)
      WORK1=AMAX1(WORK2,WORK3,WORK4)
      IF (WORK1.LE.VMLIM(JSIG)) GO TO 383
      EHNGX=EHNGX/WORK1*VMLIM(JSIG)
      EHNGY=EHNGY/WORK1*VMLIM(JSIG)
      EHNGZ=EHNGZ/WORK1*VMLIM(JSIG)
  383 WORK1=EL/SQRT(DORL)
      SGMAX=EHNGX*WORK1
      SGMAY=EHNGY*WORK1
      SGMAZ=EHNGZ*WORK1
  372 R(NN)=RS
      XB=XC
      YB=YC
      ZB=ZC
      QX=QX+(SGMAX           )*GAMMA(I,JSIG)
      QY=QY+(SGMAY           )*GAMMA(I,JSIG)
      QZ=QZ+(SGMAZ           )*GAMMA(I,JSIG)
  389 RSQ=RT
  390 IF (I.LT.NANRX) GO TO 391
      IF (I.LT.NWSTM1) GO TO 10
  391 IF (Z2.EQ.2) GO TO 393
      VX(II,J)=VX(II,J)+QX
      VY(II,J)=VY(II,J)+QY
      VZ(II,J)=VZ(II,J)+QZ
      GO TO 400
```

```
393 VXM(II,J)=VXM(II,J)+QX
    VYM(II,J)=VYM(II,J)+QY
    VZM(II,J)=VZM(II,J)+QZ
400 CONTINUE
    RETURN
    END
```

```
      SUBROUTINE SELFIN (I,II,JA,QX,QY,QZ,NRETN)
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /MODWK1/ GAMMA (54,16),R (1),AM(05,16),VXM(54,16),
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      NRETN=1
      A=AM(NANRM,JA)
      I1=II
      IF (NWSTRE.EQ.II) I1=II-1
      IM1=I1-1
      IP1=I1+1
      XA=XM(I1,JA)
      YA=YM(I1,JA)
      ZA=ZM(I1,JA)
      XB=XM(IM1,JA)
      YB=YM(IM1,JA)
      ZB=ZM(IM1,JA)
      XC=XM(IP1,JA)
      YC=YM(IP1,JA)
      ZC=ZM(IP1,JA)
      CALL POINTA (JA,I,A,JA,QSX,QSY,QSZ,II,XA,YA,ZA,XB,YB,ZB,XC,YC,ZC,
     1 GAMMA)
      QX=QX+QSX
      QY=QY+QSY
      QZ=QZ+QSZ
      RETURN
      END
```

```
      SUBROUTINE POINTA (JL,I,A,JSIG,QSX,QSY,QSZ,II,XA,YA,ZA,XB,YB,ZB,
     1 XC,YC,ZC,GAMMA)
      REAL LTMP(3),MX,MY,MZ,MXYZ
      DIMENSION GAMMA(54,16)
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      IX=II
      JK=0
      N=JL
      WORK1=XA-XB
      WORK2=YA-YB
      WORK3=ZA-ZB
      LTMP(1)=SQRT(WORK1*WORK1+WORK2*WORK2+WORK3*WORK3)
      WORK1=XA-XC
      WORK2=YA-YC
      WORK3=ZA-ZC
      LTMP(2)=SQRT(WORK1*WORK1+WORK2*WORK2+WORK3*WORK3)
      ATMP   =A
  285 MX=(YB-YA)*(ZA-ZC)-(YA-YC)*(ZB-ZA)
      MY=(ZB-ZA)*(XA-XC)-(ZA-ZC)*(XB-XA)
      MZ=(XB-XA)*(YA-YC)-(XA-XC)*(YB-YA)
      RT=MX*MX+MY*MY+MZ*MZ
      IF (RT.EQ.0.) RT=1.
      RS=SQRT(RT)
      MXYZ=RS
      DELSQ=(XB-XC)**2+(YB-YC)**2+(ZB-ZC)**2
      WORK3=   (4.*(LTMP(1)*LTMP(2))**2-(LTMP(1)**2+LTMP(2)**2-DELSQ)**
     12)
      IF (WORK3.LE.0) WORK3=5.4E-70
      RT=DELSQ/WORK3
      RS=SQRT(RT)
      RSCRP=2.*LTMP(1)*LTMP(2)*RS
      IF (II.EQ.NWSTRE) GO TO 306
  289 RT=RSCRP**2-LTMP(1)**2
      JK=JK+1
      IF (RT.LT.0.) RT=0.
      RS=SQRT(RT)
C        COMPUTE FSMAL DEPENDING ON L(1)**2.GE.DELSQ)L(2)**2.
      IF (LTMP(1)**2-DELSQ-LTMP(2)**2.GT.0) GO TO 294
  292 FSMAL=(RSCRP-RS)/LTMP(1)
      GO TO 296
  294 FSMAL=(RSCRP+RS)/LTMP(1)
  296 RT=4.*FSMAL*RSCRP/ATMP
C        COMPUTE TRAILED (JK=1,2)                    CONTRIBUTION TO
C          SELF-INDUCED VELOCITY.
      GO TO (300,316),JK
  300 IF (RT.EQ.0.) GO TO 307
      FSCRP=GAMMA(I,JSIG)*(ALOG(RT)+.25)
      GO TO 308
  306 IX=II-1
  307 FSCRP=0.
```

```
  308 LTMP(3)=LTMP(1)
      LTMP(1)=LTMP(2)
      LTMP(2)=LTMP(3)
      GO TO 289
  316 IF (RT.EQ.0.) GO TO 322
      FSCRP=(FSCRP+GAMMA(IX,JSIG)*(ALOG(RT)+.25))/(2.*RSCRP*MXYZ)
C        DEFINE TRAILED VORTEX SELF-INDUCED VELOCITY COMPONENT.
  322 QSX=MX*FSCRP
      QSY=MY*FSCRP
      QSZ=MZ*FSCRP
      RETURN
      END
```

```
      SUBROUTINE STEPX
      INTEGER OUT,WKPT,CNTR
      INTEGER T45,WW
      REAL MXYZ,MU,MUALT,MUSPK,MUCAT,MUSAT,MUDP,MUSDP,LLNTH,LNTH,LOADN,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS,IO,MB,LX,KXX
C
      DIMENSION A(05,20)
      DIMENSION ALFA1(02)
      DIMENSION ALFA2(02)
      DIMENSION ALFAS(02)
      DIMENSION ALFAT(02)
      DIMENSION ALPHAO(02)
      DIMENSION ALPHAR(02)
      DIMENSION AO(02)
      DIMENSION AR(02)
      DIMENSION AT(5)
      DIMENSION ATMP(5)
      DIMENSION B(005,16)
      DIMENSION BETA(03,2)
      DIMENSION BT(5)
      DIMENSION BTMP(5)
      DIMENSION C(09)
      DIMENSION CCLA(02)
      DIMENSION CHORD(02)
      DIMENSION DELTA(02)
      DIMENSION DI(90)
      DIMENSION DIR(2)
      DIMENSION DLNTH(016)
      DIMENSION DNTH(005,016)
      DIMENSION DSQ(016)
      DIMENSION DTMP(016)
      DIMENSION GAMMA(05,016)
      DIMENSION GAMMAG(144)
      DIMENSION GAMMK(1,016)
      DIMENSION INDXG(16)
      DIMENSION IO(02)
      DIMENSION KXX(02)
      DIMENSION LLNTH(020)
      DIMENSION LNTH(05,20)
      DIMENSION LOADN(020)
      DIMENSION LSQ(020)
      DIMENSION LTMP(020)
      DIMENSION LX(02)
      DIMENSION MB(02)
      DIMENSION MUCDP(02)
      DIMENSION MUCDS(2)
      DIMENSION MUSDP(02)
      DIMENSION MUSDS(2)
      DIMENSION NPSI(2)
      DIMENSION PSI(2)
```

```
      DIMENSION PSIR(02)
      DIMENSION R(5)
      DIMENSION RBAR(016)
      DIMENSION RCAP(02,5)
      DIMENSION RSMLL(02,20)
      DIMENSION RZERO(2)
      DIMENSION SGMA1(016,016)
      DIMENSION SGMA2(05,16)
      DIMENSION SIGBL(144)
      DIMENSION SIGMX(04)
      DIMENSION SIGMY(04)
      DIMENSION SIGMZ(04)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(02)
      DIMENSION THTAY(02)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,20)
      DIMENSION VY(05,20)
      DIMENSION VZ(05,20)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,20)
      DIMENSION XROT(02)
      DIMENSION XSIMQ(016,016)
      DIMENSION XX(02)
      DIMENSION Y(05,20)
      DIMENSION YROT(02)
      DIMENSION Z(05,20)
      DIMENSION ZAP(16)
      DIMENSION ZROT(02)
C
C
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /TEST33/ NIB,NROT,NW,NBRV1,X,Y,Z,A,B,DNTH,LNTH,NAR,NPER,
     1JSIGT,NWMK,J,NWM1,NIBV,VX,VY,VZ,NIBRV
      COMMON /TEST55/ JL,JSIG,NTV,NTV1,M,DPSI,CHORD,RSMLL,RCAP,GAMMA,QSZ
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /SUBIE/ NAS
      COMMON /WK1C/ PSI,CCLA,DIR
      COMMON /WK1A/ PSIR,DPSIK,PSIK,DELTA,RREF,MUCDS,MUSDS,THTAX,THTAY
      COMMON /WKCONT/ NWKPD
      COMMON /STEPXA/ WKPT,WW,IOUT,NOTTP1,KAT,NBC
      COMMON /MODCNT/ NTVM,NWSTRE,NWR,NANRM,NIBRVM,NIBM
      COMMON /MODWK1/ GAMMAM(54,16),RM(1),AM(05,16),VXM(54,16),
```

```
     1VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /MODWK3/ AFM(4,10),BFM(4,4)
      COMMON /OUTDI/ NNTV,NEXPWK,NSIGRW,NMODR,NMODC
      COMMON /OUTDII/ NWKCLM
      COMMON /CONVGC/ GAMMAG
      COMMON /WK2C/ SIGBL
      COMMON /WK4A/ VXX
C
C
C        COMPUTE NEW VORTEX ELEMENT END POINT LOCATIONS
C
      IF (NW.LT.NANRM) NW=NW+1
      IF (NWSTRE.LT.NWR) NWSTRE=NWSTRE+1
      NWSTM1=NWSTRE-1
      NWM1=NW-1
      NVM2=NMODC
  416 DO 420 M=1,NROT
      MUCAT=MUCDS(M)
      MUSAT=MUSDS(M)
      DO 420 I=1,NWM1
      K=NW-I
      L=NW-I+1
      J=(M-1)*NIBV
      DO 418 JJ=1,NIBV
      J=J+1
      X(L,J)=X(K,J)+VX(K,J)*DPSI+MUCAT
      Y(L,J)=Y(K,J)+VY(K,J)*DPSI
  418 Z(L,J)=Z(K,J)+VZ(K,J)*DPSI+MUSAT
      IF (I.EQ.1.OR.M.GT.1) GO TO 420
      DO 419 N=1,NGJR
  419 GAMMA(L,N)=GAMMA(K,N)
  420 CONTINUE
C
C        TRANSPORT MODIFIED WAKE
C
      IF (NAS.LT.NANRM) GO TO 560
      CALL MODCOX (NROT,NIB,NTV,NTVM,A,AM,AFM,NEXPWK,NNTV,NMODR,NANRM,
     1 NVM2,NAS)
       CALL MCDCOR (NROT,NIB,NTV,NTVM,VX,VY,VZ,VXM,VYM,VZM,AFM,
     2 NWKRW,NWKCLM,NMODR,NANRM,NVM2)
      DO 550 M=1,NROT
      MUCAT=MUCDS(M)
      MUSAT=MUSDS(M)
      NWSTMF=NWSTRE-NANRM
      DO 550 I=1,NWSTMF
      K=NWSTRE-I
      L=K+1
      J=(M-1)*NIBM
      DO 510 JJ=1,NIBM
      J=J+1
```

```
      XM(L,J)=XM(K,J)+VXM(K,J)*DPSI+MUCAT
      YM(L,J)=YM(K,J)+VYM(K,J)*DPSI
  510 ZM(L,J)=ZM(K,J)+VZM(K,J)*DPSI+MUSAT
      IF (I.EQ.1.OR.M.GT.1) GO TO 550
      DO 520 N=1,NIBRVM
  520 GAMMAM(L,N)=GAMMAM(K,N)
  550 CONTINUE
      CALL GMS (NANRM,NROT,NIB,NTVM,BFM,GAMMA,GAMMAM,NTV1,NEXPWK,
     1 NSIGRW,NMODR,NMODC)
  560 DO 441 M=1,NROT
  441 PSI(M)=PSI(M)+DPSI*DIR(M)
      NAS=NAS+1
      IF (NAS.GE.NANRM)
     1CALL MODCOR(NROT,NIB,NTV,NTVM,X,Y,Z,XM,YM,ZM,AFM,NEXPWK,NNTV,
     1 NMODR,NANRM,NVM2)
      IF (NAS.EQ.NANRM+1)
     1CALL MODCOR(NROT,NIB,NTV,NTVM,X,Y,Z,XM,YM,ZM,AFM,NEXPWK,NNTV,
     1 NMODR,NANRM-1,NVM2)
      WRITE (6,3) NAS,NW,NWSTRE
    3 FORMAT (30HOTHE NUMBER OF WAKE POSITIONS ,3I5)
      IF (NAS.GE.WW) NWKPD=4
      IF (NWKPD.EQ.4) NPER=4
      II=0
C
  902 FORMAT (2I3/(8F10.7))
C
C         OUTPUT BRANCH CONTROL
C
C
C     START NEW COMPUTATION WITH NO OUTPUT IF NBC.EQ.1
C
      GO TO (440,442,444), NBC
  442 WRITE (NOTTP1) X,Y,Z,GAMMA,VI,SIGBL,GAMMAG,PSI,II,NAS,NW,NWM1,
     1 NPER,NAS1,NWKPD,LNTH,DNTH,A,B,VX,VY,VZ,VXX,
     2 VXM,VYM,VZM,XM,YM,ZM,GAMMAM,RM,AM,NWSTRE
      REWIND NOTTP1
      IF (KAT.EQ.1) STOP
      IF (NAS.GE.WW+NIBNA) STOP
      RETURN
  444 WRITE(IOUT,902) NW,NIBRV,((X(I,J),Y(I,J),Z(I,J),I=1,NW),J=1,
     1NIBRV),PSI
      WRITE(IOUT,902) NW,NGJR,((GAMMA(I,J),I=1,NW),J=1,NGJR)
  440 RETURN
      END
```

```
      SUBROUTINE GMS (I,NROT,NIB,NTVM,BFM,GAMMA,GAMMAM,NTVM1,NANR,NGJR,
     1 NMODR,NMODC)
      DIMENSION BFM(4,4)
      DIMENSION GAMMA(5,16)
      DIMENSION GAMMAM(NMODR,NMODC)
      IM2=I-1
      IM1=I
      DO 10 M=1,NROT
      MM1=M-1
      NIBMM1=NIB*MM1
      DO 10 K=1,NIB
      KM1=K-1
      KNIBM=KM1+NIBMM1
      NTVMK=NTVM*KNIBM
      NTVK=NTVM1*KNIBM
      DO 10 LM=1,NTVM
      JM=LM+NTVMK
      GAMMAM(IM1,JM)=0.
      DO 10 L=1,NTVM1
      J=L+NTVK
      GAMMAM(IM1,JM)=-BFM(LM,L)*GAMMA(IM2,J)
     1 +GAMMAM(IM1,JM)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MODCOR (NROT,NIB,NTV,NTVM,X,Y,Z,XM,YM,ZM,AFM,NANV,NBNT,
     1NWM,I,NVM2)
      DIMENSION AFM(4,10)
      DIMENSION X(NANV,NBNT)
      DIMENSION XM(NWM,NVM2)
      DIMENSION Y(NANV,NBNT)
      DIMENSION YM(NWM,NVM2)
      DIMENSION Z(NANV,NBNT)
      DIMENSION ZM(NWM,NVM2)
C
C     MODIFIED WAKE ELEMENT END POINT POSITIONS OR VELOCITIES
C
      DO 10 M=1,NROT
      DO 10 K=1,NIB
      DO 10 LM=1,NTVM
      JM=LM+NTVM*(K-1+NIB*(M-1))
      XM(I,JM)=0.
      YM(I,JM)=0.
      ZM(I,JM)=0.
      DO 10 L=1,NTV
      J=L+NTV*(K-1+NIB*(M-1))
      N=L+NTV*(M-1)
      XM(I,JM)=AFM(LM,N)*X(I,J)+XM(I,JM)
      YM(I,JM)=AFM(LM,N)*Y(I,J)+YM(I,JM)
      ZM(I,JM)=AFM(LM,N)*Z(I,J)+ZM(I,JM)
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MODCOX (NROT,NIB,NTV,NTVM,X,XM,AFM,NANV,NBNT,NWM,I,
     1 NVM2,NAS)
      DIMENSION AFM(4,10)
      DIMENSION X(NANV,NBNT)
      DIMENSION XM(NANV,NVM2)
C
C     MODIFIED WAKE ELEMENT END POINT POSITIONS OR VELOCITIES
C
      DO 20 M=1,NROT
      DO 20 K=1,NIB
      DO 20 LM=1,NTVM
      JM=LM+NTVM*(K-1+NIB*(M-1))
      IF (NAS.EQ.I.AND.LM.LT.NTVM) GO TO 15
      IF (LM.LT.NTVM) GO TO 20
      XM(I,JM)=0.
      GO TO 16
   15 READ (5,100) XM(I,JM)
  100 FORMAT (29X,E14.7)
      GO TO 20
   16 DO 10 L=1,NTV
      J=L+NTV*(K-1+NIB*(M-1))
      N=L+NTV*(M-1)
      XM(I,JM)=AFM(LM,N)*X(I-1,J)+XM(I,JM)
   10 CONTINUE
   20 CONTINUE
      WRITE(6,1002) NAS,(XM(I,J),J=1,8)
 1002 FORMAT (7HOMODCOX,9(1X,G11.4))
      RETURN
      END
```

Machine Capability

The Wake Geometry Program has been run on the University of Rochester's IBM 360/65 under MVT Release 18, General Computer Corporation's CDC 6600 under Scope 3.2, and NASA-Langley's CDC 6600 under Scope 3.2. The program is standard FORTRAN IV and is also WATFIV compatible.

Recommended CDC 6600 Overlay Statements:

Mainline:

```
OVERLAY(WKOVL,0,0)
PROGRAM GEOM(INPUT, OUTPUT, BDSIG, BDGAM, TAPE5=INPUT,
   TAPE6=OUTPUT, TAPE4=BDSIG, TAPE8=BDGAM)
```

Subroutine WK1:

```
CALL OVERLAY(5LWKOVL,3,0,6HRECALL) replaces CALL WK2
CALL OVERLAY(5LWKOVL,4,0,6HRECALL) replaces CALL CONVG
CALL OVERLAY(5LWKOVL,5,0,6HRECALL) replaces CALL AL1BD2
CALL OVERLAY(5LWKOVL,6,0,6HRECALL) replaces CALL APXLD
CALL OVERLAY(5LWKOVL,7,0,6HRECALL) replaces CALL WK3
CALL OVERLAY(5LWKOVL,8,0,6HRECALL) replaces CALL STEPX
```

Subroutine WK2:

```
OVERLAY(WKOVL,3,0)
PROGRAM WK2
```

Subroutine CONVG:

```
OVERLAY(WKOVL,4,0)
PROGRAM CONVG
```

Subroutine AL1BD2:

```
OVERLAY(WKOVL,5,0)
PROGRAM AL1BD2
```

Subroutine APXLD:

```
OVERALY(WKOVL,6,0)
PROGRAM APXLD
```

Subroutine WK3:

```
OVERLAY(WKOVL,7,0)
PROGRAM WK3
```

Subroutine STEPX:

```
OVERLAY(WKOVL,10,0)
PROGRAM STEPX
```

```
C       MAIN PROGRAM BLD34
        DIMENSION VX(20,19),WX(20,19)
        DIMENSION CSIX(114),CSIA(114),CSIB(114)
        DIMENSION DUMBD4(2122),DOMBD4(2122)
        DIMENSION SIGKJA(6,6)
        DIMENSION SIGKJB(6,6)
        DIMENSION SINEBJ(20)
        DIMENSION SINEJ(18)
        DIMENSION SAVE(144),WBR(08,18),EL(08,18)
        DIMENSION HDOT(08,18),PHI(08,18),THET(08,18)
        DIMENSION ELNTA(20),ELNTB(20),EMAA(20),EMAB(20),EIXA(20),EIXB(20)
        DIMENSION EIZA(20),EIZB(20)
        DIMENSION RBL(08),RBLA(4),RBLB(4)
        DIMENSION XINA(20),XINB(20),DPHA(20),DPHB(20),EIYA(20),EIYB(20)
        DIMENSION UNKWN(144),SV3(144)
        DIMENSION EPA(20),EPB(20),DLA(20),DLB(20),ZAA(20),ZAB(20)
        DIMENSION WDOT(4,18),WPHI(4,18),WTHET(4,18)
        DIMENSION FORCX(08,18),FORCZ(08,18),EMOME(08,18),YINA(20),YINB(20)
        DIMENSION WFX(4,18),WFZ(4,18),WEMO(4,18),BD1(15),BD2(15)
        DIMENSION BI(08),BET(08)
        DIMENSION VDOT(20,19),XDOT(20,19),PHIDT(20,19),SIDT(20,19),
       1 CSIDT(06,19)
        DIMENSION OMEGA(06),SIGA(06),AVA(20,6),AWA(20,6),APHIA(20,6),
       1 ASIA(20,6), ATHEA (20,6),ATA(20,6),AMZA(20,6),AVYA(20,6),
       2 AMYA(20,6),AVZA(20,6),CSIDTA(1,18)
        DIMENSION OMEGB(06),SIGB(06),AVB(20,6),AWB(20,6),APHIB(20,6),
       1 ASIB(20,6),ATHEB(20,6),ATB(20,6),AMZB(20,6),AVYB(20,6),
       2 AMYB(20,6),AVZB(20,6),CSIDTB(1,18)
        DIMENSION DUMSDA(1634),DUMSDB(1634)
C
        EQUIVALENCE (BD1(1),ZY),(BD1(2),THETA),(BD1(3),XROOA),(BD1(4),AKA)
       1,(BD1(5),ACA),(BD1(6),BCA),(BD1(7),ISECA),(BD1(8),NRPTA),
       2 (BD1(9),CTA),(BD1(10),ALPTA),(BD1(11),EMTA),(BD1(12),AKIA),
       3 (BD1(13),OMSQA),(BD1(14),RA)
C
        EQUIVALENCE (BD2(1),ZQ),(BD2(2),THETB),(BD2(3),XROOB),
       1(BD2(4),AKB),(BD2(5),ACB),(BD2(6),BCB),(BD2(7),ISECB),
       2(BD2(8),NRPTB),(BD2(9),CTB),(BD2(10),ALPTB),(BD2(11),EMTB),
       3 (BD2(12),AKIB),(BD2(13),OMSQB),(BD2(14),RB)
C
        EQUIVALENCE (DUMBD4(1),ELNTA(1)),(DUMBD4(21),EMAA(1)),(DUMBD4(41),
       1EIXA(1)),(DUMBD4(61),RBLA(1)),(DUMBD4(70),XINA(1)),(DUMBD4(90),DPH
       2A(1)),(DUMBD4(110),EIYA(1)),(DUMBD4(130),EPA(1)),(DUMBD4(150),DLA(
       31)),(DUMBD4(170),ZAA(1)),(DUMBD4(190),YINA(1)),(DUMBD4(210),OMEGA(
       41)),(DUMBD4(69),PSIRA),
       4    (DUMBD4(230),SIGA(1)),(DUMBD4(250),AVA(1)),(DUMBD4(430),AWA(1))
       5,
       5(DUMBD4(610),APHIA(1)),(DUMBD4(790),ASIA(1)),(DUMBD4( 970),ATHEA(1
       6)),(DUMBD4(1150),ATA(1)),(DUMBD4(1330),AMZA(1)),(DUMBD4(1510),AVYA
       7(1)),(DUMBD4(1690),AMYA(1)),(DUMBD4(1870),AVZA(1)),(DUMBD4(2050),B
```

```
     8D1(1)),(DUMBD4(2065),RWKA),(DUMBD4(2066),CSALA),(DUMBD4(2067),NMA)
     8,
     9(DUMBD4(2068),CSIDTA(1,1))
      EQUIVALENCE (DUMBD4(2086),SIGKJA(1,1)),(DUMBD4(2122),DAMPCA)
C
      EQUIVALENCE (DOMBD4(1),ELNTB(1)),(DOMBD4(21),EMAB(1)),(DOMBD4(41),
     1EIXB(1)),(DOMBD4(61),RBLB(1)),(DOMBD4(70),XINB(1)),(DOMBD4(90),DPH
     2B(1)),(DOMBD4(110),EIYB(1)),(DOMBD4(130),EPB(1)),(DOMBD4(150),DLB(
     31)),(DOMBD4(170),ZAB(1)),(DOMBD4(190),YINB(1)),(DOMBD4(210),OMEGB(
     41)),(DOMBD4(69),PSIRB),
     4    (DOMBD4(230),SIGB(1)),(DOMBD4(250),AVB(1)),(DOMBD4(430),AWB(1))
     5,
     5(DOMBD4(610),APHIB(1)),(DOMBD4(790),ASIB(1)),(DOMBD4( 970),ATHEB(1
     6)),(DOMBD4(1150),ATB(1)),(DOMBD4(1330),AMZB(1)),(DOMBD4(1510),AVYB
     7(1)),(DOMBD4(1690),AMYB(1)),(DOMBD4(1870),AVZB(1)),(DOMBD4(2050),B
     8D2(1)),(DOMBD4(2065),RWKB),(DOMBD4(2066),CSALB),(DOMBD4(2067),NMB)
     8,
     9(DOMBD4(2068),CSIDTB(1,1))
      EQUIVALENCE (DOMBD4(2086),SIGKJB(1,1)),(DOMBD4(2122),DAMPCB)
      EQUIVALENCE (DUMSDA(1),VDOT(1,1))
C
      COMMON/SAD3/CSIX
      COMMON /PUNCH/ NPCH
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /SHEAR1/ NASHER
      COMMON/BLD4X1/DUMBD4
      COMMON /BLD4X2/ WDOT,WPHI,WTHET,WFX,WFZ,WEMO,NMAS,NMODE,NAPSON,
     1 NR11
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /BLD3X1/ ALPT1,ALPT2,CT1,CT2,NRNTNA,NRNT,UNKWN,SV3,RBL,
     1 FORCZ,EMOME,HDOT,PHI,THET,BI,BET,SAVE,WBR,EL,FORCX
      COMMON /SAD2/ VDOT,XDOT,PHIDT,SIDT,CSIDT
      COMMON /GARY2/ XINPT,FINPT
      COMMON /SGSLS/ SIGLM,WBRLM
      COMMON /SAD6/ VX,WX
      COMMON/PRNT/NPRNT
C
      ISW=2
        ISW=1
      DO 2384 I=1,2095
      DUMBD4(I)=0.
 2384 DOMBD4(I)=0.
      DO 2344 I=1,114
      CSIA(I)=0.
      CSIB(I)=0.
```

```
 2344 CSIX(I)=0.
      DO 2389 I=1,1634
      DUMSDB(I)=0.
 2389 DUMSDA(I)=0.
      IN=5
      NOUT=6
      IT7=8
      IT8=7
      READ(IN,899)
      READ(IN,900) NBL,NB,NR,NA,NW,NMA,NMB,KTEST,NCV,NROT
      READ(IN,900) NIT1,NIT2,NIT3,MAXMO
      READ (IN,900) NPCH,NPRNT
      READ(IN,901)AMU,ALL1,ALL2,ATIME
      READ(IN,901) CPOMG,ROAIR
      READ (IN,901) XINPT,FINPT
      READ(IN,901) SIGLM,WBRLM
      NR1=NR/2
      IF (NBL.EQ.1) NR1=NR
      NR2=NR1+1
      NRNT=NR
      NAPSON=NA+1
      NR11=NR1+1
      NRNTNA=NRNT*NA
      NAOVTO=NA/2+1
      NROT=NBL
      NMAS=NMA
      NMODE=MAXMO
      NASHER=NAOVTO
      NC=NR
C
C
      WRITE(NOUT,899)
      WRITE(NOUT,9212)
      WRITE(NOUT,920) NBL,NB,   NR,NA,NW,CPOMG,ROAIR,AMU
      WRITE(NOUT,924)ALL1,ALL2
      IF (FINPT.LT..5.OR.FINPT.GT.1.0) FINPT=.5
      WRITE(NOUT,9216) SIGLM,WBRLM
 9216 FORMAT(39X,31HLIMIT ON OFF-DIAGONAL SIGMAS = ,F7.4,/,
     1       35X,35HLIMIT ON WAKE-INDUCED VELOCITIES = ,F7.4,//)
      WRITE(NOUT,9211)
C
C
      PI=3.1415926
      TWOPI=2.*PI
      DSI=TWOPI/NA
      NRP1=NR+1
      NAP1=NA+1
      NWP1=NW+1
      IJ=NA*NR
      IT3=1
```

```
      I1=1
      I2=NR1
C
C
      CALL  INPUT(BI,BET,ISECA,NRPTA,ELNTA,EIXA,EIYA,XINA,YINA,
     1 EMAA,DPHA,EPA,DLA,ZAA,ALPTA,CTA,EMTA,OMSQA,AKIA,THETA,XROOA,
     2 AKA,RA,ACA,BCA,ATA,AMZA,AVYA,AMYA,AVZA,AVA,AWA,APHIA,
     3 ASIA,ATHEA,SIGA,OMEGA,RBL,EIZA,RWKA,I1,I2,NMA,PSIRA,DIRA,
     1 NA,NR1,NRNT,NMAS,NMODE,NC,DAMPCA,SIGKJA)
C
C     DO 7326 I=1,NMAS
C     DO 7327 K=1,3
C7327 APHIA(I,K)=0.
C     AVA(I,4)=0.
C     DO 7329 K=5,6
C7329 APHIA(I,K)=0.
C7326 CONTINUE
C
      CALL  OUTPUT(THETA,XROOA,AKA,ACA,BCA,RA,CTA,ALPTA,EMTA,AKIA,
     1 OMSQA,BI,BET,ELNTA,EMAA,EIXA,EIZA,XINA,YINA,DPHA,EPA,DLA,
     2 ZAA,OMEGA,SIGA,AVA,AWA,APHIA,ASIA,ATHEA,ATA,AMZA,AVYA,AMYA,AVZA,
     3 RBL,EIYA,RWKA,I1,I2,NMA,PSIRA,DIRA,NR1,NMAS,NMODE,NC,DAMPCA,
     4 SIGKJA)
C
C
      IF(NBL-2)9214,9213,9213
9213  WRITE(NOUT,9215)
      I1=NR2
      I2=NR
C
C
      CALL  INPUT(BI,BET,ISECB,NRPTB,ELNTB,EIXB,EIYB,XINB,YINB,
     1 EMAB,DPHB,EPB,DLB,ZAB,ALPTB,CTB,EMTB,OMSQB,AKIB,THETB,XROOB,
     2 AKB,RB,ACB,BCB,ATB,AMZB,AVYB,AMYB,AVZB,AVB,AWB,APHIB,ASIB,ATHEB
     3 ,SIGB,OMEGB,RBL,EIZB,RWKB,I1,I2,NMB,PSIRB,DIRB,
     1 NA,NR1,NRNT,NMAS,NMODE,NC,DAMPCB,SIGKJB)
C
C     DO 7330 I=1,NMAS
C     DO 7331 K=1,3
C7331 APHIB(I,K)=0.
C     AVB(I,4)=0.
C     DO 7333 K=5,6
C7333 APHIB(I,K)=0.
C7330 CONTINUE
C
      CALL  OUTPUT(THETB,XROOB,AKB,ACB,BCB,RB,CTB,ALPTB,EMTB,AKIB,
     1 OMSQB,BI,BET,ELNTB,EMAB,EIXB,EIZB,XINB,YINB,DPHB,EPB,DLB,
     2 ZAB,OMEGB,SIGB,AVB,AWB,APHIB,ASIB,ATHEB,ATB,AMZB,AVYB,AMYB,AVZB,
     3 RBL,EIYB,RWKB,I1,I2,NMB,PSIRB,DIRB,NR1,NMAS,NMODE,NC,DAMPCB,
     4 SIGKJB)
```

```
C
C       RECALCULATE SOME INPUT VALUES
9214    CONTINUE
        CNVRT=PI/180.0
        CALL CONV(CNVRT,THETA,ACA,BCA,ALPTA,DPHA,NMA,NMAS)
        WBARA=WBAR(AMU,ALPTA,CTA)
        AMSNA=AMU*SIN(ALPTA)
        CSALA=COS(ALPTA)
        AMCSA=AMU*CSALA
        AMSNB=0.
        IF (NBL.NE.2) GO TO 38
        CALL CONV(CNVRT,THETB,ACB,BCB,ALPTB,DPHB,NMB,NMAS)
C
        WBARB=WBAR(AMU,ALPTB,CTB)
        AMSNB=AMU*SIN(ALPTB)
        CSALB=COS(ALPTB)
        AMCSB=AMU*CSALB
   38   IX1=NR*NBL
C
C       COMPUTE DISTANCES L SUB I J ACCORDING TO FORMULA PAGE III-3
        CSI=-DSI*DIRA+PSIRA
C
        DO 40 J=1,NA
        CSI=CSI+DSI*DIRA+PSIRA
   40   SINEJ(J)=SIN(CSI)
2001    CONTINUE
C       RBL(I) IS PETERS RBAR SUB I AND RWK(M,1) IS PETERS R SUB M
C
        DO 50 I=1,NR1
        BET(I)=BET(I)*CNVRT
        DO 50 J=1,NA
   50   EL(I,J)=DSI*(RBL(I)+AMCSA*SINEJ(J)*DIRA)
C
        IF(NBL-1) 143,143,144
  144   CSI=-DSI*DIRB+PSIRB
        DO 55 J=1,NA
        CSI=CSI+DSI*DIRB
   55   SINEBJ(J)=SIN(CSI)
        DO 59 I=NR2,NR
        BET(I)=BET(I)*CNVRT
        DO 59 J=1,NA
   59   EL(I,J)=DSI*(RBL(I)+AMCSB*SINEBJ(J)*DIRB)
C
143     CONTINUE
C
C       INITIALIZE COLUMN VECTOR OF UNKNOWNS FOR OVERALL ITERATION
        DO 100 J=1,IJ
  100   SAVE(J)=0.
C
C       ENTER OVERALL ITERATIVE SCHEME
```

```
C      CALL AERODYNAMIC PORTION (BLD III)
C    KTEST = 0 BYPASSES READ OF SIGMA AND MU
C
       NNR1=NR1
       CALL START (WBARA,AMU,ALPTA,BCA,ACA,AKIA,THETA,DSI,CSIA,
     1 BI,BET,RBL,OMSQA,1,NNR1,AVA(NMA,1),CSIDTA,RA,PSIRA,DIRA,
     1 HDOT,PHI,THET,NA,NRNT)
       EMTA1=EMTA
       RA1=RA
C
C      IF ONLY ONE ROTOR IS USED GO TO 83
       IF(NBL-1)83,83,84
   84  NNR=NR
       NNR2=NR2
       CALL START (WBARB,AMU,ALPTB,BCB,ACB,AKIB,THETB,DSI,CSIB,
     1 BI,BET,RBL,OMSQB,NNR2,NNR,AVB(NMB,1),CSIDTB,RB,PSIRB,DIRB,
     1 HDOT,PHI,THET,NA,NRNT)
C
       EMTB1=EMTB
       RB1=RB
       CT2=CTB
       ALPT2=ALPTB
    83 CONTINUE
       ZY=RA
       ZQ=RA
C      WRITE(6,20)
20     FORMAT(   6HOHDOT ,//)
C      WRITE(6,200)((HDOT (I,J),J=1,NA),I=1,NR)
200    FORMAT(10(1X,E12.5),/)
C       WRITE(6,21)
21     FORMAT(   6HOPHI  ,//)
C      WRITE(6,200)((PHI  (I,J),J=1,NA),I=1,NR)
C      WRITE(6,22)
22     FORMAT(   6HOTHET ,//)
C      WRITE(6,200)((THET (I,J),J=1,NA),I=1,NR)
85     CONTINUE
       CT1=CTA
       ALPT1=ALPTA
       GO TO (2004,2000),ISW
 2004  CALL BLD3
       GO TO 2003
2000   CONTINUE
       ISW=1
        READ(2)HDOT
        READ(2)PHI
        READ(2)THET
        READ(2)FORCZ
        READ(2)FORCX
        READ(2)EMOME
        READ(2)SAVE
```

```
        READ(2)EL
        READ(2)RBL
        READ(2)BI
        READ(2)BET
        READ(2)IT3
        READ(2)SV3
        READ(2)ERRSV
        READ(2)UNKWN
2003    CONTINUE
        WRITE (6,7994)
 7994   FORMAT (6HOFORCZ//)
        WRITE (6,200) ((FORCZ(I,J),J=1,NA),I=1,NR)
        WRITE (6,7995)
 7995   FORMAT (6HOFORCX//)
        WRITE (6,200) ((FORCX(I,J),J=1,NA),I=1,NR)
        WRITE (6,7996)
 7996   FORMAT (6HOEMOME//)
        WRITE (6,200) ((EMOME(I,J),J=1,NA),I=1,NR)
C
C ASSIGN CORRECT FORCES FOR INPUT TO BLD4
C
        DO 93 I=1,NR1
        DO 93 J=1,NA
        WFX(I,J)=FORCX(I,J)
        WFZ(I,J)=FORCZ(I,J)
        WEMO(I,J)=EMOME(I,J)
93      CONTINUE
        DO 931 I=1,NR1
931     RBLA(I)=RBL(I)
C
C ENTER BLD4 WITH PARAMETERS FOR FIRST ROTOR
        DO 2345 I=1,114
 2345   CSIX(I)=CSIA(I)
C
        CALL BLD4
C
C ASSIGN CALCULATED VALUES FROM BLD4 TO CORRECT ARRAYS
        DO 2346 I=1,114
 2346   CSIA(I)=CSIX(I)
        DO 94 I=1,NR1
        DO 94 J=1,NA
        HDOT(I,J)=WDOT(I,J)
        PHI(I,J)=WPHI(I,J)
        THET(I,J)=WTHET(I,J)
94      CONTINUE
C IF ONLY ONE ROTOR IS USED GO TO 85
        IF (NBL-1) 9411,9411,86
C
C ASSIGN CORRECT FORCES FOR INPUT TO BLD4
86      DO 95 I=NR2,NR
```

```
      DO 95 J=1,NA
      L=I-NR1
      WFX(L,J)=FORCX(I,J)
      WFZ(L,J)=FORCZ(I,J)
      WEMO(L,J)=EMOME(I,J)
95    CONTINUE
      DO 951 I=NR2,NR
      L=I-NR1
951   RBLB(L)=RBL(I)
C
C ENTER BLD4 WITH PARAMETERS FOR SECOND ROTOR
      DO 2347 I=1,114
 2347 CSIX(I)=CSIB(I)
      DO 2387 I=1,1634
      TEMPX=DUMSDA(I)
      DUMSDA(I)=DUMSDB(I)
 2387 DUMSDB(I)=TEMPX
      DO 2386 I=1,2095
      TEMPX=DUMBD4(I)
      DUMBD4(I)=DOMBD4(I)
 2386 DOMBD4(I)=TEMPX
      CALL BLD4
C
C ASSIGN CALCULATED VALUES FROM BLD4 TO CORRECT ARRAYS
      DO 2385  I=1,2095
      TEMPX=DOMBD4(I)
      DOMBD4(I)=DUMBD4(I)
 2385 DUMBD4(I)=TEMPX
      DO 2388 I=1,1634
      TEMPX=DUMSDA(I)
      DUMSDA(I)=DUMSDB(I)
 2388 DUMSDB(I)=TEMPX
      DO 2348 I=1,114
 2348 CSIB(I)=CSIX(I)
      I1=0
      DO 941 I=NR2,NR
      I1=I1+1
      DO 941 J=1,NA
      HDOT(I,J)=WDOT(I1,J)
      PHI(I,J)=WPHI(I1,J)
      THET(I,J)=WTHET(I1,J)
941   CONTINUE
9411  CONTINUE
      IF (IT3.EQ.1000) STOP
      GO TO 85
  899 FORMAT(80H
     1                                        )
  900 FORMAT (16I5)
  901 FORMAT (8F10.0)
9212  FORMAT(3(/),48X,37HDYNAMIC RESPONSE OF HELICOPTER BLADES,5(/))
```

```
920   FORMAT(1H ,51X,18HNUMBER OF ROTORS= ,I3,/,
     1 52X,18HNUMBER OF BLADES= ,I3,/,
     3 40X,30HNUMBER OF RADIAL LOAD POINTS= ,I3,/,
     4 39X,31HNUMBER OF AZIMUTHAL POSITIONS= ,I3,/,
     5 47X,23HNUMBER OF WAKE POINTS= ,I3,/,
     6 41X,28H ROTATIONAL SPEED CAP OMEGA= ,F9.4,8H RAD/SEC ,/,
     7 53X,17HAIR DENSITY RHO= ,F8.5,/,
     8 52X,18HADVANCE RATIO MU= ,F7.3)
  924 FORMAT(32X,37HCONVERGENCE ON TWO INNER ITERATIONS =E8.1/
     1       37X,32HCONVERGENCE ON OUTER ITERATION =E8.1,2(/))
9211  FORMAT(1HO,48X,29HBLADE PROPERTIES- FIRST BLADE, 3(/))
9215  FORMAT(1HO,48X,30HBLADE PROPERTIES- SECOND BLADE,3(/))
      END
```

```
      SUBROUTINE INPUT(BI,BET,ISEC,NRPT,ELNTH,EIX,EIY,XINR,YINR,EMAS,
     1 DPHI,EPS,DLZ,ZA,ALPHT,CT,EMT,OMSQ,AKI,THETO,XROOT,AKL,R,AC,BC,
     2 AT,AMZ,AVY,AMY,AVZ,AV,AW,APHI,ASI,ATHET,SIG,OMEGA,RBL,EIZ,
     3 RWK,I1,I2,NM,PSIR,DIR,NA,NR1,NRNT,NMAS,NMODE,NC,DAMPC,SIGKJ)
C
C
C     INPUT READS IN BLADE PROPERTIES
C
C
      DIMENSION RBL(8)
      DIMENSION SIGKJ(6,6)
      DIMENSION BI(8),BET(8),ELNTH(20),EIX(20),EIY(20),XINR(20),YINR(20)
     1,EMAS(20),DPHI(20),EPS(20),DLZ(20),ZA(20),OMEGA(6),EIZ(20),SIG(6)
      DIMENSION AMY(20,6),AV(20,6),AW(20,6),APHI(20,6),ASI(20,6),ATHET(2
     10,6),AT(20,6),AMZ(20,6),AVY(20,6),AVZ(20,6)
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPCMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      READ(IN,901) ALPHT,CT,EMT,OMSQ,AKI,RWK
      READ (IN,901) DAMPC
      READ(IN,901) THETO,XROOT,AKL,R,AC,BC,PSIR,DIR
      READ(IN,901) (BI(I),I=I1,I2)
      READ(IN,901) (BET(I),I=I1,I2)
      DO 16 I=1,NM
      READ (IN,900) ISEC,NRPT
C     BLADE PROPERTIES REPEATED AUTOMATICALLY IF NRPT=1
      IF(NRPT) 15,10,15
   10 READ(IN,902) ELNTH(I),EIX(I),EIY(I),EIZ(I),XINR(I),YINR(I),
     1              EMAS(I),DPHI(I),EPS(I),DLZ(I),ZA(I)
      GO TO 16
   15 IM1=I-1
      EIX(I)=EIX(IM1)
      EIY(I)=EIY(IM1)
      EIZ(I)=EIZ(IM1)
      YINR(I)=YINR(IM1)
      ELNTH(I)=ELNTH(IM1)
      EMAS(I)=EMAS(IM1)
      XINR(I)=XINR(IM1)
      EPS(I)=EPS(IM1)
      DLZ(I)=DLZ(IM1)
      DPHI(I)=DPHI(IM1)
      ZA(I)=ZA(IM1)
   16 CONTINUE
   21 READ(IN,901) (SIG(K),K=1,MAXMO)
      DO 22 K=1,MAXMO
       READ(5,588)OMEGA(K)
      DO 22 I=1,NM
C       THIS READ SHOULD BE REPLACED WITH A IT8 UNFORMATED READ
      READ(5,588)AV(I,K),AW(I,K),APHI(I,K),ASI(I,K),ATHET(I,K),
```

```
     1 AT(I,K),AMZ(I,K),AVY(I,K),AMY(I,K),AVZ(I,K)
       AW(I,K)=-AW(I,K)
C      AVY(I,K)=-AVY(I,K)
   22 CONTINUE
      DO 24 K=1,MAXMO
      TWK=2.*OMEGA(K)
      TDA=DAMPC*APHI(1,K)
      DO 23 J=1,MAXMO
      SIGKJ(K,J)=TDA*APHI(1,J)
   23 CONTINUE
      TSIG=SIG(K)
      SIG(K)=TSIG+SIGKJ(K,K)/TWK
      SIGKJ(K,K)=-TWK*TSIG
   24 CONTINUE
C     REPLACE THIS READ WITH IT7
      READ(5,589)(RBL(I),I=I1,I2)
589   FORMAT(10F8.0)
588   FORMAT(5(G12.7,3X),/,5G15.7)
      RETURN
  900 FORMAT (16I5)
  901 FORMAT (8F10.7)
  902 FORMAT(10E8.7)
      END
```

```
      SUBROUTINE OUTPUT(THETO,XROOT,AKL,AC,BC,R,CT,ALPHT,EMT,
     1AKI,OMSQ,BI,BET,ELNTH,EMAS,EIX,EIZ,XINR,YINR,DPHI,
     2 EPS,DLZ,ZA,OMEGA,SIG,AV,AW,APHI,ASI,ATHET,AT,AMZ,AVY,AMY,AVZ,
     3 RBL,EIY,RWK,I1,I2,NM,PSIR,DIR,NR1,NMAS,NMODE,NC,DAMPC,SIGKJ)
C
C
C     PRINTS OUT BLADE PROPERTIES
C
C
      DIMENSION RBL(8)
      DIMENSION SIGKJ(6,6)
      DIMENSION BI(8),BET(8),ELNTH(20),EIX(20),EIY(20),XINR(20),YINR(20)
     1,EMAS(20),DPHI(20),EPS(20),DLZ(20),ZA(20),OMEGA(6),EIZ(20),SIG(6)
      DIMENSION AMY(20,6),AV(20,6),AW(20,6),APHI(20,6),ASI(20,6),ATHET(2
     10,6),AT(20,6),AMZ(20,6),AVY(20,6),AVZ(20,6)
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      WRITE(NOUT,9221) THETO,XROOT,AKL,AC,BC,R,NM,PSIR,DIR
      WRITE (NOUT,930) CT,ALPHT,EMT,AKI,OMSQ,RWK,DAMPC
      WRITE(NOUT,932)(BI(I),I=I1,I2)
      WRITE(NOUT,933)(BET(I),I=I1,I2)
      WRITE(NOUT,922)
      WRITE(NOUT,925)(I,ELNTH(I),EIX(I),EIY(I),EIZ(I),XINR(I),YINR(I),
     1 EMAS(I),DPHI(I),EPS(I),DLZ(I),ZA(I),I=1,NM)
      WRITE (NOUT,927)
      DO 5 K=1,MAXMO
      WRITE (NOUT,929) (SIGKJ(K,J),J=1,MAXMO)
    5 CONTINUE
      WRITE(NOUT,935)
      DO 27 K=1,MAXMO
      WRITE(NOUT,936) K,OMEGA(K),SIG(K),(I,AV(I,K),AW(I,K),APHI(I,K),
     1                ASI(I,K),ATHET(I,K),I=1,NM)
   27 CONTINUE
      WRITE(NOUT,9351)
      DO 2711 K=1,MAXMO
      WRITE(NOUT,9361)(I,AT(I,K),AMZ(I,K),AVY(I,K),AMY(I,K),
     1 AVZ(I,K),I=1,NM)
2711  CONTINUE
      WRITE(NOUT,940)(RBL(I),I=I1,I2)
      RETURN
  940 FORMAT (2(/),56X,17HBLADE POINT RADII/(6E20.5))
9361  FORMAT(13X,I4,5E20.7)
9351  FORMAT(5(/),13X,8H SECTION,10X,1HT,19X,2HMZ,17X,3H-VY,18X,2HMY,
     1 18X,2HVZ)
  935 FORMAT (1H1 ,59X,12HNORMAL MODES/13X,7HSECTION,10X,1HV,19X,1HW,
     118X,3HPHI,18X,2HSI,16X,5HTHETA)
  925 FORMAT (1X,I3,3X,F7.4,3X,10E11.4)
  922 FORMAT(3(/),57X,16HBLADE PROPERTIES//1X,3H  I,4X,6HLENGTH,6X,
     1 3HEIX,8X,3HEIY,8X,3HEIZ,8X,2HIX,09X,2HIY,9X,4HMASS,7X,9HDELTA PHI
```

```
     2 ,2X,7HEPSILON,4X,8HDELTA LZ,4X,2HZA/9X,4HFEET,7X,3(6HLB-FT2,5X),
     3 3(7HLB-SEC2,4X),7HDEGREES,4X,3(4HFEET,7X),/,
     4 42X,3(4HFEET,7X),2(/))
  930 FORMAT(                                        46X,23HTHRUST COEFFICIENT CT
     1=F9.5/43X,26HSHAFT AXIS ANGLE ALPHA S =F7.2,8H DEGREES/
     249X,20HTIP MACH NUMBER MT =F9.4/64X,5HK I =F7.3/54X,15HOMEGA SQUAR
     3ED =F8.4,/,50X,19HBLADE POINT RADIUS=,F9.5,/,
     4 49X,20HDAMPING COEFFICENT=,F9.5)
  933 FORMAT (/30X,30HBLADE TWIST ANGLES IN DEGREES 3F10.5/(60X,3F10.5))
 9221  FORMAT(1H ,36X,33HNOMINAL PITCH ANGLE, THETA ZERO = ,F8.3,
     1 8H DEGREES ,/, 39X,30HOFFSET OF FLAP HINGE, X ROOT = ,F9.4,
     2 5H FEET ,/,
     3 40X,29H LAG DAMPING COEFFICIENT KL = ,F9.2,15H FT LBS/RAD/SEC ,/,
     4 41X,28HCYCLIC PITCH AMPLITUDES AC = ,F7.3,8H DEGREES ,/,
     5 65X,4HBC = ,F7.3,8H DEGREES, /,
     6 53X,16HROTOR RADIUS R = ,F7.3,5H FEET,/,
     2 47X,23HNUMBER OF MASS POINTS= ,I3,/
     345X,24HROTOR REFERENCE ANGLE = ,F7.3, 8H DEGREES/
     4 42X,27HROTOR ROTATION DIRECTION = ,F3.0/)
  932 FORMAT (/41X,19HSEMI CHORD LENGTHS 3F10.5/(60X,3F10.5))
  936 FORMAT (29X,4HMODE,I3,4X,9HFREQUENCY,F10.6,12H RADIANS/SEC,5X,
     2      15HDAMPING SIGMA =F10.6/(13X,I4,5E20.7))
  927 FORMAT (/10X,17HSIGKJ(K,J) MATRIX/)
  929 FORMAT (2X,15F8.4)
      END
```

```
      SUBROUTINE CONV(CNVRT,THETO,AC,BC,ALPHT,DPHI,NM,NMAS)
      DIMENSION DPHI(20)
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      THETO=THETO*CNVRT
      AC=AC*CNVRT
      BC=BC*CNVRT
      ALPHT=ALPHT*CNVRT
      DO 30 I=1,NM
   30 DPHI(I)=DPHI(I)*CNVRT
      RETURN
      END
```

```
      FUNCTION WBAR(AMU,ALPHT,CT)
C     WBAR RETURNS EITHER WBARA OR WBARB FOR USE IN SUBROUTINE START
      AMUCS=AMU*COS(ALPHT)
      AMCS2=AMUCS*AMUCS
      SQT=SQRT(AMCS2*AMCS2+CT*CT)
      WBAR=SQRT(.5*(SQT-AMCS2))
      RETURN
      END
```

```
      SUBROUTINE START (WBARX,AMU,ALPHT,BC,AC,AKI,THET0,DSI,CSI,
     1 BI,BET,RBL,OMSQ,L,M,AV,CSIDT,R,PSIR,DIR,HDOT,PHI,THET,NA,NRNT)
      DIMENSION BI(08),BET(08),RBL(08)
      DIMENSION HDOT(08,18),PHI(08,18),THET(08,18),CSIDT(1,18)
      DIMENSION CSI(6,19)
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
C     START SETS INITIAL VALUES FOR HDOT, PHI, THET
97     CONTINUE
1     CONTINUE
      FNA=NA
      HOLD=WBARX+AMU*ALPHT
      AMUSQ=AMU*AMU
      A0=(.25*(1.+AMUSQ)*THET0+AMU*AC/3.-.25*AMU*BC*BI(L)-HOLD/3.+
     1 BET(M )*(.2+AMUSQ/6.))*AKI
      AI=((.25+.125*AMUSQ)*BC+AC*BI(L)/3.)*AKI
      BCIR=(2.*AMU*THET0/3.+(.25+.375*AMUSQ)*AC-BC*BI(L)/3.-.5*AMU*HOLD+
     1 .5*AMU*BET(M ))*AKI
      C0=.25*AKI
      C1=AMU*AKI/3.
      D0=OMSQ-BI(L)*AKI/3.
      D1=-.5*AMU*BI(L)*AKI
      D2=.25*AMUSQ*AKI
      ETA1=((D0-1.)*(BCIR*D0-D1*A0)-(AI*D0-C1*A0)*(.5*D2-C0))/
     1 ((C0-1.)*(D0*(D0-1.)-.5*D1*D1)-(.5*D2-C0)*(D0*(.5*D2+C0)
     2 -.5*D1*C1))
      CSI1=(AI-C1*A0/D0-(.5*D2+C0-D1*C1/(2.*D0))*ETA1)/(D0-1.)
      CSI0=(A0-.5*D1*ETA1)/D0
      WRITE (6,200) ETA1,CSI1
      DO 100 I=L,M
      SI=-DSI*DIR+PSIR
      DO 100 J=1,NA
      SI=SI+DSI*DIR
      SN=SIN(SI)
      CS=COS(SI)
      HDOT(I,J)=RBL(I)*(CSI1*SN-ETA1*CS)+.5*BI(L)*((AC-CSI1)*CS
     2 -(BC+ETA1)*SN)-AMU*CS*(CSI0+CSI1*CS+ETA1*SN)
      THET(I,J)=THET0+AC*SN+BC*CS
      PHI(I,J)=CSI0+CSI1*CS+ETA1*SN
      IF (I.EQ.1.OR.I.EQ.NR2) CSIDT(1,J)=CPOMG*R*(CSI1*SN-ETA1*CS)/AV
      IF (I.EQ.1.OR.I.EQ.NR2) CSI(1,J)=-R*PHI(I,J)/AV
      HDOT(I,J)=0.
      PHI(I,J)=0.
      CSI(1,J)=0.
      CSIDT(1,J)=0.
100   CONTINUE
      WRITE (6,20)
20    FORMAT(   6HOHDOT ,//)
      WRITE(6,200)((HDOT (I,J),J=1,NA),I=L,M)
200   FORMAT(10(1X,E12.5),/)
```

```
      WRITE(6,21)
21    FORMAT(   6HOPHI   ,//)
      WRITE(6,200)((PHI  (I,J),J=1,NA),I=L,M)
      WRITE(6,22)
22    FORMAT(   6HOTHET  ,//)
      WRITE(6,200)((THET (I,J),J=1,NA),I=L,M)
      WRITE(6,23)
23    FORMAT (6HOCSIDT//)
      WRITE (6,9876)  (CSIDT(1,J),J=1,NA)
9876  FORMAT (1HO,6E15.7)
      RETURN
      END
```

```
      SUBROUTINE HARMN (NM,NA,DSI,F,MTYP,NMAS,NROW,NCOL)
      DIMENSION A(20,10),B(20,10),F(NROW,NCOL)
C
      WRITE(6,900)
      NCNT=0
      NAP1=NA+1
      NA02=NA/2
      N5=NA02/5
      N5T10=(NA02*10)/5
      IF(N5*10-N5T10)  1000,1010,1000
 1000 N5=N5+1
 1010 CONTINUE
      ANA=NA
      T=2./ANA
      GO TO (1,2,3,4,5,6),MTYP
    1 WRITE(6,901)
      GO TO 6
    2 WRITE(6,902)
      GO TO 6
    3 WRITE(6,903)
      GO TO 6
    4 WRITE(6,904)
      GO TO 6
    5 WRITE(6,905)
    6 CONTINUE
      DO 240 L=1,NM
      DO 200 N=1,NA02
      A(L,N) = 0
      B(L,N) = 0
      M=N-1
      DO 100 K=1,NA
      AKDSI=(K-1)*M*DSI
      C=COS(AKDSI)
      S = SIN(AKDSI)
       A(L,N)=A(L,N)+C*F(L,K)
      B(L,N)=B(L,N)+S*F(L,K)
  100 CONTINUE
      B(L,N) = T*B(L,N)
  200 A(L,N) = T*A(L,N)
  240 A(L,1) = .5*A(L,1)
  270 CONTINUE
      NG2=0
      NGRP=0
      WRITE(6,906) (N,N=1,5)
      WRITE(6,908)
  210 NGRP=NGRP+1
      NG1=NG2+1
      NG2=5*NGRP+1
      IF(NGRP-N5) 220,215,300
  215 NG2=NA02
```

```
  220 IF(NGRP-1) 250,250,260
  250 DO 255 I=1,NM
  255 WRITE(6,910) I,A(I,1),(A(I,N),B(I,N),N=2,NG2)
      GO TO 210
260    NX1=NG1-1
      NX2=NG2-1
      IF (NG1.EQ.NG2) GO TO 300
      WRITE(6,907)(N,N=NX1,NX2)
      WRITE(6,909)
      DO 265 I=1,NM
  265 WRITE(6,911) I,(A(I,N),B(I,N),N=NG1,NG2)
      GO TO 210
  300 CONTINUE
      IF (NCNT.GT.0) RETURN
      DO 86 I=1,NM
      DO 86 N=1,NAO2
      U=A(I,N)
      V=B(I,N)
   80 G=SQRT(U*U+V*V)
      ARG=0.
      IF(  G  .EQ.0.)GO TO 85
      ARG=ATAN2(V,U)
   85 B(I,N)=ARG
   86 A(I,N)=G
      WRITE (6,920)
  920 FORMAT (1H ,///,51X,29HHARMONIC ANALYSIS, POLAR FORM)
      NCNT=NCNT+1
      IF (NCNT.LT.2) GO TO 270
900   FORMAT(1H ,3(/),57X,17HHARMONIC ANALYSIS)
  901 FORMAT(3(/),60X,11HLEFT TORQUE)
  902 FORMAT(3(/),59X,13HLEFT Z MOMENT)
  903 FORMAT(3(/),59X,13HLEFT Y SHEAR )
  904 FORMAT(3(/),59X,13HLEFT Y MOMENT)
  905 FORMAT(3(/),59X,13HLEFT Z SHEAR )
  906 FORMAT(2(/),8X,5HN = 0,3X,5(9X,3HN =I2,8X))
  907 FORMAT(2(/),8X,5(17X,3HN =I2))
908   FORMAT(5H  I  ,5X,1HA,5(11X,1HA,10X,1HB))
909   FORMAT(5H  I  ,6X,5(11X,1HA,10X,1HB))
910   FORMAT(I3,E11.4,5(1X,2E11.4))
911   FORMAT(I3,11X,5(1X,2E11.4))
      RETURN
      END
```

```
      SUBROUTINE BLD3
C
C
C
C     THIRD PROGRAM IN BLADE LOADS SEQUENCE
C     SOLUTION OF EQUATIONS BY ITERATION
C
      DIMENSION USAVE(144),CO(08,18),CO1(08,18),SIGMA(144)
      DIMENSION UNKWN(144),SV3(144),RBL(08),FORCX(08,18),FORCZ(08,18),
     1 EMOME(08,18),HDOT(08,18),PHI(08,18),THET(08,18),BI(08),
     2 BET(08),SAVE(144),WBR(08,18),EL(08,18)
C
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLAD3/ CMPD,AMNA2,AMNB2,IT2,ON2DS,IT1,CMPAR
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /BLD3X1/ ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,UNKWN,SV3,RBL,
     1 FORCZ,EMOME,HDOT,PHI,THET,BI,BET,SAVE,WBR,EL,FORCX
C
      NANR=NA*NR
      NRNA=NR*NA
971   CONTINUE
3396  IF(IT3-1)3398,3398,3399
3398   DO 3397 I=1,NANR
3397  SV3(I)=0.0
3399   CONTINUE
370   IF(KTEST)373,373,372
  373 DO 3333 M=1,NRNA
      SIGMA(M)=0.
 3333 CONTINUE
372   CONTINUE
      IF(KTEST)1,1,2
2     REWIND IT7
      REWIND IT8
  1   CONTINUE
C
C     DEFINE NECESSARY CONSTANTS
   20 ONOTP=1./TWOPI
      FNA=NA
      PIO2=PI*.5
      ON2DS=0.5/DSI
      AMNA2=AMSNA*AMSNA*DSI*DSI
      AMNB2=AMSNB*AMSNB*DSI*DSI
      IT1=0
      IT2=1
      IF(IT3-1)40,40,50
```

```
40      IF(IT2-1)41,41,50
41      IF(KTEST)42,42,43
   43 READ (IT8) (UNKWN(K),K=1,NANR)
      DO 57 K=1,NA
      M=(K-1)*NR
      DO 57 J=1,NR
      I=M+J
   57 UNKWN(I)=UNKWN(I)/BI(J)
      GO TO 50
42      DO 44 I=1,IJ
44      UNKWN(I)=.01
   50 CALL GAMAS (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL)
      CMPD=0.0
      CMPAR=0.0
C     CHECK FOR CONVERGENCE ON OVERALL ITERATION
      DO 302 INDEX=1,IJ
      CMPAR=CMPAR+(UNKWN(INDEX)-SV3(INDEX))**2
      CMPD=CMPD+(UNKWN(INDEX))**2
302   SV3(INDEX)=UNKWN(INDEX)
      EPLON=SQRT(CMPAR/CMPD)
      IF (IT3-3) 3028,3026,3021
C     CUTOFF ON OVERALL ITERATION FOR SEVERE DIVERGENCE
3021  IF(EPLON-ERRSV)3026,3026,303
3026  ERRSV=EPLON
3028  WRITE(NOUT,940)IT3,EPLON
      IF(EPLON-ALL2)303,303,310
303   IT3=1000
      WRITE(NOUT,940)IT3,EPLON
310   CONTINUE
      IT3=IT3+1
      IF (IT3.GT.NIT3) IT3=1000
      IF (IT3.EQ.1000) CALL HARMN (NR,NA,DSI,WBR,6,0,8,18)
      CALL BLD3B (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL,INDEX)
      RETURN
  940  FORMAT(2(/),30X,5HIT3 =,I5,30X,7HERROR =,E15.7/)
       END
```

```
      SUBROUTINE GAMAS (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL)
C
      DIMENSION USAVE(144),CO(08,18),CO1(08,18),SIGMA(144)
      DIMENSION UNKWN(144),SV3(144),RBL(08),FORCX(08,18),FORCZ(08,18),
     1EMOME(08,18),HDOT(08,18),PHI(08,18),THET(08,18),BI(08),
     2BET(08),SAVE(144),WBR(08,18),EL(08,18)
C
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLAD3/ CMPD,AMNA2,AMNB2,IT2,ON2DS,IT1,CMPAR
      COMMON /SGSLS/ SIGLM,WBRLM
      COMMON/PRNT/NPRNT
C
C     DO LOOPS TO COMPUTE CERTAIN COEFFICIENTS AS FUNCTIONS OF RADIAL
C     POSITION AND AZIMUTH WHICH APPEAR IN EXPRESSIONS FOR UNKNOWNS,
C    PAGE III-5.
      NR1=NR2-1
      NANR=NA*NR
      NANB=NA/NB
   50 DO 100 JJ=1,NANB
      DO 100 MROT=1,NBL
      DO 100 K=1,NB
      J=(JJ+(K-1)*NANB)
      JM1=J-1
      JP1=J+1
      IF(JM1) 55,55,60
   55 JM1=NA
      GO TO 65
   60 IF(JP1-NA) 65,65,62
   62 JP1=1
   65 DO 100 IM=1,NR1
      I=IM+(MROT-1)*NR1
      IF (I-NR1)3137,3137,3138
3137  AMUSN=AMSNA
      AMSN2=AMNA2
      EMT=EMTA1
      UUBAR=WBAR(AMU,ALPTA,CTA)
      GO TO 3139
3138  AMUSN=AMSNB
      AMSN2=AMNB2
      EMT=EMTB1
      UUBAR=WBAR(AMU,ALPTB,CTB)
3139  CONTINUE
      IF(KTEST)63,63,64
```

```
   64 READ (IT7) MSET,(SIGMA(M),M=1,NANR)
      SABS=ABS(SIGMA(MSET))
      SABS=SABS*SIGLM
      DO 20 M=1,NANR
      IF(M.EQ.MSET) GO TO 20
      SABSM=ABS(SIGMA(M))
      IF(SABSM.LT.SABS) GO TO 20
      SIGMA(M)=SIGMA(M)*SIGLM/SABSM
  20  CONTINUE
63     CONTINUE
      INDEX=(J-1)*NR+I
   73 INJM1=(JM1-1)*NR+I
      WBR(I,J)=0.
C     RECOMPUTE DOWNWASH VALUES ACCORDING TO PAGE III-4, IF IT2=1.
      DO 75 M=1,NR
      DO 75 N=1,NA
      IND=(N-1)*NR+M
   75 WBR(I,J)=WBR(I,J)-BI(M)*SIGMA(IND)*UNKWN(IND)
      WABS=ABS(WBR(I,J))
      IF(WABS.LT.WBRLM) GO TO 76
      WBR(I,J)=WBR(I,J)*WBRLM/WABS
  76  CONTINUE
C     COMPUTE U ACCORDING TO PAGE III-3 AND V ACCORDING TO PAGE III-4
   80 U=EL(I,J)/DSI
      V=HDOT(I,J)-AMUSN-WBR(I,J)
      SQTUV=SQRT(U*U+V*V)
      ARG=0.
      IF(SQTUV.EQ.0.)GO TO 85
      ARG=ATAN2(V,U)
   85 APHIJ=BET(I)+THET(I,J)+ARG
```

```
C      SUBROUTINE SERIES COMPUTES VALUES OF CLIFT=LIFT COEFFICIENT,
C      ASLOP=LIFT CURVE SLOPE, CMOME=MOMENT COEFFICIENT, CDRAG=DRAG
C      COEFFICIENT
       CALL SERIES (I,J,1,EMT,U,V,APHIJ,CLIFT,ASLOP,CMOME,CDRAG)
       UAPVC=U*ASLOP+V*CLIFT
       CO(I,J)=UAPVC/SQTUV
       CO1(I,J)=CLIFT*SQTUV+CO(I,J)*WBR(I,J)
C
C      INITIAL GAMMA VALUES ACCORDING TO FORMULA PAGE III-9
C      PETERS GAMMAS ARE UNKWN(INDEX) IN THIS PROGRAM
C
C      SKIP FOLLOWING WHEN IT3=1 AND USE INPUT FROM BLADE GEO
   95  IF(IT3-1)100,100,951
951    IF(IT2-1) 97,97,100
    97 INDEX=(J-1)*NR+I
       UNKWN(INDEX)=SQTUV*CLIFT+PI*BI(I)*((THET(I,JP1)-THET(I,JM1))
      1              *ON2DS+PHI(I,J))
   100 CONTINUE
       IF(NPRNT.LT.1) GO TO 1000
       WRITE(6,9999)((WBR(I,J),J=1,NA),I=1,NR)
  9999 FORMAT(/1HO,3HWBR/(7G14.6))
  1000 CONTINUE
       IF(KTEST) 110,110,102
102    REWIND IT7
       IF (IT3.EQ.1.AND.IT2.EQ.1.AND.KTEST.GT.0) RETURN
C
C      ENTER ITERATION SCHEME ON GAMMAS
C
   110 IT1=IT1+1
       IF(IT1-NIT1) 115,115,360
   115 CSI=-DSI
C
C      UPDATE VALUES OF GAMMAS ACCORDING TO EQ 3 PAGE III-5 IN  THE
C      FORM SHOWN ON PAGE III-5A
       DO 300 JJ=1,NANB
       DO 300 MROT=1,NBL
       DO 300 K=1,NB
       J=(JJ+(K-1)*NANB)
       DO 300 IM=1,NR1
       I=IM+(MROT-1)*NR1
C
       INDEX=(J-1)*NR+I
        IF(KTEST) 106,106,107
107    CONTINUE
       READ (IT7) MSET,(SIGMA(LM),LM=1,NANR)
         SABS=ABS(SIGMA(MSET))
       SABS=SABS*SIGLM
       DO 21 M=1,NANR
       IF(M.EQ.MSET) GO TO 21
       SABSM=ABS(SIGMA(M))
```

```
      IF(SABSM.LT.SABS) GO TO 21
      SIGMA(M)=SIGMA(M)*SIGLM/SABSM
   21 CONTINUE
  106 CONTINUE
      DENOM=1.-BI(I)*(CO(I,J)*SIGMA(INDEX))
      IF (ABS(DENOM)-1.E-06) 120,120,125
  120 WRITE(NOUT,979)
      GO TO 300
  125 USAVE(INDEX)=UNKWN(INDEX)
      SUM1=0.
      DO 150 LM=1,NR
      DO 146 LN=1,NA
      IND=(LN-1)*NR+LM
      IF(IND-INDEX) 145,146,145
  145 SUM1=-BI(LM)*SIGMA(IND)*UNKWN(IND)+SUM1
  146 CONTINUE
  150 CONTINUE
      JM1=J-1
      IF(JM1) 155,155,160
  155 JM1=NA
  160 INJM1=(JM1-1)*NR+I
  165 JP1=J+1
      IF(JP1-NA) 200,200,170
  170 JP1=1
  200 UNKWN(INDEX)= CO1(I,J)+PI*BI(I)*((THET(I,JP1)-THET(I,JM1))*ON2DS
     1    +PHI(I,J))-CO(I,J)*SUM1
      UNKWN(INDEX)=UNKWN(INDEX)/DENOM
  300 CONTINUE
C     END UPDATE OF GAMMAS
      IF(KTEST)304,304,113
  113 REWIND IT7
C     CHECK FOR CONVERGENCE ON INNERMOST ITERATION
  304 CMPAR=0.
      CMPD=0.
      DO 350 INDEX=1,IJ
      CMPD=CMPD+(UNKWN(INDEX))**2
  350 CMPAR=CMPAR+(UNKWN(INDEX)-USAVE(INDEX))**2
      EPLON=SQRT(CMPAR/CMPD)
      WRITE(NOUT,942) IT1,EPLON
      IF(EPLON-ALL1) 360,360,110
C
C
C     CHECK FOR CONVERGENCE ON SECOND LEVEL ITERATION
  360 IT1=C
      CMPAR=0.
      DO 379 INDEX=1,IJ
  379 CMPAR=CMPAR+(UNKWN(INDEX)-SAVE(INDEX))**2
      EPLON=SQRT(CMPAR/CMPD)
      WRITE(NOUT,941) IT2,EPLON
      IF(EPLON-ALL1) 500,500,380
```

```
  380 IF(IT2-NIT2) 390,390,500
  390 IT2=IT2+1
      DO 400 INDEX=1,IJ
  400 SAVE(INDEX)=UNKWN(INDEX)
      GO TO 50
500   CONTINUE
      RETURN
  979 FORMAT (//52X,16HDIVISION BY ZERO//)
  941  FORMAT(2(/),30X,5HIT2 =,I5,30X,7HERROR =,E15.7/)
  942  FORMAT(2(/),30X,5HIT1 =,I5,30X,7HERROR =,E15.7/)
      END
```

```
      SUBROUTINE BLD3B (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL,INDEX)
C
C
      DIMENSION USAVE(144),CO(08,18),CO1(08,18),SIGMA(144)
      DIMENSION UNKWN(144),SV3(144),RBL(08),FORCX(08,18),FORCZ(08,18),
     1EMOME(08,18),HDOT(08,18),PHI(08,18),THET(08,18),BI(08),
     2BET(08),SAVE(144),WBR(08,18),EL(08,18)
C
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLAD3/ CMPD,AMNA2,AMNB2,IT2,ON2DS,IT1,CMPAR
C
C     COMPUTATION FOR BLADE LOADS
C
      WRITE (6,9872) (UNKWN(INDEX),INDEX=1,IJ)
 9872 FORMAT (7H GAMMAS//(10G12.4))
      IF(IT3-1000)3121,896,3121
3121  DO 825 I=1,NR
      IF (I-NR1)3127,3127,3128
3127  AMUSN=AMSNA
      AMSN2=AMNA2
      EMT=EMTA1
      GO TO 3129
3128  AMUSN=AMSNB
      AMSN2=AMNB2
      EMT=EMTB1
3129  CONTINUE
      BO2DS=ON2DS*BI(I)
      TPI82=6.*PI*BO2DS*BI(I)
      DO 700 JC=1,NA
      J=JC
      JM1=JC-1
      IF(JM1) 5351,5351,545
 5351 JM1=NA
  545 INDEX=(JC-1)*NR+I
      INJM1=(JM1-1)*NR+I
  600 U=EL(I,JC)/DSI
      V=HDOT(I,JC)-AMUSN-WBR(I,JC)
      SQTUV=SQRT(U*U+V*V)
      ARG=0.
      IF (SQTUV.EQ.0.) GO TO 605
      ARG=ATAN2(V,U)
  605 APHIJ=BET(I)+THET(I,JC)+ARG
C     RECOMPUTE COEFFICIENTS FROM SERIES SUBROUTINE
```

```
      CALL SERIES (I,J,2,EMT,U,V,APHIJ,CLIFT,ASLOP,CMOME,CDRAG)
C     COMPUTE CAP I FUNCTIONS GIVEN ON PAGE III-8
      CO(I,JC)= 2.*BO2DS*CMOME*SQTUV
C     COMPUTE BLADE LOADS ACCORDING TO PAGE III-7
      EMOME(I,JC)=2.*CMOME*SQTUV*SQTUV
      FORCZ(I,JC)=(U*CLIFT+V*CDRAG)*SQTUV
      FORCX(I,JC)=(-V*CLIFT+U*CDRAG)*SQTUV
  700 CONTINUE
C     CORRECT BLADE LOADS FOR J+1 AND J-1 TERMS
      DO 800 J=1,NA
      JM1=J-1
      JP1=J+1
      IF(JM1) 702,702,705
  702 JM1=NA
      GO TO 710
  705 IF(J-NA) 710,707,707
  707 JP1=1
  710 INDEX=(J-1)*NR+I
      INJM1=(JM1-1)*NR+I
      INJP1=(JP1-1)*NR+I
      FORCZ(I,J)=FORCZ(I,J)+BO2DS*(UNKWN(INJP1)-UNKWN(INJM1))
      FORCZ(I,J)=FORCZ(I,J)+CO(I,JP1)-CO(I,JM1)
      EMOME(I,J)=EMOME(I,J)-.25*BO2DS*(UNKWN(INJP1)-UNKWN(INJM1))
      EMOME(I,J)=EMOME(I,J)-TPIB2*.25*ON2DS*(THET(I,JP1)-2.*THET(I,J)
     1           +THET(I,JM1))-TPIB2/16.*(PHI(I,JP1)-PHI(I,JM1))
  800 CONTINUE
  825 CONTINUE
      CALL BLD3C (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL,INDEX)
      RETURN
  896 CALL BLD3C  (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL,INDEX)
      RETURN
      END
```

```
      SUBROUTINE BLD3C (AMU,ALPTA,ALPTB,CTA,CTB,NRNTNA,NRNT,NA,NR1,
     1 USAVE,UNKWN,CO,CO1,SIGMA,SV3,RBL,FORCX,FORCZ,EMOME,HDOT,PHI,
     2 THET,BI,BET,SAVE,WBR,EL,INDEX)
C
C
      DIMENSION USAVE(144),CO(08,18),CO1(08,18),SIGMA(144)
      DIMENSION UNKWN(144),SV3(144),RBL(08),FORCX(08,18),FORCZ(08,18),
     1EMOME(08,18),HDOT(08,18),PHI(08,18),THET(08,18),BI(08),
     2BET(08),SAVE(144),WBR(08,18),EL(08,18)
C
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLAD3/ CMPD,AMNA2,AMNB2,IT2,ON2DS,IT1,CMPAR
      COMMON/PRNT/NPRNT
C
C     MODIFY LOADS FOR RESPONSE COMPUTATION ACCORDING TO PAGE IV-8
      ENTRY BLD3C1
      IF(NPRNT.LT.2) GO TO 9
896   WRITE(NOUT,980)
      DO 850 J=1,NA
      I=1
      WRITE(NOUT,8961)I,J,FORCZ(I,J),FORCX(I,J),EMOME(I,J)
8961  FORMAT(28X,2I5,3E20.7)
850   WRITE(NOUT,981)(I,FORCZ(I,J),FORCX(I,J),EMOME(I,J),I=2,NR)
981   FORMAT(28X,I5,5X,3E20.7)
   9   CONTINUE
      IF(IT3.EQ.1000) GO TO 891
      CPSQ=CPOMG*CPOMG
871   R=RA1
873   FAC  =ROAIR*CPSQ*R*R*R
      DO 87 M=1,NR
      FACTR=FAC  *BI(M)
      DO 87 N=1,NA
      TGARY=THET(M,N)+BET(M)
      EMOME(M,N)=(EMOME(M,N)-.5*(FORCZ(M,N)*COS(TGARY)+
     1 FORCX(M,N)*SIN(TGARY)))*FACTR*R*BI(M)
      FORCZ(M,N)=FORCZ(M,N)*FACTR
      FORCX(M,N)=FORCX(M,N)*FACTR
87    CONTINUE
      GO TO 890
C
C     FORMATS
 980  FORMAT(1H1///56X,14HLOADS ON BLADE//32X,1HI,4X,1HJ,10X,7HFORCE Z
     1        ,13X,7HFORCE X,13X,6HMOMENT/)
983   FORMAT(1H1,55X,20HFORCE IN Z DIRECTION   )
984   FORMAT(1H1,55X,20HFORCE IN X DIRECTION   )
```

```
985   FORMAT(1H1,57X,16HTORSIONAL MOMENT    )
C
  890 IF(IT3-1000) 895,891,891
  891 WRITE(NOUT,983)
      NAOVTO=NA/2+1
      NROW=8
      NCOL=18
      CALL HARMN(NR,NA,DSI,FORCZ,6,NMAS,NROW,NCOL)
      WRITE(NOUT,984)
      CALL HARMN(NR,NA,DSI,FORCX,6,NMAS,NROW,NCOL)
      WRITE(NOUT,985)
      CALL HARMN(NR,NA,DSI,EMOME,6,NMAS,NROW,NCOL)
  895 RETURN
      END
```

```
      SUBROUTINE SERIES(I,J,NCODE,EMT,U,V,APHIJ,CLIFT,ASLOP,CMOME,CDRAG)
C
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
C
C
```

```
C      SUBROUTINE TO COMPUTE CLIFT=LIFT COEFFICIENT
C                           ASLOP=LIFT CURVE SLOPE
C                           CMOME=MOMENT COEFFICIENT
C                           CDRAG=DRAG COEFFICIENT
C      FORMULAS TAKEN FROM CURVE FITS BY P.C.
C
C      INSERT HERE ANY COMPUTATIONS TO OBTAIN F,G, AND H COEFFICIENTS
C
       F=1.
       G=1.
       H=1.
C
       CLIFT=0.
       ASLOP=0.
       CMOME=0.
       CDRAG=0.
C
C
  180  NEG=1
       EMIJ=EMT*ABS(U)
       SQT=SQRT(1.-EMIJ*EMIJ)
       C1=1.-EMIJ
       C2=.22689*C1
   97  IF(APHIJ) 181,182,182
  181  APHIJ=-APHIJ
       NEG=-1*NEG
  182  IF(APHIJ-3.1415926) 184,184,183
  183  APHIJ=APHIJ-3.1415926*2.
       GO TO 97
  184  IF(APHIJ-C2) 185,187,187
  185  ASLOP=5.7296/SQT
       CLIFT=ASLOP*APHIJ
       CDRAG=.006+.13131*APHIJ*APHIJ
       CMOME=1.4324*APHIJ/SQT
       GO TO 250
  187  IF(APHIJ-.34906) 189,191,191
  189  CLIFT=.29269*C1+(1.3*EMIJ-.59)*APHIJ
       CMOME=CLIFT/(SQT*(.48868+.90756*EMIJ))
       C2=(.12217+.22689*EMIJ)*SQT
       CLIFT=CLIFT/C2
       ASLOP=(1.3*EMIJ-.59)
       GO TO 210
  191  IF(APHIJ-2.7402) 193,195,195
  193  S=SIN(APHIJ)
       S2=SIN(2.*APHIJ)
       S3=SIN(3.*APHIJ)
       S4=SIN(4.*APHIJ)
       CLIFT=(.080373*S+1.04308*S2-.011059*S3+.023127*S4)/SQT
       CMOME=(-.02827*S+.14022*S2-.00622*S3+.01012*S4)/SQT
       C=COS(APHIJ)
```

```
      C2=COS(2.*APHIJ)
      C3=COS(3.*APHIJ)
      C4=COS(4.*APHIJ)
      ASLOP=(.080373*C+2.08616*C2-.033177*C3+.092508*C4)/SQT
      CDRAG=(1.1233-.029894*C-1.00603*C2+.003115*C3-.091487*C4)/SQT
      GO TO 250
  195 IF(APHIJ-3.0020) 197,199,199
  197 CLIFT=-(.4704+.10313*APHIJ)/SQT
      ASLOP=-.10313/SQT
      CMOME=-(.4786+.02578*APHIJ)/SQT
      GO TO 210
  199 IF(APHIJ-3.1415926) 200,200,260
  200 CLIFT=(-17.550+5.5864*APHIJ)/SQT
      ASLOP=5.5864/SQT
      CMOME=(-12.5109+3.9824*APHIJ)/SQT
  210 CDRAG=(1.1233-.029894*COS(APHIJ)-1.00603*COS(2.*APHIJ)
     1       +.003115*COS(3.*APHIJ)-.091487*COS(4.*APHIJ))/SQT
  250 IF(NEG) 255,255,260
  255 CLIFT=-CLIFT
      CMOME=-CMOME
      APHIJ=-APHIJ
  260 CONTINUE
C
300   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BLD4
C
      DIMENSION CSIX(06,19)
      DIMENSION DXMBD4(2122)
      DIMENSION SIGKJ(6,6)
      DIMENSION ELNTH(20),EMAS(20),EIX(20),EIZ(20),RBL(09),
     1 XINR(20),DPHI(20),EIY(20),EPS(20),DLZ(20),ZA(20),
     2 YINR(20),OMEGA(20),SIG(20),BD(15)
C
      DIMENSION AV(20,06),AW(20,06),APHI(20,06),ASI(20,06),
     1 ATHET(20,06),AT(20,06),AMZ(20,06),AVY(20,06),
     2 AMY(20,06),AVZ(20,06),HDOT(04,18),PHI(04,18),
     3 THET(04,18),FX(04,18),FZ(04,18),EMO(04,18)
C
      DIMENSION VDOT(20,19),WDOT(20,19),PHIDT(20,19),
     1 SIDT(20,19),CSIDT(6,19)
C
      DIMENSION FV(20,19),FW(20,19),EMOME(20,19),
     1 CSI(6,19),CS2DT(6,19),CH(12,20),CCP(19),
     2 SSP(19),CCPS(20),SCPS(20),SUMA(20),H(20),X(20),
     3 RADIS(20),PHIV(20,19),SI(20,19),XCSIDT(1,18),
     4 THETA(20,19)
C
      DIMENSION CSPH(20),CAPHI(20),CC(19),SC(19),
     1 EX(19),EXMT(19),SN(05),SMLAZ(20),SMLAX(20),
     2 SMLAM(20),ALAM(19),SFZ(19),SFX(19),SFM(19),
     3 FORC(19)
C
      DIMENSION JM(4),JMP1(4),HDT(4),APH(4),ATH(4)
C
      EQUIVALENCE (DXMBD4(1),ELNTH(1)),(DXMBD4(21),EMAS(1)),(DXMBD4(41),
     1EIX (1)),(DXMBD4(61),RBL (1)),(DXMBD4(70),XINR(1)),(DXMBD4(90),DPH
     2I(1)),(DXMBD4(110),EIY (1)),(DXMBD4(130),EPS(1)),(DXMBD4(150),DLZ(
     31)),(DXMBD4(170),ZA (1)),(DXMBD4(190),YINR(1)),(DXMBD4(210),OMEGA(
     41)),(DXMBD4(69),PSIR ),
     4   (DXMBD4(230),SIG (1)),(DXMBD4(250),AV (1)),(DXMBD4(430),AW (1))
     5,
     5(DXMBD4(610),APHI (1)),(DXMBD4(790),ASI (1)),(DXMBD4( 970),ATHET(1
     6)),(DXMBD4(1150),AT (1)),(DXMBD4(1330),AMZ (1)),(DXMBD4(1510),AVY
     7(1)),(DXMBD4(1690),AMY (1)),(DXMBD4(1870),AVZ (1)),(DXMBD4(2050),B
     8D (1)),(DXMBD4(2065),RWK ),(DXMBD4(2066),CSAL ),(DXMBD4(2067),NM )
     8,
     9(DXMBD4(2068),XCSIDT(1,1))
      EQUIVALENCE (DXMBD4(2086),SIGKJ(1,1)),(DXMBD4(2122),DAMPC)
C
      COMMON/SAD3/CSIX
      COMMON /PUNCH/ NPCH
      COMMON /GCORD1/ THETO,CPSQ,DT,AC,BC,AKL,DTO2,R,ONOCP,CSALT
      COMMON /CONL1/ ENR,PIO2,XROOT,RAB,NMP1,NRP1,NAP1
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
```

```
     1 ,CPOMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLD4X1/ DXMBD4
      COMMON /BLD4X2/ HDOT,PHI,THET,FX,FZ,EMO,NMAS,NMODE,NAPSON,NR11
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /SAD1/ FV,FW,EMONE,CSI,CS2DT,CH,
     1CCP,SSP,CCPS,SCPS,SUMA,H,X,RADIS,PHIV,SI,THETA,CSPH,CAPHI,CC,SC,EX
     2,EXMT,SN,SMLAZ,SMLAX,SMLAM,ALAM,SFZ,SFX,SFM,FORC,JM,JMP1,HDT,APH,
     3ATH
      COMMON /SVSNR/ SNR
      COMMON /GARY/ NGOTO
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
C
C     BLADE DYNAMICS PROGRAM
      COMMON /CIR/PI,TWOPI,DIS
C
      NPAGE=1
C     SAVE NR  AND SET NEW NR FOR BLADE RESPONSE SUBPROGRAM
97     CONTINUE
      NAP1=NA+1
      DO 19 JA=1,NAP1
      DO 19 K=1,MAXMO
   19 CSI(K,JA)=CSIX(K,JA)
      SNR=NR
      NR=NR1
      NRP1=NR+1
C     DEFINE CONSTANTS
      R=BD(1)
      RAB=BD(14)
      THETO=BD(2)
      XROOT=BD(3)
      AKL=BD(4)
      AC=BD(5)
      BC=BD(6)
      ISEC=BD(7)
      NRPT1=BD(8)
      CT=BD(9)
      ALPHT=BD(10)
      EMT=BD(11)
      AKI=BD(12)
      OMSQ=BD(13)
C     INITIALIZE DYNAMIC EFFECTS TO ZERO IF IN FIRST OVERALL ITERATION
      IF(IT3-1000) 2941,2940,2940
2941  IF(IT3-2)10,10,20
   10 DO 15 JA=1,NAP1
      DO 12 I=1,NM
      VDOT(I,JA)=0.
      WDOT(I,JA)=0.
```

```
      PHIDT(I,JA)=0.
   12 SIDT(I,JA)=0.
      DO 14 K=2,MAXMO
   14 CSIDT(K,JA)=0.
   15 CONTINUE
      DO 21 JA=1,NA
   21 CSIDT(1,JA)=XCSIDT(1,JA)
   20 NMP1=NM+1
      FNA=NA
      ENR=NR
      ONOCP=1./(CPOMG*R)
      CAPT=TWOPI/CPOMG
      DT=CAPT/NA
      DTO2=DT*.5
      PIO2=.5*PI
      CSALT=CSAL
      CPSQ=CPOMG*CPOMG
      SFX(1)=0.
      SFZ(1)=0.
      SFM(1)=0.
      CAPHI(1)=THETO+DPHI(1)
      CSPH(1)=COS(DPHI(1))
      CCPS(1)=COS(CAPHI(1))
      SCPS(1)=SIN(CAPHI(1))
      H(1)=DLZ(1)*COS(THETO)
      RADIS(1)=XROOT+ELNTH(1)
C     COMPUTE TABLES OF COSINES AND OFFSET DISTANCES EXACTLY THE SAME AS
C     IN THE NATURAL FREQUENCY PROGRAM
      DO 40 I=2,NM
      IM1=I-1
C     CUMULATIVE PHI ANGLE ALONG BLADE
      CAPHI(I)=CAPHI(IM1)+DPHI(I)
      CCPS(I)=COS(CAPHI(I))
      SCPS(I)=SIN(CAPHI(I))
      CSPH(I)=COS(CAPHI(I)-THETO)
C     TOTAL DISTANCE TO RIGHT SIDE OF SECTION FROM CENTER OF ROTATION
      RADIS(I)=XROOT+ELNTH(I)
C     AVERAGE MASS USED IN COMPUTATION
   31 SUMA (IM1)=.5*(EMAS(IM1)+EMAS(I))
      H(I)=H(1)
      DO 35 J=1,IM1
      H(I)=H(I)+DLZ(J+1)*CCPS(J)
      RADIS(I)=RADIS(I)+ELNTH(J)
   35 CONTINUE
   40 CONTINUE
C     SINE AND COSINE OF OMEGA * T
      DO 45 JA=1,NAP1
      OMT=CPOMG*(JA-1.)*DT+PSIR
      CCP(JA)=COS(OMT)
   45 SSP(JA)=SIN(OMT)
```

```
      SUMA (NM)=.5*EMAS(NM)
      NGOTO=1
      CALL CONVL
      CALL GCOORD
      CALL RSPNS
      GO TO 2881
2940  NGOTO=2
      CALL RSPNS
      CALL SHEAR
      IF (NPCH.EQ.1) CALL RSPZZ
2881  CONTINUE
      DO 50 J=1,NAP1
      DO 50 K=1,MAXMO
 50   CSIX(K,J)=CSI(K,J)
      RETURN
      END
```

```
      SUBROUTINE CONVL
C
      REAL XX(06),FXX(06),FZZ(06),EM00(06)
C
      DIMENSION HDOT(04,18),PHI(04,18),THET(04,18),FX(04,18),FZ(04,18),
     1EM0(04,18)
      DIMENSION EIZ(20)
      DIMENSION VDOT(20,19),WDOT(20,19),PHIDT(20,19),SIDT(20,19),CSIDT
     1(06,19)
      DIMENSION FV(20,19),FW(20,19),EMOME(20,19),CSI(06,19),CS2DT(06,19)
     1,CH(12,20),CCP(19),SSP(19),CCPS(20),SCPS(20),SUMA(20),H(20),X(20),
     2RADIS(20),PHIV(20,19),SI(20,19),XCSIDT(1,18),THETA(20,19)
      DIMENSION CSPH(20),CAPHI(20),CC(19),SC(19),EX(19),EXMT(19),SN(05),
     1SMLAZ(20),SMLAX(20),SMLAM(20),ALAM(19),SFZ(19),SFX(19),SFM(19),FOR
     2C(19)
      DIMENSION JM(04),JMP1(04),HDT(04),APH(04),ATH(04)
C
      DIMENSION DXMBD4(2122)
      DIMENSION SIGKJ(6,6)
      DIMENSION ELNTH(20),EMAS(20),EIX(20),          RBL(04),
     1 XINR(20),DPHI(20),EIY(20),EPS(20),DLZ(20),ZA(20),
     2 YINR(20),OMEGA(20),SIG(20),BD(15)
C
      DIMENSION AV(20,06),AW(20,06),APHI(20,06),ASI(20,06),
     1 ATHET(20,06),AT(20,06),AMZ(20,06),AVY(20,06),
     2 AMY(20,06),AVZ(20,06)
C
      EQUIVALENCE (DXMBD4(1),ELNTH(1)),(DXMBD4(21),EMAS(1)),(DXMBD4(41),
     1EIX (1)),(DXMBD4(61),RBL (1)),(DXMBD4(70),XINR(1)),(DXMBD4(90),DPH
     2I(1)),(DXMBD4(110),EIY (1)),(DXMBD4(130),EPS(1)),(DXMBD4(150),DLZ(
     31)),(DXMBD4(170),ZA (1)),(DXMBD4(190),YINR(1)),(DXMBD4(210),OMEGA(
     41)),(DXMBD4(69),PSIR ),
     4   (DXMBD4(230),SIG (1)),(DXMBD4(250),AV (1)),(DXMBD4(430),AW (1))
     5,
     5(DXMBD4(610),APHI (1)),(DXMBD4(790),ASI (1)),(DXMBD4( 970),ATHET(1
     6)),(DXMBD4(1150),AT (1)),(DXMBD4(1330),AMZ (1)),(DXMBD4(1510),AVY
     7(1)),(DXMBD4(1690),AMY (1)),(DXMBD4(1870),AVZ (1)),(DXMBD4(2050),B
     8D (1)),(DXMBD4(2065),RWK ),(DXMBD4(2066),CSAL ),(DXMBD4(2067),NM )
     8,
     9(DXMBD4(2068),XCSIDT(1,1))
      EQUIVALENCE (DXMBD4(2086),SIGKJ(1,1)),(DXMBD4(2122),DAMPC)
C
      COMMON /BLD4X1/ DXMBD4
C
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /CONL1/ ENR,PIO2,XROOT,RAB,NMP1,NRP1,NAP1
```

```
      COMMON /BLD4X2/ HDOT,PHI,THET,FX,FZ,EMO,NMAS,NMODE,NAPSON,NR11
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /SAD1/ FV,FW,EMOME,CSI,CS2DT,CH,
     1CCP,SSP,CCPS,SCPS,SUMA,H,X,RADIS,PHIV,SI,THETA,CSPH,CAPHI,CC,SC,EX
     2,EXMT,SN,SMLAZ,SMLAX,SMLAM,ALAM,SFZ,SFX,SFM,FORC,JM,JMP1,HDT,APH,
     3ATH
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      COMMON/PRNT/NPRNT
C
C
C     COMPUTE LAMBDAS (ARGUEMENTS OF COSINE)
C     COMPUTE FOURIER COEFFICIENTS AT EASH AZIMUTH
C
      R=BD(1)
      IR=1
      DO 90 I=1,NM
      IM1=I-1
      X(I)=XROOT+.5*ELNTH(I)
      IF (IM1) 88,88,83
   83 DO 85 J=1,IM1
   85 X(I)=X(I)+ELNTH(J)
      X(NM+1)=RAB
C
C     COMPUTE LAMBDA
C
      IF (X(I).GT.RWK) GO TO 90
   88 IF (X(I).GT.RWK) GO TO 90
      SFZ(I)=0.
      SFX(I)=0.
      SFM(I)=0.
      IR=I
   90 CONTINUE
 3122 CONTINUE
C
C     SMALL A COEFFICIENTS
C
      RX=R
      XX(1)=RWK*RX
      DO 95 JA=1,NR1
      JAP1X=JA+1
   95 XX(JAP1X)=RBL(JA)*RX
      NR1P2=NR1+2
      DO 200 JA=1,NA
      DO 11 JX=1,NR1
      JG=JX+1
      FXX(JG)=FX(JX,JA)
      FZZ(JG)=FZ(JX,JA)
      EMOO(JG)=EMO(JX,JA)
   11 CONTINUE
      FXX(1)=0.0
```

```
      FZZ(1)=0.0
      EMOO(1)=0.0
      FXX(NR1P2)=0.
      FZZ(NR1P2)=0.
      EMOO(NR1P2)=0.
      XX(NR1P2)=RAB
      NZ=NM+1
      CALL ALINT (X,XX,FXX,SFX,NZ,NR1P2)
      CALL ALINT (X,XX,FZZ,SFZ,NZ,NR1P2)
      CALL ALINT (X,XX,EMOO,SFM,NZ,NR1P2)
C
C     LOADS READY FOR RESPONSE CALCULATION
C
      DO 150 I=1,NM
      FV(I,JA)=-SFZ(I)*CCPS(I)-SFX(I)*SCPS(I)
      FW(I,JA)=SFZ(I)*SCPS(I)-SFX(I)*CCPS(I)
      EMOME(I,JA)=SFM(I)+ZA(I)*FV(I,JA)
  150 CONTINUE
  200 CONTINUE
C
C     MAKE FIRST AND LAST ELEMENTS THE SAME
C
      DO 210 I=1,NM
      FV(I,NAP1)=FV(I,1)
      FW(I,NAP1)=FW(I,1)
      EMOME(I,NAP1)=EMOME(I,1)
      SIDT(I,NAP1)=SIDT(I,1)
  210 CONTINUE
      IF(IT3.EQ.1000) GO TO 100
      IF(NPRNT.LT.2) RETURN
  100 CONTINUE
      WRITE (6,211)
  211 FORMAT (3HOFV//)
      WRITE(6,9875)       ((   FV(I,JX),JX=1,NAP1),I=1,NM)
      WRITE (6,212)
  212 FORMAT (3HOFW//)
      WRITE(6,9875)       ((   FW(I,JX),JX=1,NAP1),I=1,NM)
      WRITE (6,213)
  213 FORMAT (6HOEMOME//)
      WRITE(6,9875)       ((EMOME(I,JX),JX=1,NAP1),I=1,NM)
      WRITE (6,214)
  214 FORMAT (5HOSIDT//)
      WRITE(6,9875)       (( SIDT(I,JX),JX=1,NAP1),I=1,NM)
C
C     END CONVERSION OF AERODYNAMIC LOADS
C
      RETURN
 9875 FORMAT (       (1H ,10G12.4))
      END
```

```
      SUBROUTINE ALINT (X,R,F,FR,NM,NL)
      DIMENSION X(NM),R(NL),F(NL),FR(NM)
      NMM1=NM-1
      DO 2 I=1,NM
    2 FR(I)=0.0
      I=1
      L=1
    4 XIP1=X(I+1)
      RL=R(L)
      IF(XIP1 .GT. RL) GO TO 5
      I=I+1
      GO TO 4
    7 IF(I .GE. NMM1) RETURN
      I=I+1
      XI=X(I)
      XIP1=X(I+1)
      IF(XIP1 .LT. RLP1) GO TO 10
      FR(I)=FR(I)+.5*(FL+FL+BS*(XI+RLP1-RL-RL))*(RLP1-XI)
    8 L=L+1
    5 LP1=L+1
      IF(LP1 .GT. NL) RETURN
      FLP1=F(LP1)
      FL=F(L)
      RLP1=R(LP1)
      RL=R(L)
      IF(XIP1 .LT. RLP1) GO TO 9
      FR(I)=FR(I)+.5*(FLP1+FL)*(RLP1-RL)
      GO TO 8
    9 BS=(FLP1-FL)/(RLP1-RL)
      FR(I)=FR(I)+.5*(FL+FL+BS*(XIP1-RL))*(XIP1-RL)
      GO TO 7
   10 FR(I)=FR(I)+.5*(FL+FL+BS*(XI+XIP1-RL-RL))*(XIP1-XI)
      GO TO 7
      END
```

```
      SUBROUTINE GCOORD
C
      DIMENSION HDOT(04,18),PHI(04,18),THET(04,18),FX(04,18),FZ(04,18),
     1EMO(04,18)
      DIMENSION EIZ(20)
      DIMENSION VDOT(20,19),WDOT(20,19),PHIDT(20,19),SIDT(20,19),CSIDT
     1(06,19)
      DIMENSION FV(20,19),FW(20,19),EMOME(20,19),CSI(06,19),CS2DT(06,19)
     1,CH(12,20),CCP(19),SSP(19),CCPS(20),SCPS(20),SUMAS(20),H(20),
     1 X(20),
     2RADIS(20),PHIV(20,19),SI(20,19),XCSIDT(1,18),THETA(20,19)
      DIMENSION CSPH(20),CAPHI(20),CC(19),SC(19),EX(19),EXMT(19),SN(05),
     1SMLAZ(20),SMLAX(20),SMLAM(20),ALAM(19),SFZ(19),SFX(19),SFM(19),FOR
     2C(19)
      DIMENSION JM(04),JMP1(04),HDT(04),APH(04),ATH(04)
C
      DIMENSION DXMBD4(2122)
      DIMENSION SIGKJ(6,6)
      DIMENSION ELNTH(20),EMAS(20),EIX(20),        RBL(04),
     1 XINR(20),DPHI(20),EIY(20),EPS(20),DLZ(20),ZA(20),
     2 YINR(20),OMEGA(20),SIG(20),BD(15)
C
      DIMENSION AV(20,06),AW(20,06),APHI(20,06),ASI(20,06),
     1 ATHET(20,06),AT(20,06),AMZ(20,06),AVY(20,06),
     2 AMY(20,06),AVZ(20,06)
C
      EQUIVALENCE (DXMBD4(1),ELNTH(1)),(DXMBD4(21),EMAS(1)),(DXMBD4(41),
     1EIX (1)),(DXMBD4(61),RBL (1)),(DXMBD4(70),XINR(1)),(DXMBD4(90),DPH
     2I(1)),(DXMBD4(110),EIY (1)),(DXMBD4(130),EPS(1)),(DXMBD4(150),DLZ(
     31)),(DXMBD4(170),ZA (1)),(DXMBD4(190),YINR(1)),(DXMBD4(210),OMEGA(
     41)),(DXMBD4(69),PSIR ),
     4    (DXMBD4(230),SIG (1)),(DXMBD4(250),AV (1)),(DXMBD4(430),AW (1))
     5,
     5(DXMBD4(610),APHI (1)),(DXMBD4(790),ASI (1)),(DXMBD4( 970),ATHET(1
     6)),(DXMBD4(1150),AT (1)),(DXMBD4(1330),AMZ (1)),(DXMBD4(1510),AVY
     7(1)),(DXMBD4(1690),AMY (1)),(DXMBD4(1870),AVZ (1)),(DXMBD4(2050),B
     8D (1)),(DXMBD4(2065),RWK ),(DXMBD4(2066),CSAL ),(DXMBD4(2067),NM),
     9(DXMBD4(2068),XCSIDT(1,1))
      EQUIVALENCE (DXMBD4(2086),SIGKJ(1,1)),(DXMBD4(2122),DAMPC)
C
      COMMON /BLD4X1/ DXMBD4
C
      COMMON /GCORD1/ THETO,CPSQ,DT,AC,BC,AKL,DTO2,R,ONOCP,CSALT
      COMMON /CONL1/ ENR,PIO2,XROOT,RAB,NMP1,NRP1,NAP1
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPCMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLD4X2/ HDOT,PHI,THET,FX,FZ,EMO,NMAS,NMODE,NAPSON,NR11
```

```
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /SAD1/ FV,FW,EMOME,CSI,CS2DT,CH,
     1 CCP,SSP,CCPS,SCPS,SUMAS,H,X,RADIS,PHIV,SI,THETA,CSPH,CAPHI,CC,SC,
     1 EX,
     2 EXMT,SN,SMLAZ,SMLAX,SMLAM,ALAM,SFZ,SFX,SFM,FORC,JM,JMP1,HDT,APH,
     3ATH
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      COMMON /GARY2/ XINPT,FINPT
      COMMON/PRNT/NPRNT
C
C     CERTAIN COEFFICIENTS FC COMPUTATION CF TOTAL GENERALIZED FORCES
C     AND MOMENTS
C
      SNTHC=SIN(THETO)
      DO 300 I=1,NM
      G1=SUMAS(I)*CPSQ*(H(I)+EPS(I)*CCPS(I))
      CH(1,I)=G1*SCPS(I)
      CH(2,I)=SUMAS(I)*(EPS(I)+H(I)*CSPH(I)/CCPS(I))
      CH(3,I)=-2.*SUMAS(I)*CPOMG*EPS(I)
      CH(4,I)=G1*CCPS(I)
      CH(5,I)=-SUMAS(I)*H(I)*SIN(CAPHI(I)-THETO)/CCPS(I)
      CH(6,I)=CH(3,I)*CCPS(I)
      CH(3,I)=CH(3,I)*SCPS(I)
      CH(7,I)=-CPSQ*SCPS(I)*XINR(I)*CCPS(I)-SCPS(I)*EPS(I)*G1
      CH(8,I)=-XINR(I)-EPS(I)*CH(2,I)
      CH(9,I)=2.*CPOMG*SCPS(I)*(XINR(I)*SUMAS(I)*EPS(I)*EPS(I))
      CH(10,I)=SUMAS(I)*CPSQ*EPS(I)*RADIS(I)
      CH(12,I)=-2.*CPOMG*(XINR(I)*SUMAS(I)*EPS(I)*EPS(I))
      CH(11,I)=CH(12,I)*SCPS(I)-2.*CPOMG*SUMAS(I)*EPS(I)*H(I)*SNTHO
     1          /CCPS(I)
  300 CONTINUE
      IF(NPRNT.LT.2) GO TO 9
      WRITE (6,301)
  301 FORMAT (3HOCH//)
      WRITE (6,9875) ((CH(IX,I),IX=1,11),I=1,NM)
    9  CONTINUE
C     COMPUTE SUPERPOSITION INTEGRALS IN EACH KTH MODE
      DO 550 K=1,MAXMO
      SIGOM=SIG(K)*OMEGA(K)
      OMK2=OMEGA(K)*OMEGA(K)
      OMBR=OMEGA(K)*SQRT(1.-SIG(K)*SIG(K))
      OMT=OMBR*DT
      IF(NPRNT.LT.2) GO TO 10
      WRITE(6,6001) K,SIG(K),OMEGA(K),DT
 6001 FORMAT(1HO,4HK = ,I3,5X,8HSIG =    ,G10.3,2X,8HOMEGA = ,G10.3,2X,
     1                          8HDT =     ,G10.3)
   10  CONTINUE
      SIKDT=SIN(OMT)
      CSKDT=COS(OMT)
      OMT2=CMT*OMT
```

```
      OMT3=CMT2*OMT
      IF(OMT-.05) 303,307,307
  303 CMT4=CMT2*OMT2
      OMT6=CMT4*OMT2
C     ALPHA,BETA,GAMMA COEFFICIENTS FOR FILONS RULE OF INTEGRAION .
C     SEE APPENDIX OF BOOK ON INTEGRAL TRANSFORMS BY TRANTER
      FILA=OMT3/22.5-(OMT4*OMT)/157.5+(OMT6*OMT)/2362.5
      FILB=.66666666666667+OMT2/7.5-OMT4/26.25+OMT6/283.5
      FILG=1.3333333333333-OMT2/7.5+OMT4/210.-OMT6/11340.
      GO TO 308
  307 FILA=(OMT2+OMT*SIKDT*CSKDT-2.*SIKDT*SIKDT)/OMT3
      FILB=2.*(OMT*(1.+CSKDT*CSKDT)-2.*SIKDT*CSKDT)/OMT3
      FILG=4.*(SIKDT-OMT*CSKDT)/OMT3
  308 ADT=FILA*DT
      BDT=FILB*DT
      GDT=FILG*DT
      DO 310 JA=1,NAP1
      T=(JA-1.)*DT
      FORC(JA)=0.
      OMT=OMBR*T
      CC(JA)=COS(OMT)
      SC(JA)=SIN(OMT)
  310 EXMT(JA)=EXP(-SIGOM*T)
      DO 360 JA=1,NA
      THC1=CPOMG*(AC*CCP(JA)-BC*SSP(JA))
      THC2=-CPSQ*(AC*SSP(JA)+BC*CCP(JA))
C     COMPUTE TOTAL GENERALIZED FORCES, PAGE IV-3,IV-4
      DO 350 I=1,NM
      QV=CH(1,I)+CH(2,I)*THC2+CH(3,I)*SIDT(I,JA)+FV(I,JA)
      QW=CH(4,I)+CH(5,I)*THC2+CH(6,I)*SIDT(I,JA)+FW(I,JA)
      QPHI=CH(7,I)+CH(8,I)*THC2+CH(9,I)*SIDT(I,JA)+EMOME(I,JA)
      QSI=CH(10,I)+CH(11,I)*THC1+CH(12,I)*PHIDT(I,JA)-CH(3,I)*
     1    VDOT(I,JA)-CH(6,I)*WDOT(I,JA)
  318 IF(I-2) 330,320,330
  320 G1=AKL/ELNTH(2)*SIDT(1,JA)
      QV=QV+SCPS(2)*G1
      QW=QW+CCPS(2)*G1
C     GENERALIZED FORCE ACTING IN EACH NORMALIZED MODE
  330 FORC(JA)=FORC(JA)+QV*AV(I,K)+QW*AW(I,K)+QPHI*APHI(I,K)+
     1          QSI*ASI(I,K)
  350 CONTINUE
      DO 355 KJ=1,MAXMO
  355 FORC(JA)=FORC(JA)-SIGKJ(K,KJ)*CSIDT(KJ,JA)
  360 CONTINUE
      FORC(NAP1)=FORC(1)
      WRITE(6,361)
  361 FORMAT (5HOFORC//)
      WRITE(6,9875)          (FORC(JA),JA=1,NA)
      DO 370 JA=1,NAP1
  370 EX(JA)=(1./EXMT(JA))*FORC(JA)
```

```
      JGO=1
C     COMPUTE SUPERPOSITION INTEGRALS AT EACH AZIMUTH POSITION
C     INTEGRALS ON PAGE IV-6
      DO 500 JAA=1,NAP1
      CSITMP=CSI(K,JAA)
      CSITXP=CSIDT(K,JAA)
  380 JAAM=JAA-1
      T=(JAA-1.)*DT
      GO TO (400,410,420,430),JGO
  400 CSI(K,JAA)=0.
      CSIDT(K,JAA)=0.
      JGO=2
      GO TO 500
  410 SAVE=DTO2*EX(2)
      CSINT=SAVE*CC(2)+DTO2*FORC(1)
      SNINT=SAVE*SC(2)
      JGO=3
      GO TO 450
  420 JGO=4
 4201 CSINT=EX(JAA)*(ADT*SC(JAA)+.5*BDT*CC(JAA))+FORC(1)*BDT*.5
      SNINT=EX(JAA)*(-ADT*CC(JAA)+.5*BDT*SC(JAA))+FORC(1)*ADT
  421 JAGO=1
      DO 425 JA=2,JAAM
      GO TO (422,423),JAGO
  422 SAVE=GDT*EX(JA)
      CSINT=CSINT+CC(JA)*SAVE
      SNINT=SNINT+SC(JA)*SAVE
      JAGO=2
      GO TO 425
  423 SAVE=BDT*EX(JA)
      CSINT=CSINT+CC(JA)*SAVE
      SNINT=SNINT+SC(JA)*SAVE
      JAGO=1
  425 CONTINUE
      SAVEC=CSINT
      SAVES=SNINT
      GO TO 450
  430 CSINT=DTO2*(CC(JAA)*EX(JAA)+CC(JAA-1)*EX(JAA-1))+SAVEC
      SNINT=DTO2*(SC(JAA)*EX(JAA)+SC(JAA-1)*EX(JAA-1))+SAVES
      JGO=3
  450 CSI(K,JAA)=EXMT(JAA)*(SC(JAA)*CSINT-CC(JAA)*SNINT)
      CSIDT(K,JAA)=EXMT(JAA)*(CC(JAA)*CSINT+SC(JAA)*SNINT)
      CSI(K,JAA)=CSI(K,JAA)/OMBR
      IF(JAA-NAP1) 457,455,455
  455 SKINT=EXMT(JAA)*(SC(JAA)*CSINT-CC(JAA)*SNINT)
      CKINT=EXMT(JAA)*(CC(JAA)*CSINT+SC(JAA)*SNINT)
      CK=1.-EXMT(JAA)*CC(JAA)
      SK=EXMT(JAA)*SC(JAA)
      S=(SK*SK+CK*CK)
      SI=(SK*CKINT+CK*SKINT)/S
```

```
      S2=(CK*CKINT-SK*SKINT)/S
  457 CONTINUE
  500 CONTINUE
C     COMPUTE QUANTITIES ZETA (CSI) AND ZETA DOT (CSI DOT) FROM
C     SUPERPOSITION INTEGRALS
C     COMPUTATION OF ZETA,ZETA-DOT, SEE PAGE IV-6
      DO 520 JAA=1,NAP1
      CSI(K,JAA)=CSI(K,JAA)+EXMT(JAA)*(CC(JAA)*S1+SC(JAA)*S2)/OMBR
      CSIDT(K,JAA)=-SIGOM*CSI(K,JAA)+EXMT(JAA)*(CC(JAA)*S2-SC(JAA)*S1)
     1             +CSIDT(K,JAA)
  520 CS2DT(K,JAA)=FORC(JAA)-OMK2*CSI(K,JAA)-2.*SIGOM*CSIDT(K,JAA)
C     IF (K.NE.1) GO TO 550
C     DO 540 JAA=1,NAP1
C     CSI(K,JAA)=(CSI(K,JAA)*FINPT+CSITMP*(1.-FINPT))
C 540 CSIDT(K,JAA)=(CSIDT(K,JAA)*FINPT+CSITXP*(1.-FINPT))
  550 CONTINUE
      IF(NPRNT.LT.2) RETURN
      WRITE (6,551)
  551 FORMAT (4HOCSI//)
      WRITE(6,9875)       ((  CSI(K,JAA),K=1,MAXMO),JAA=1,NAP1)
      WRITE (6,552)
  552 FORMAT (6HOCSIDT//)
      WRITE(6,9875)       ((CSIDT(K,JAA),K=1,MAXMO),JAA=1,NAP1)
      WRITE (6,553)
  553 FORMAT (6HOCS2DT//)
      WRITE(6,9875)       ((CS2DT(K,JAA),K=1,MAXMO),JAA=1,NAP1)
      RETURN
 9875 FORMAT(       (1H ,06G12.4))
      END
```

```
      SUBROUTINE RSPNS
C
      DIMENSION VX(20,19),WX(20,19)
      DIMENSION HDOT(04,18),PHI(04,18),THET(04,18),FX(04,18),FZ(04,18),
     1EMO(04,18)
      DIMENSION EIZ(20)
      DIMENSION VDOT(20,19),WDOT(20,19),PHIDT(20,19),SIDT(20,19),CSIDT
     1(06,19)
      DIMENSION FV(20,19),FW(20,19),EMOME(20,19),CSI(06,19),CS2DT(06,19
     1,CH(12,20),CCP(19),SSP(19),CCPS(20),SCPS(20),SUMA(20),H(20),X(20)
     2RADIS(20),PHIV(20,19),SI(20,19),XCSIDT(1,18),THETA(20,19)
      DIMENSION CSPH(20),CAPHI(20),CC(19),SC(19),EX(19),EXMT(19),SN(05)
     1SMLAZ(20),SMLAX(20),SMLAM(20),ALAM(19),SFZ(19),SFX(19),SFM(19),FC
     2C(19)
      DIMENSION JM(04),JMP1(04),HDT(04),APH(04),ATH(04)
C
      DIMENSION DXMBD4(2122)
      DIMENSION SIGKJ(6,6)
      DIMENSION ELNTH(20),EMAS(20),EIX(20),        RBL(04),
     1 XINR(20),DPHI(20),EIY(20),EPS(20),DLZ(20),ZA(20),
     2 YINR(20),OMEGA(20),SIG(20),BD(15)
C
      DIMENSION AV(20,06),AW(20,06),APHI(20,06),ASI(20,06),
     1 ATHET(20,06),AT(20,06),AMZ(20,06),AVY(20,06),
     2 AMY(20,06),AVZ(20,06)
C
      EQUIVALENCE (DXMBD4(1),ELNTH(1)),(DXMBD4(21),EMAS(1)),(DXMBD4(41)
     1EIX (1)),(DXMBD4(61),RBL (1)),(DXMBD4(70),XINR(1)),(DXMBD4(90),DP
     2I(1)),(DXMBD4(110),EIY (1)),(DXMBD4(130),EPS(1)),(DXMBD4(150),DLZ
     31)),(DXMBD4(170),ZA (1)),(DXMBD4(190),YINR(1)),(DXMBD4(210),OMEGA
     41)),(DXMBD4(69),PSIR ),
     4   (DXMBD4(230),SIG (1)),(DXMBD4(250),AV (1)),(DXMBD4(430),AW (1)
     5,
     5(DXMBD4(610),APHI (1)),(DXMBD4(790),ASI (1)),(DXMBD4( 970),ATHET(
     6)),(DXMBD4(1150),AT (1)),(DXMBD4(1330),AMZ (1)),(DXMBD4(1510),AVY
     7(1)),(DXMBD4(1690),AMY (1)),(DXMBD4(1870),AVZ (1)),(DXMBD4(2050),
     8D (1)),(DXMBD4(2065),RWK ),(DXMBD4(2066),CSAL ),(DXMBD4(2067),NM
     8,
     9(DXMBD4(2068),XCSIDT(1,1))
      EQUIVALENCE (DXMBD4(2086),SIGKJ(1,1)),(DXMBD4(2122),DAMPC)
C
      COMMON /BLD4X1/ DXMBD4
C
      COMMON /GCORD1/ THETO,CPSQ,DT,AC,BC,AKL,DTO2,R,ONOCP,CSALT
      COMMON /CONL1/ ENR,PIO2,XROOT,RAB,NMP1,NRP1,NAP1
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMC,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
```

```
      COMMON /BLD4X2/ HDOT,PHI,THET,FX,FZ,EMO,NMAS,NMODE,NAPSON,NR11
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /SAD1/ FV,FW,EMOME,CSI,CS2DT,CH,
     1CCP,SSP,CCPS,SCPS,SUMA,H,X,RADIS,PHIV,SI,THETA,CSPH,CAPHI,CC,SC,EX
     2,EXMT,SN,SMLAZ,SMLAX,SMLAM,ALAM,SFZ,SFX,SFM,FORC,JM,JMP1,HDT,APH,
     3ATH
      COMMON /SVSNR/ SNR
      COMMON /GARY/ NGOTO
      COMMON /SAD6/ VX,WX
C
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      COMMON/PRNT/NPRNT
C
C     COMPUTE RESPONSES FROM MODE SHAPES AND ZETAS
      DO 575 JA=1,NAP1
      DO 575 I=1,NM
      VX(I,JA)=0.
      WX(I,JA)=0.
      VDOT(I,JA)=0.
      WDOT(I,JA)=0.
      PHIV(I,JA)=0.
      PHIDT(I,JA)=0.
      SI(I,JA)=0.
      SIDT(I,JA)=0.
  575 THETA(I,JA)=0.
C     RESPONSE VARIABLES OF INTEREST, PAGE IV-7
      DO 580 JA=1,NAP1
      DO 580 I=1,NM
      DO 580 K=1,MAXMO
      VX(I,JA)=VX(I,JA)+AV(I,K)*CSI(K,JA)
      WX(I,JA)=WX(I,JA)+AW(I,K)*CSI(K,JA)
      VDOT(I,JA)=VDOT(I,JA)+AV(I,K)*CSIDT(K,JA)
      WDOT(I,JA)=WDOT(I,JA)+AW(I,K)*CSIDT(K,JA)
      PHIV(I,JA)=PHIV(I,JA)+APHI(I,K)*CSI(K,JA)
      PHIDT(I,JA)=PHIDT(I,JA)+APHI(I,K)*CSIDT(K,JA)
      SI(I,JA)=SI(I,JA)+ASI(I,K)*CSI(K,JA)
      SIDT(I,JA)=SIDT(I,JA)+ASI(I,K)*CSIDT(K,JA)
      THETA(I,JA)=THETA(I,JA)+ATHET(I,K)*CSI(K,JA)
  580 CONTINUE
      IF(IT3.EQ.1000) GO TO 9
      IF(NPRNT.LT.2) GO TO 3
    9 CONTINUE
      NAP1=NA
      WRITE (6,9988) ((VX(I,JA),JA=1,NAP1),I=1,NM)
      WRITE (6,9989) ((WX(I,JA),JA=1,NAP1),I=1,NM)
 9988 FORMAT (/1H0,3HVX //(8E14.6))
 9989 FORMAT (/1H0,3HWX //(8E14.6))
      WRITE(6,9990)(( VDOT(I,JA),JA=1,NAP1),I=1,NM)
      WRITE(6,9991)(( WDOT(I,JA),JA=1,NAP1),I=1,NM)
      WRITE(6,9992)(( PHIV(I,JA),JA=1,NAP1),I=1,NM)
```

```
      WRITE(6,9993)((PHIDT(I,JA),JA=1,NAP1),I=1,NM)
      WRITE(6,9994)((   SI(I,JA),JA=1,NAP1),I=1,NM)
      WRITE(6,9995)(( SIDT(I,JA),JA=1,NAP1),I=1,NM)
      WRITE(6,9996)((THETA(I,JA),JA=1,NAP1),I=1,NM)
      GO TO (3,4),NGOTO
 9990 FORMAT(/1HO,5HVDOT //(7G14.6))
 9991 FORMAT(/1HO,5HWDOT //(7G14.6))
 9992 FORMAT(/1HO,5HPHIV //(7G14.6))
 9993 FORMAT(/1HO,5HPHIDT//(7G14.6))
 9994 FORMAT(/1HO,5HSI   //(7G14.6))
 9995 FORMAT(/1HO,5HSIDT //(7G14.6))
 9996 FORMAT(/1HO,5HTHETA//(7G14.6))
C
C
C
C     CONVERT RESPONSE TO FORM FOR LOADS COMPUTATIONS
    3 DO 610 I=1,NM
  610 X(I)=RADIS(I)
      MJ=1
      DO 625 I=1,NR
      RRBL=R*RBL(I)
      M=MJ
      DO 620 J=M,NM
      MJ=J
      IF (X(J).LT.RRBL) GO TO 620
      K=J
      IF (K.EQ.1) K=K+1
      JM(I)=K-1
      JMP1(I)=K
      GO TO 625
  620 CONTINUE
      JM(I)=NM-1
      JMP1(I)=NM
  625 CONTINUE
      DO 700 JA=1,NA
      THETC=AC*SSP(JA)+BC*CCP(JA)
      THC1=CPOMG*(AC*CCP(JA)-BC*SSP(JA))
      DO 650 M=1,NR
      RRBL=R*RBL(M)
      IF(JM(M)) 630,650,630
  630 I=JM(M)
      IP1=JMP1(M)
      FCTR=(RRBL-X(I))/(X(IP1)-X(I))
      DO 640 II=1,2
       THT=THETO+THETC+PHIV(I,JA)
      PH=-THETA(I,JA)*CCPS(I)-SI(I,JA)*SCPS(I)
      HD=CNOCP*(VDOT(I,JA)*CCPS(I)-WDOT(I,JA)*SCPS(I)+ZA(I)*PHIDT(I,JA)
     1CCPS(I)+(ZA(I)-H(I))*(THC1+CPOMG*PH))-AMU*CSALT*CCP(JA)*PH
      GO TO (637,645),II
  637 HDT(M)=HD*(1.-FCTR)
```

```
      ATH(M)=THT*(1.-FCTR)
      APH(M)=PH*(1.-FCTR)
      I=IP1
  640 CONTINUE
  645 HDT(M)=HDT(M)+FCTR*HD
      APH(M)=APH(M)+FCTR*PH
      ATH(M)=ATH(M)+FCTR*THT
  650 CONTINUE
      DO 660 M=1,NR
      HDOT(M,JA)=HDT(M)
      PHI(M,JA)=APH(M)
  660 THET (M,JA)=ATH(M)
  700 CONTINUE
      IF(IT3.EQ.1000) GO TO 10
      IF(NPRNT.LT.2) GO TO 4
   10 CONTINUE
      WRITE(6,24)
   24 FORMAT (5HOHDOT//)
      WRITE(6,29)((HDOT (I,J),J=1,NA),I=1,NR)
29    FORMAT(10(1X,E12.5),/)
      WRITE(6,22)
   22 FORMAT (4HOPHI//)
      WRITE(6,29)((PHI  (I,J),J=1,NA),I=1,NR)
      WRITE(6,23)
   23 FORMAT (6HOTHETA//)
      WRITE(6,29)((THETA(I,J),J=1,NA),I=1,NR)
      WRITE (6,28)
   28 FORMAT (5HOTHET//)
      WRITE (6,29) ((THET(I,J),J=1,NA),I=1,NR)
 2935 FORMAT (2(/),55X,10HCSI VALUES)
    4  WRITE(NOUT,2935)
      WRITE(NOUT,938)  (( II,JJ,CSI(II,JJ) ,II=1,MAXMO),JJ=1,NA)
938   FORMAT(6(2I3,F10.5))
      IF(IT3-1000)890,750,750
 750  CONTINUE
  900 FORMAT(1X,5E20.7)
C
      NAOVTO=20
      NRNRA=19
      WRITE(6,8993)
      CALL HARMN(NM,NAP1,DSI,VDOT ,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8994)
      CALL HARMN(NM,NAP1,DSI,WDOT ,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8995)
      CALL HARMN(NM,NAP1,DSI,PHIV ,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8996)
      CALL HARMN(NM,NAP1,DSI,PHIDT,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8997)
      CALL HARMN(NM,NAP1,DSI,SI   ,6,NMAS,NACVTO,NRNRA)
      WRITE(6,8998)
```

```
      CALL HARMN(NM,NAP1,DSI,SIDT ,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8999)
      CALL HARMN(NM,NAP1,DSI,THETA,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8991)
      CALL HARMN(NM,NAP1,DSI,VX   ,6,NMAS,NAOVTO,NRNRA)
      WRITE(6,8992)
      CALL HARMN(NM,NAP1,DSI,WX   ,6,NMAS,NAOVTO,NRNRA)
8991  FORMAT(/57X,21HFLATWISE DISPLACEMENT)
8992  FORMAT(/57X,22HCHORDWISE DISPLACEMENT)
8993  FORMAT(/57X,17HFLATWISE VELOCITY)
8994  FORMAT(/57X,18HCHORDWISE VELOCITY)
8995  FORMAT(/57X,26HTORSIONAL DEFLECTION ANGLE)
8996  FORMAT(/57X,33HTORSIONAL DEFLECTION ANGULAR RATE)
8997  FORMAT(/57X,23HCHORDWISE BENDING-SLOPE)
8998  FORMAT(/57X,36HCHORDWISE BENDING-SLOPE ANGULAR RATE)
8999  FORMAT(/57X,35HFLATWISE BENDING-SLOPE ANGULAR RATE)
      NAP1=NA+1
890   NR=SNR
      RETURN
      END
```

```
      SUBROUTINE SHEAR
      DIMENSION HDOT(4,18),PHI(4,18),THET(4,18),FX(4,18),FZ(4,18),
     1EMO(4,18)
      DIMENSION EIZ(20)
      DIMENSION VDOT(20,19),WDOT(20,19),PHIDT(20,19),SIDT(20,19),CSIDT
     1(6,19)
      DIMENSION FV(20,19),FW(20,19),EMOME(20,19),CSI(6,19),CS2DT(6,19)
     1,CH(12,20),CCP(19),SSP(19),CCPS(20),SCPS(20),SUMA(20),H(20),X(20),
     2RADIS(20),PHIV(20,19),SI(20,19),XCSIDT(1,18),THETA(20,19)
      DIMENSION CSPH(20),CAPHI(20),CC(19),SC(19),EX(19),EXMT(19),SN(05),
     1SMLAZ(20),SMLAX(20),SMLAM(20),ALAM(19),SFZ(19),SFX(19),SFM(19),FOR
     2C(19)
      DIMENSION JM(4),JMP1(4),HDT(4),APH(4),ATH(4)
      DIMENSION DXMBD4(2122)
     A,ELNTH(20),EMAS(20),EIX(20),RBL(4),
     1XINR(20),DPHI(20),EIY(20),EPS(20),DLZ(20),ZA(20),
     2 YINR(20),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(20,6),AW(20,6),APHI(20,6),ASI(20,6),
     1 ATHET(20,6),AT(20,6),AMZ(20,6),AVY(20,6),
     2 AMY(20,6),AVZ(20,6)
      EQUIVALENCE (DXMBD4(1),ELNTH(1)),(DXMBD4(21),EMAS(1)),(DXMBD4(41),
     1EIX(1)),(DXMBD4(61),RBL(1)),(DXMBD4(70),XINR(1)),(DXMBD4(90),DPH
     2I(1)),(DXMBD4(110),EIY(1)),(DXMBD4(130),EPS(1)),(DXMBD4(150),DLZ(
     31)),(DXMBD4(170),ZA(1)),(DXMBD4(190),YINR),(DXMBD4(210),OMEGA(
     41)),(DXMBD4(69),PSIR),
     4(DXMBD4(230),SIG(1)),(DXMBD4(250),AV),(DXMBD4(430),AW),
     5(DXMBD4(610),APHI),(DXMBD4(790),ASI(1)),(DXMBD4(910),ATHET
     6),(DXMBD4(1150),AT),(DXMBD4(1330),AMZ),(DXMBD4(1510),AVY
     7),(DXMBD4(1690),AMY),(DXMBD4(1870),AVZ),(DXMBD4(2050),B
     8D),(DXMBD4(2065),RWK),(DXMBD4(2066),CSAL),(DXMBD4(2067),NM),
     9(DXMBD4(2068),XCSIDT)
C
      COMMON /BLD4X1/ DXMBD4
C
      COMMON /GCORD1/ THETO,CPSQ,DT,AC,BC,AKL,DTO2,R,ONOCP,CSALT
      COMMON /CONL1/ ENR,PIO2,XROOT,RAB,NMP1,NRP1,NAP1
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLD4X2/ HDOT,PHI,THET,FX,FZ,EMO,NMAS,NMODE,NAPSON,NR11
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /SAD1/ FV,FW,EMOME,CSI,CS2DT,CH,
     1CCP,SSP,CCPS,SCPS,SUMA,H,X,RADIS,PHIV,SI,THETA,CSPH,CAPHI,CC,SC,EX
     2,EXMT,SN,SMLAZ,SMLAX,SMLAM,ALAM,SFZ,SFX,SFM,FORC,JM,JMP1,HDT,APH,
     3ATH
      COMMON /SVSNR/ SNR
      COMMON /GARY/ NGOTO
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      DIMENSION CT(20,19),CMZ(20,19),CMY(20,19),CVZ(20,19),CVY(20,19)
      EQUIVALENCE (CT(1,1),FV(1,1)),(CMZ(1,1),FW(1,1)),(CMY(1,1),EMOME(1
     1,1)),(CVZ(1,1),VDOT(1,1)),(CVY(1,1),WDOT(1,1))
```

```
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPONG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /SHEAR1/ NAOVTO
      NRNRNA=NR*NA
      DO 200 JA=1,NAP1
      DO 200 I=1,NM
      DT=0.
      DMY=0.
      DMZ=0.
      DVY=0.
      DVZ=0.
      DO 100 K=1,MAXMO
      CSIX=CSI(K,JA)
      DT=DT+AT(I,K)*CSIX
      DMZ=DMZ+AMZ(I,K)*CSIX
      DMY=DMY+AMY(I,K)*CSIX
      DVZ=DVZ+AVZ(I,K)*CSIX
      DVY=DVY+AVY(I,K)*CSIX
  100 CONTINUE
      CT(I,JA)=DT
      CMZ(I,JA)=DMZ
      CMY(I,JA)=DMY
      CVZ(I,JA)=DVZ
      CVY(I,JA)=DVY
  200 CONTINUE
      NSAVE=NA
      NAOVTO=20
      NRNRNA=19
      CALL HARMN (NM,NA,DSI,CT,1,NMAS,NAOVTO,NRNRNA)
      WRITE (6,900)
  900 FORMAT (///,56X,19HRADIAL VS AZIMUTHAL)
      WRITE (6,901) ((CT(I,J),J=1,NA),I=1,NM)
  901 FORMAT (///(8E14.6))
      CALL HARMN (NM,NA,DSI,CMZ,2,NMAS,NAOVTO,NRNRNA)
      WRITE (6,900)
      WRITE (6,901) ((CMZ(I,J),J=1,NA),I=1,NM)
      CALL HARMN (NM,NA,DSI,CMY,4,NMAS,NAOVTO,NRNRNA)
      WRITE (6,900)
      WRITE (6,901) ((CMY(I,J),J=1,NA),I=1,NM)
      CALL HARMN (NM,NA,DSI,CVZ,5,NMAS,NAOVTO,NRNRNA)
      WRITE (6,900)
      WRITE (6,901) ((CVZ(I,J),J=1,NA),I=1,NM)
      CALL HARMN (NM,NA,DSI,CVY,3,NMAS,NAOVTO,NRNRNA)
      WRITE (6,900)
      WRITE (6,901) ((CVY(I,J),J=1,NA),I=1,NM)
      NA=NSAVE
      RETURN
      END
```

```
      SUBROUTINE RSPZZ
C
      DIMENSION RCAP(5)
      DIMENSION ZWK(5,18)
      DIMENSION VX(20,19),WX(20,19)
      DIMENSION HDOT(04,18),PHI(04,18),THET(04,18),FX(04,18),FZ(04,18),
     1EMO(04,18)
      DIMENSION EIZ(20)
      DIMENSION VDOT(20,19),WDOT(20,19),PHIDT(20,19),SIDT(20,19),CSIDT
     1(06,19)
      DIMENSION FV(20,19),FW(20,19),EMOME(20,19),CSI(06,19),CS2DT(06,19)
     1,CH(12,20),CCP(19),SSP(19),CCPS(20),SCPS(20),SUMA(20),H(20),X(20),
     2RADIS(20),PHIV(20,19),SI(20,19),XCSIDT(1,18),THETA(20,19)
      DIMENSION CSPH(20),CAPHI(20),CC(19),SC(19),EX(19),EXMT(19),SN(05),
     1SMLAZ(20),SMLAX(20),SMLAM(20),ALAM(19),SFZ(19),SFX(19),SFM(19),FOR
     2C(19)
      DIMENSION KM(05),KMP1(05)
      DIMENSION JM(04),JMP1(04),HDT(04),APH(04),ATH(04)
C
      DIMENSION DXMBD4(2122)
      DIMENSION ELNTH(20),EMAS(20),EIX(20),        RBL(04),
     1 XINR(20),DPHI(20),EIY(20),EPS(20),DLZ(20),ZA(20),
     2 YINR(20),OMEGA(20),SIG(20),BD(15)
C
      DIMENSION AV(20,06),AW(20,06),APHI(20,06),ASI(20,06),
     1 ATHET(20,06),AT(20,06),AMZ(20,06),AVY(20,06),
     2 AMY(20,06),AVZ(20,06)
C
      EQUIVALENCE (DXMBD4(1),ELNTH(1)),(DXMBD4(21),EMAS(1)),(DXMBD4(41),
     1EIX (1)),(DXMBD4(61),RBL (1)),(DXMBD4(70),XINR(1)),(DXMBD4(90),DPH
     2I(1)),(DXMBD4(110),EIY (1)),(DXMBD4(130),EPS(1)),(DXMBD4(150),DLZ(
     31)),(DXMBD4(170),ZA (1)),(DXMBD4(190),YINR(1)),(DXMBD4(210),OMEGA(
     41)),(DXMBD4(69),PSIR ),
     4    (DXMBD4(230),SIG (1)),(DXMBD4(250),AV (1)),(DXMBD4(430),AW (1))
     5,
     5(DXMBD4(610),APHI (1)),(DXMBD4(790),ASI (1)),(DXMBD4( 970),ATHET(1
     6)),(DXMBD4(1150),AT (1)),(DXMBD4(1330),AMZ (1)),(DXMBD4(1510),AVY
     7(1)),(DXMBD4(1690),AMY (1)),(DXMBD4(1870),AVZ (1)),(DXMBD4(2050),B
     8D (1)),(DXMBD4(2065),RWK ),(DXMBD4(2066),CSAL ),(DXMBD4(2067),NM )
     8,
     9(DXMBD4(2068),XCSIDT(1,1))
C
      COMMON /BLD4X1/ DXMBD4
C
      COMMON /GCORD1/ THETO,CPSQ,DT,AC,BC,AKL,DTO2,R,ONOCP,CSALT
      COMMON /CONL1/ ENR,PIO2,XROOT,RAB,NMP1,NRP1,NAP1
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
```

```
     1 RA1,RB1,ERRSV
      COMMON /BLD4X2/ HDOT,PHI,THET,FX,FZ,EMO,NMAS,NMODE,NAPSON,NR11
      COMMON /BLD4X3/ AMU,NA,NR1
      COMMON /SAD1/ FV,FW,EMOME,CSI,CS2DT,CH,
     1CCP,SSP,CCPS,SCPS,SUMA,H,X,RADIS,PHIV,SI,THETA,CSPH,CAPHI,CC,SC,EX
     2,EXMT,SN,SMLAZ,SMLAX,SMLAM,ALAM,SFZ,SFX,SFM,FORC,JM,JMP1,HDT,APH,
     3ATH
      COMMON /SVSNR/ SNR
      COMMON /GARY/ NGOTO
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      COMMON /SAD6/ VX,WX
C
C
C     COMPUTE Z S FOR CASE 12 WAKE AND LOADS RUNS
    4 READ(5,9990)(RCAP(I),I=1,NRP1)
 9990 FORMAT (8F10.9)
C
C
C
      DO 610 I=1,NM
      X(I)=XROOT
      DO 610 J=1,I
  610 X(I)=X(I)+ELNTH(J)
      MJ=1
      DO 625 I=1,NRP1
      RRBL=RCAP(I)
      M=MJ
      DO 620 J=M,NM
      MJ=J
      IF (X(J).LT.RRBL) GO TO 620
      K=J
      IF (K.EQ.1) K=K+1
      KM(I)=K-1
      KMP1(I)=K
      GO TO 625
  620 CONTINUE
      KM(I)=NM-1
      KMP1(I)=NM
  625 CONTINUE
      DO 700 JA=1,NA
      DO 650 M=1,NRP1
      RRBL=RCAP(M)
      IF(KM(M)) 630,650,630
  630 I=KM(M)
      IP1=KMP1(M)
      FCTR=(RRBL-X(I))/(X(IP1)-X(I))
      DO 640 II=1,2
      HD=-VX(I,JA)
      GO TO (637,645),II
  637 HDT(M)=HD*(1.-FCTR)
```

```
      I=IP1
  640 CONTINUE
  645 HDT(M)=HDT(M)+FCTR*HD
  650 CONTINUE
      DO 660 M=1,NRP1
  660 ZWK(M,JA)=HDT(M)/R
  700 CONTINUE
      WRITE(6,24)
   24 FORMAT(    5HOZWK ,//)
      WRITE(3,29)(( ZWK (I,J),I=1,NRP1),J=1,NA)
      WRITE(6,30)(( ZWK (I,J),I=1,NRP1),J=1,NA)
   29 FORMAT( 29X,F10.7)
   30 FORMAT (1X,8E16.7)
    3 CONTINUE
      RETURN
      END
```

Machine Compatibility

The Blade Loads Program has been run on the University of Rochester's IBM 360/65 under MVT Release 18, General Computer Corporation's CDC 6600 under Scope 3.2, and NASA-Langley's CDC 6600 under Scope 3.2. The program is standard FORTRAN IV and is also WATFIV compatible.

Recommended CDC 6600 Overlay Statements:

Mainline:

```
OVERLAY(BLADES,0,0)
PROGRAM BLD34(INPUT, OUTPUT, BDSTRT, BDGAM, BDSIG, PUNCH,
   TAPE2=BDSTRT, TAPE5=INPUT, TAPE6=OUTPUT,
   TAPE7=BDGAM, TAPE8=BDSIG, TAPE3=PUNCH)
2004 CALL OVERLAY(6LBLADES,1,0,6HRECALL) replaces
   2004 CALL BLD3
CALL OVERLAY(6LBLADES,2,0,6HRECALL) replaces
   CALL BLD4
```

Subroutine BLD3:

```
OVERALY(BLADES,1,0)
PROGRAM BLD3
```

Subroutine BLD4:

```
OVERLAY(BLADES,2,0)
PROGRAM BLD4
CALL OVERLAY(6LBLADES,2,1,6HRECALL) replaces CALL CONVL
CALL OVERLAY(6LBLADES,2,2,6HRECALL) replaces CALL GCOORD
CALL OVERLAY(6LBLADES,2,3,6HRECALL) replaces CALL RSPNS
CALL OVERLAY(6LBLADES,2,4,6HRECALL) replaces CALL SHEAR
IF(NPCH.EQ.1) CALL OVERLAY(6HBLADES,2,5,6HRECALL)
   replaces IF(NPCH.EQ.1)CALL RSPZZ
```

Subroutine CONVL:

```
OVERLAY(BLADES,2,1)
PROGRAM CONVL
```

Subroutine GCOORD:

```
OVERLAY(BLADES,2,2)
PROGRAM GCOORD
```

Subroutine RSPNS:

```
OVERLAY(BLADES,2,3)
PROGRAM RSPNS
```

Subroutine SHEAR:

```
OVERLAY(BLADES,2,4)
PROGRAM SHEAR
```

Subroutine RSPZZ

```
OVERLAY(BLADES,2,5)
PROGRAM RSPZZ
```