NASA CONTRACTOR REPORT

NASA CR-2111

NASA CR-2111

MAIN ROTOR FREE WAKE GEOMETRY EFFECTS ON BLADE AIR LOADS AND RESPONSE FOR HELICOPTERS IN STEADY MANEUVERS

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. • SEPTEMBER 1972

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-2111		

4. Title and Subtitle	5. Report Date
MAIN ROTOR FREE WAKE GEOMETRY EFFECTS ON BLADE AIR LOADS AND RESPONSE FOR HELICOPTERS IN STEADY MANEUVERS. VOLUME II – PROGRAM LISTINGS	September 1972
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
S. Gene Sadler	RASA Rep. No. 71-13
	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, N.Y. 14618	NAS 1-8448
	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

16. Abstract

A mathematical model and computer program was implemented to study the main rotor free wake geometry effects on helicopter rotor blade air loads and response in steady maneuvers. Volume I (NASA CR-2110) contains the theoretical formulation and analysis of results. Volume II contains the computer program listing.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Rotor wake Blade loads Maneuvers Math models	Unclassified – Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	163	$3.00

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

CONTENTS

CONTENTS

MAIN ROTOR FREE WAKE GEOMETRY EFFECTS ON BLADE AIR LOADS AND RESPONSE FOR HELICOPTERS IN STEADY MANEUVERS

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 programs compute blade loads and response for single rotor helicopters in steady maneuver flight conditions. The listings in this volume correspond to the calculations discussed in VOLUME I.

INTRODUCTION

Vortex-blade interactions are an important source of high frequency, high amplitude aerodynamic loading of helicopter rotors. Increasingly more complete models of both the aerodynamics and elastomechanics of the helicopter rotor system are being developed. The programs listed here include the effects of free wake distortions, blade flexibilities, nonlinear aerodynamics, and uses an iterative solution technique to obtain compatible blade loads and response.

Four steps are necessary in obtaining blade loads and response results including the effects of free wake distortions by using the programs listed in this report:

1. Preliminary calculations (or measured data) are used to define rotor system performance parameters and flight conditions. Definitions of model parameters and program control variables are necessary for program operation.

2. A wake geometry calculation is made to obtain wake-induced velocity influence coefficients and initial estimates of bound circulations for use in the blade loads calculation. Wake geometry data is also printed during this calculation. (If uniform inflow approximations are desired, this step may be omitted, and the blade

*VOLUME I - THEORETICAL FORMULATION AND ANALYSIS OF RESULTS is contained in NASA CR-2110.

loads and response calculations performed without the effects of a freely distorting wake.)

3. Blade natural frequencies and normal modes are computed for use in calculating blade response. The normal modes may be coupled or uncoupled, but must be orthogonal and must have a generalized mass of unity. At least one normal mode is required for program operation. (Steps 2 and 3 are independent, and their order unimportant.)

4. Given the wake program input for use in blade loads calculations and the natural frequency and normal mode shape input for use in blade response calculations, the blade loads and response calculations are then performed by the blade loads and response program. Output of this program includes the wake-induced velocities, angles of attack, aerodynamic loads, and blade lineal and angular motions, moments and shears as computed from the appropriate normal mode quantities and generalized coordinate magnitudes.

Program input and output is in English units.

```
      OVERLAY(WKOVL,0,0)
      PROGRAM GEOW (INPUT,OUTPUT,BDSIG,BDGAM,WKGEO,TAPE5=INPUT,TAPE6=OUT
     1PUT,TAPE4=BDSIG,TAPE8=BDGAM,TAPE10=WKGEO)
C     PROGRAM WAKE GEOMETRY
C
      DIMENSION TM(9),TV(3)
C
      COMMON /MUVXYZ/ TM,TV,DEL,VDT,RC,CAPPHI,AQ,AZ,YR,ZR,RP,IVAR
      COMMON /STPSZ/ NRATIO,NAA,LRGWKS,LIMLSS,LSWW
C
      CALL GEOM
      CALL WK1
C
      STOP
      END
```

```
      SUBROUTINE GEOM
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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION AO(01)
      DIMENSION ALPHAO(01)
      DIMENSION ALPHAR(01)
      DIMENSION AR(01)
      DIMENSION ATMP(11)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION BTMP(11)
      DIMENSION C(09)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DI(90)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION DSQ(040)
      DIMENSION DTMP(040)
      DIMENSION GAMMA(05,040)
      DIMENSION GAMMAG(360)
      DIMENSION GAMMK(1,040)
      DIMENSION INDXG(40)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LOADN(044)
      DIMENSION LSQ(044)
      DIMENSION LTMP(044)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION NPTS(60)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION R(11)
      DIMENSION RBAR(040)
      DIMENSION RCAP(01,11)
```

```
      DIMENSION RMOD(11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION SGMA1(044,044)
      DIMENSION SGMA2(05,44)
      DIMENSION SIGBL(360)
      DIMENSION SIGMZ(10)
      DIMENSION T(3,3)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(01)
      DIMENSION THTAY(01)
      DIMENSION TM(9),TV(3)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VLL(40)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION VXX(01,01)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZAP(40)
      DIMENSION ZROT(01)
      DIMENSION ZSTOR(800)
C
      COMMON /AL1BDA/ ABK
      COMMON /APXLDB/ LOADN
      COMMON /BETA1/BETA,MBETR,NBETC,PI,D,U,F,AS,ALPHA1,ALPHA2,RO
      COMMON /BETA3/OM,OMSQ,AO,AR,V,RHO,MU
      COMMON /CONT/ NA,NR,NANR,JA,JJ,NN,N,SIGN,II,NGJR
      COMMON /CONVGA/ EPSG,NWKRQ
      COMMON /CONVGB/ SGMA1,INDXG
      COMMON /CONVGC/ GAMMAG
      COMMON /DART1/ SGRATO
      COMMON /ELNTHS/ ELL(16)
      COMMON /ITRG/ ITRGX
      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(4,11),BFM(4,10)
      COMMON /MUVXY7/ TM,TV,DEL,VDT,RC,CAPPHI,AQ,AZ,YR,ZR,RP,IVAR
      COMMON /OUTDI/ NNTV,NEXPWK,NSIGRW,NMODR,NMODC
      COMMON /OUTDII/ NWKCLM
      COMMON /OUTIN/ IN,OUT
```

```
      COMMON /STEPXA/ WKPT,WW,IOUT,NOTTP1,KAT,NBC
      COMMON /SUBIB/ KX,LSQ,DSQ,DTMP,LTMP,ATMP,BTMP
      COMMON /SUBIC/ R,C,DTWOPI
      COMMON /SUBID/ I,IP1,IM1
      COMMON /SUBIE/ NAS
      COMMON /STPSZ/ NRATIO,NAA,LRGWKS,LIMLSS,LSWW
      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 /VLIMIT/ VLIM(11),VMLIM(16)
      COMMON /VLNTHS/ NALIM,VLL
      COMMON /WAKE1/ VOOMR,NUWKPT,VI,WKX,WKY,WKZ,COSB3,SINB3,NAS1,
     1NIBNA,NWKLST,NWKRW,NWKCL,NLP1,NLP2,NLP3,NLP4,TWOPI
      COMMON /WKCONT/ NWKPD
      COMMON /WKQ/ NUMXYZ
      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 /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /WK2C/ SIGBL
      COMMON /WK2GAM/ GAMFAC
      COMMON /WK4A/ VXX
      COMMON /ZCNTRL/ NZS
      COMMON /ZSS/ ZSTOR
C
      DATA BLANK/1H /
      DATA NWKX,NWKY,NWKZ/3HWKX,3HWKY,3HWKZ/
C
   30 FORMAT(4HOAFM/)
   31 FORMAT(4HOBFM/)
   32 FORMAT(33HOVLIM(NTV),VMLIM(NTVM*NIB),GAMFAC/)
   33 FORMAT(12HONALIM  ELIM/2X,I3,G16.7)
   34 FORMAT(12HOVLL(NIBRVM)/)
   35 FORMAT (23HONUMBER OF SMALL STEPS ,I5,24H   START SMALL STEPS AT ,
     1 I5)
  907 FORMAT(16X,3F8.8)
  906 FORMAT (29X,I5)
   16 FORMAT (20A4)
  901 FORMAT (19X,2X,7X,E11.8)
 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(1H //
     11H ,38HAMPLITUDE OF LATERAL CYCLIC PITCH, RAD,      12X,1(F9.4,5X)/
     21H ,42HLATERAL SHAFT TILT ANGLE, POS TO PORT, RAD,  8X,1(F9.4,5X)/
     31H ,43HAMPLITUDE OF LONGITUDINAL CYCLIC PITCH, RAD, 7X,1(F9.4,5X)/
     41H ,39HLONG FIRST HARMONIC FLAPPING ANGLE, RAD,     11X,1(F9.4,5X)/
     71H ,23HBLADE CONING ANGLE, RAD   ,                  27X,1(F9.4,5X)/
     61H ,42HLONG SHAFT TILT ANGLE, POS AFT FR VRT, RAD,  8X,1(F9.4,5X))
 8889 FORMAT (
     11H ,34HBLADE INBOARD AIRFOIL RADIUS, FEET,          16X,1(F9.4,5X)/
     21H ,41HOFFSET OF HINGE FM CNTR OF ROTATION, FEET,   9X,1(F9.4,5X)/
     91H ,43HDIRECTION OF ROTOR, POS IS COUNTERCLOCKWISE, 7X,1(F9.4,5X)/
     41H ,26HROTOR REFERENCE ANGLE, RAD,                  24X,1(F9.4,5X)/
     51H ,22HBLADE TWIST ANGLE, RAD,                      28X,1(F9.4,5X)/
     71H ,20HCHORD, DIMENSIONLESS,                        30X,1(F9.4,5X)/
     81H ,29HSHAFT TILT, POSITIVE AFT, RAD,               21X,1(F9.4,5X))
 8887 FORMAT (
     11H ,42HEXTRAPOLATED ANGLE OF ATTACK AT SHAFT, RAD,  8X,1(F9.4,5X)/
     21H ,29HX-AXIS ROTOR COORDINATE, FEET,               21X,1(F9.4,5X)/
     41H ,29HY-AXIS ROTOR COORDINATE, FEET,               21X,1(F9.4,5X)/
     51H ,42HLATERAL FIRST HARMONIC FLAPPING ANGLE, RAD,  8X,1(F9.4,5X)/
     61H ,29HZ-AXIS ROTOR COORDINATE, FEET,21X,1(F9.4,5X)////)
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
      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
      DEL=0.
      VDT=0.
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) NAA,LSWW
      READ (IN,906) WKPT,ITRGX
      READ (IN,906) NALIM
      READ (IN,906) NUMXYZ
      READ (IN,901) SGRATO
      READ (IN,901) VDOMR,ABK,OM,V,RHO
      NTV1=NTV-1
      NIBRVM=NTVM*NIB*NROT
      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(I),I=1,NTV),(VMLIM(I),I=1,NIBRVM)
      READ (5,901) GAMFAC
      READ (IN,901) ELIM
C
      ZNA=NA
      SINALS=SIN(ALFAS(1))
      COSALS=COS(ALFAS(1))
      CALL MANEUV(V,OM,ZNA,RREF,SINALS,COSALS)
C
      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        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
C
C        COMPUTE TIP PATH ANGLE FROM APPROXIMATE EQUATIONS OF MOTION OF
C             RIGID, SPRING-HINGED BLADE.
C
      MBETR=3
      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)
      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,PHO,RREF,RO,OM,V,CLA,ABK
C
      WRITE(OUT,8883)(ALFA1(I),I=1,NROT),(THTAY(I),I=1,NROT),
     1                (ALFA2(I),I=1,NROT),( BETA(3,I),I=1,NROT),
     2     (   BETA(1,I),I=1,NROT),( THTAX(I),I=1,NROT)
      WRITE(OUT,8889)(RZERO(I),I=1,NROT),(DELTA(I),I=1,NROT),
     1     (       DIR(I),I=1,NROT),(  PSIR(I),I=1,NROT)      ,
     2     (        AR(I),I=1,NROT),
     1     (     CHORD(I),I=1,NROT),( ALFAS(I),I=1,NROT)
      WRITE(OUT,8887)(   AO(I),I=1,NROT),( XROT(I),I=1,NROT),
     1                          (  YROT(I),I=1,NROT)      ,
     2                (BETA(2,I),I=1,NROT),(ZROT(I),I=1,NROT)
      WRITE(OUT,8890) TM,TV,DEL,VDT,RC,CAPPHI,AQ,AZ,YR,ZR,RP,IVAR
     1,NBC,NWKRQ,WW,NUWKPT,NTVM,NANRM,NUMXYZ,SGRATO,VOOMR
 8890 FORMAT(/1X,  19HMANEUVER PARAMETERS,// 6X,4HTM =,
     19(F10.7)/        6X,4HTV =,3F10.7,/ 5X,5HDEL =,F10.7/ 5X,5HVDT =,F1
     20.7/ 6X,4HRC =,F10.7/ 2X,8HCAPPHI =,F10.7/ 6X,4HAQ =,F10.7/ 6X,4HA
     37 =,F10.7/ 6X,4HYR =,F10.7/ 6X,4HZR =,F10.7/ 6X,4HRP =,F10.7/ 4X,
     46HIVAR =,I5//
     5 5X,15HPROGRAM CONTROL/
     6 5X,5HNBC =,I5/ 3X,7HNWKRQ =,I5/ 6X,4HWW =,I5/2X,8HNUWKPT =I5/4X,6
     7HNTVM =,I5/ 3X,7HNANRM =,I5/ 2X,8HNUMXYZ =,I5/ 2X,8HSGRATO =,F10.7
     8/ 3X,7HVOOMR =,F10.7/)
      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)
      WRITE (OUT,9875)((AFM(I,J),J=1,NTVMP2),I=1,NTVM)
      WRITE(6,31)
      WRITE (OUT,9875)((BFM(I,J),J=1,NTV1),I=1,NTVM)
      WRITE(6,32)
      WRITE (OUT,9875) (VLIM(I),I=1,NTV),(VMLIM(I),I=1,NIBRVM),GAMFAC
      WPITE(OUT,33)NALIM,ELIM
```

```
      WRITE (OUT,35) NAA,LSWW
      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=?
      NLP1=NROT
      NLP2=NIB
      NLP3=NTV
      NWKPD=0
      NPER=1
      NR=NTV1*NROT
      NANR=NA*NR
      NIBNA=NA/NIB
      NWSTRE=2
      NWR=NA*NREV
      LRGWKS=1
      NRATIO=NAA/NA
C
C        CONSTANTS USED TO CONTROL ARRAY SIZES
C
      NNTV=44
      NEXPWK=5
      NSIGRW=44
      NWKRW=11
      NWKCLM=44
      NMODR=54
      NMODC=16
      NNROT=1
      NNTVM=11
      NTVMX=4
C
C        COMPUTE LENGTHS FOR USE IN MODIFIED WAKE
C
      DO 3 I=1,NTVM
```

```
      DO 3 J=1,NIBRVM
      XM(I,J)=0.
      YM(I,J)=0.
    3 ZM(I,J)=0.
      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)
      WRITE (OUT,9875) (VLL(I),I=1,NIBRVM)
C
C         INIT FOR WAKE CALCULATIONS
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 13
      NWKLST=1
      NWKCL=NWKCL+1
   13 READ (IN,907) WKX(NWKLST,NWKCL),WKY(NWKLST,NWKCL),WKZ(NWKLST,
     1NWKCL)
      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,NEXPWK,NWKCLM)
      CALL MPRECT (NWKY,WKY,NWKRW,NWKCL,NEXPWK,NWKCLM)
      CALL MPRECT (NWKZ,WKZ,NWKRW,NWKCL,NEXPWK,NWKCLM)
  200 DO 18 I=1,NWKRW
      DO 18 J=1,NWKCL
      VXX(I,J)=0.
   18 VI(I,J)=0.
C
C         PERFORM A STEP
```

```
C
   20 NAS1=0
      NW=1
      II=1
      WKPT=3
      RETURN
      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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHAO(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION C(09)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION GAMMA(05,040)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION R(11)
      DIMENSION RBAR(040)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION T(03,03)
      DIMENSION TCOR(03,03)
      DIMENSION THTAX(01)
      DIMENSION THTAY(01)
      DIMENSION TR(09)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
```

```
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZAP(40)
      DIMENSION ZROT(01)
      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,MRETR,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,I14/(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)+MUCOS(M)
      Z(2,J)=Z(2,J)+MUSOS(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=A0(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
C
C        ENTER TYPICAL AZIMUTHAL STEP COMPUTATION
C
   70 IF (II.GT.1) GO TO 72
      CALL OVERLAY (5LWKOVL,3,0,6HRECALL)
      CALL OVERLAY (5LWKOVL,4,0,6HRECALL)
      CALL OVERLAY (5LWKOVL,5,0,6HRECALL)
      CALL OVERLAY (5LWKOVL,6,0,6HRECALL)
      CALL OVERLAY (5LWKOVL,7,0,6HRECALL)
      CALL OVERLAY (5LWKOVL,08,0,6HRECALL)
      GO TO 21
   72 II=0
      NW=NW+1
      DO 73 M=1,NROT
   73 PSI(M)=PSI(M)+DPSI*DIR(M)
      GO TO 21
      END
```

```
      SUBROUTINE MATINV
      DIMENSION A(040,040)
      DIMENSION B(040,1)
      DIMENSION INDEX(40,3)
      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)
      DIMENSION  N(8,36)
      L1=L-1
      IF(M.LT.18) K=3
      IF(K.GT.L1)GO TO 12
      DO 10 I=K,L1
      DO 10 J=1,M
   10 N(I,J)=1000.*R(I,J)
      WRITE(10)((N(I,J),I=K,L1),J=1,M)
   12 CONTINUE
      J1=0
      J2=0
      JSEC=0
    1 J1=J2+1
      J2=J1+15
      IF(J2.LE.M)GO TO 2
      J2=M
    2 JSEC=JSEC+1
      WRITE(6,3)NI,JSEC
    3 FORMAT (1H0,A4,1X,6HMATRIX,8X,7HSECTION,I3)
      WRITE(6,4)(I,I=J1,J2)
    4 FORMAT(2X,3HROW,I6,15I7/)
      DO 5 I=1,L
    5 WRITE(6,6)I,(R(I,K),K=J1,J2)
    6 FORMAT(I4,4X,16F7.3)
      IF(J2.LT.M)GO TO 1
      RETURN
      END
```

```
      SUBROUTINE MANEUV(V,OM,ZNA,RREF,SINALS,COSALS)
C
      DIMENSION SDEL(9),RDEL(3),SPHI(9),RPHI(3),CAA(9),RALPHA(3),
     1 SW(9),RW(3)
C
      COMMON /MUVXYZ/ SDEL,RDEL,DEL,VDT,RC,CAPPHI,AS,AR,YR,ZR,RP,IVAR
C
      EQUIVALENCE (SDEL(1),SPHI(1),CAA(1),SW(1)),
     1            (RDEL(1),RPHI(1),RALPHA(1),RW(1)),
     2            (DEL,DELT,AA,WDT)
C
      DATA G,PI/32.2,3.141593/
C
      WRITE(6,1000)
 1000 FORMAT(2X,19HMAVEUVER INPUT DATA)
      TWPINA=2.*PI/ZNA
      OMRREF=OM*RREF
      READ(5,6)IVAR
      GO TO (10,20,30,40,40),IVAR
C
C        STEADY TURNS
C
   10 READ (5,100) DELDOT,RC,F,AS
      WRITE(6,1001) DELDOT,RC,F,AS
 1001 FORMAT(2X,15HTURN   DELDOT =,G14.4,6H  RC =,G14.4,5H  F =,G14.4,6H
     1  AS =,G14.4)
      IF(RC.NE.0.)DELDOT=V/RC
      IF(DELDOT.NE.0.)RC=V/DELDOT
      IF(F.EQ.0.)GO TO 11
      DELDOT=G*SQRT(F**2-1.)/V
      RC=V**2/(G*SQRT(F**2-1.))
   11 DEL=DELDOT*2.*PI/(ZNA*OM)
      CAPPHI=ATAN(V**2/(RC*G))
      RC=RC/RREF
      CALL TURN (RDEL,SDEL,DEL,RC,CAPPHI,AS)
      GO TO 5
C
C        STEADY ROLLS
C
   20 READ (5,100) AR,DEL,YR,ZR
      WRITE(6,1002)  AR,DEL,YR,ZR
 1002 FORMAT(2X,11HROLL   AR =,G14.4,7H  DEL =,G14.4,6H  YR =,G14.4,6H
     1ZR =,G14.4)
      DELT=DEL*2.*PI/(ZNA*OM)
      VDT=V*(2.*PI/(ZNA*OM))
      YR=YR/RREF
      ZR=ZR/RREF
      VDT=VDT/RREF
      CALL ROLL(RPHI,SPHI,AR,DELT,YR,ZR,VDT)
      GO TO 5
```

```
C
C         SYMETRICAL PULL-UPS
C
   30 READ (5,100) AA,RP
      WRITE(6,1003)  AA,RP
 1003 FORMAT(2X,25HSYMMETRICAL PULL-UP  AA =,G14.4,6H  RP =,G14.4)
      AA=AA*2.*PI/(ZNA*OM)
      RP=RP/RREF
      CALL SUMPUP(RALPHA,CAA,AA,RP)
      GO TO 5
C
C      STEADY CLIMB OR STEADY FORWARD FLIGHT
C
   40 READ (5,100) W
      WRITE(6,1004)  W
 1004 FORMAT(2X,10HCLIMB  W =,G14.4)
      VDT=(V*COSALS-W*SINALS)*TWPINA/OMRREF
      WDT=(V*SINALS+W*COSALS)*TWPINA/OMRREF
      CALL STYCLB(RW,SW,VDT,WDT)
    5 RETURN
    6 FORMAT (33X,I1)
  100 FORMAT(28X,G11.7)
      END
```

```
      SUBROUTINE TURN (RDEL,SDEL,DEL,RC,CAPPHI,AS)
C
      DIMENSION AAS(9),TDEL(9),DDEL(9),AASINV(9),TZOINV(9),SDEL(9),
     1 RDEL(3),TZERO(9),ASA(9)
C
      COSAS=COS(AS)
      SINAS=SIN(AS)
      COSDEL=COS(DEL)
      SINDEL=SIN(DEL)
      CSCPHI=COS(CAPPHI)
      SNCPHI=SIN(CAPPHI)
      AAS(1)=COSAS
      AAS(2)=0.
      AAS(3)=SINAS
      AAS(4)=0.
      AAS(5)=1.
      AAS(6)=0.
      AAS(7)=-SINAS
      AAS(8)=0.
      AAS(9)=COSAS
      TDEL(1)=COSDEL
      TDEL(2)=SINDEL
      TDEL(3)=0.
      TDEL(4)=-SINDEL
      TDEL(5)=COSDEL
      TDEL(6)=0.
      TDEL(7)=0.
      TDEL(8)=0.
      TDEL(9)=1.
      DDEL(1)=SINDEL
      DDEL(2)=1.-COSDEL
      DDEL(3)=0.
      AASINV(1)=COSAS
      AASINV(2)=0.
      AASINV(3)=-SINAS
      AASINV(4)=0.
      AASINV(5)=1.
      AASINV(6)=0.
      AASINV(7)=SINAS
      AASINV(8)=0.
      AASINV(9)=COSAS
      TZERO(1)=1.
      TZERO(2)=0.
      TZERO(3)=0.
      TZERO(4)=0.
      TZERO(5)=CSCPHI
      TZERO(6)=SNCPHI
      TZERO(7)=0.
      TZERO(8)=-SNCPHI
      TZERO(9)=CSCPHI
```

```
TZOINV(1)=1.
TZOINV(2)=0.
TZOINV(3)=0.
TZOINV(4)=0.
TZOINV(5)=CSCPHI
TZOINV(6)=-SNCPHI
TZOINV(7)=0.
TZOINV(8)=SNCPHI
TZOINV(9)=CSCPHI
CALL GMPRD (AAS,TZERO,ASA,3,3,3,9,9,9)
CALL GMPRD (ASA,DDEL,RDEL,3,3,1,9,9,3)
CALL SMPY (RDEL,RC,3,1,3)
CALL GMPRD (ASA,TDEL,AAS,3,3,3,9,9,9)
CALL GMPRD (AAS,TZOINV,TDEL,3,3,3,9,9,9)
CALL GMPRD (TDEL,AASINV,SDEL,3,3,3,9,9,9)
RETURN
END
```

```
      SUBROUTINE ROLL (RPHI,SPHI,AR,DELT,YR,ZR,VDT)
C
      DIMENSION BMAT(9),BINV(9),PHI1(9),PHI2(9),SPHI(9),RVECT(3),
     1 RPHI(3),ASV(9)
C
      SINPHI=SIN(DELT)
      COSPHI=COS(DELT)
      ONEPHI=1.-COSPHI
      SINAR=SIN(AR)
      COSAR=COS(AR)
      BMAT(1)=COSAR
      BMAT(2)=0.
      BMAT(3)=-SINAR
      BMAT(4)=0.
      BMAT(5)=1.
      BMAT(6)=0.
      BMAT(7)=SINAR
      BMAT(8)=0.
      BMAT(9)=COSAR
      BINV(1)=COSAR
      BINV(2)=0.
      BINV(3)=SINAR
      BINV(4)=0.
      BINV(5)=1.
      BINV(6)=0.
      BINV(7)=-SINAR
      BINV(8)=0.
      BINV(9)=COSAR
      PHI1(1)=1.
      PHI1(2)=0.
      PHI1(3)=0.
      PHI1(4)=0.
      PHI1(5)=COSPHI
      PHI1(6)=SINPHI
      PHI1(7)=0.
      PHI1(8)=-SINPHI
      PHI1(9)=COSPHI
      PHI2(1)=1.
      PHI2(2)=0.
      PHI2(3)=0.
      PHI2(4)=0.
      PHI2(5)=ONEPHI
      PHI2(6)=-SINPHI
      PHI2(7)=0.
      PHI2(8)=SINPHI
      PHI2(9)=ONEPHI
      RVECT(1)=VDT
      RVECT(2)=YR
      RVECT(3)=ZP
      CALL GMPRD (BMAT,PHI2,ASV,3,3,3,9,9,9)
```

```
      CALL GMPRD (ASV,RVECT,RPHI,3,3,1,9,3,3)
      CALL GMPRD (BMAT,PHI1,ASV,3,3,3,9,9,9)
      CALL GMPRD (ASV,BINV,SPHI,3,3,3,9,9,9)
      RETURN
      END
```

```
      SUBROUTINE SUMPUP (RALPHA,CAA,AA,RP)
C
      DIMENSION CAA(9),RALPHA(3)
C
      SINAA=SIN(AA)
      COSAA=COS(AA)
      RALPHA(1)=SINAA*RP
      RALPHA(2)=0.
      RALPHA(3)=(1.-COSAA)*RP
      CAA(1)=COSAA
      CAA(2)=0.
      CAA(3)=SINAA
      CAA(4)=0.
      CAA(5)=1.
      CAA(6)=0.
      CAA(7)=-SINAA
      CAA(8)=0.
      CAA(9)=COSAA
      RETURN
      END
```

```
      SUBROUTINE GMPRD (A,B,R,N,M,L,M1,M2,M3)
C
      DIMENSION A(M1),B(M2),R(M3)
C
      IR=0
      IK=-M
      DO 10 K=1,L
      IK=IK+M
      DO 10 J=1,N
      IR=IR+1
      JI=J-N
      IB=IK
      R(IR)=0.
      DO 10 I=1,M
      JI=JI+N
      IB=IB+1
   10 R(IR)=R(IR)+A(JI)*B(IB)
      RETURN
      END
```

```
      SUBROUTINE SMPY (A,C,N,M,MX)
C
      DIMENSION A(MX)
C
      NM=N*M
      DO 1 I=1,NM
    1 A(I)=A(I)*C
      RETURN
      END
```

```
      SUBROUTINE STYCLB(RW,SW,VDT,WDT)
C
      DIMENSION RW(3),SW(9)
C
      RW(1)=VDT
      RW(2)=0.
      RW(3)=-WDT
      SW(1)=1.
      SW(2)=0.
      SW(3)=0.
      SW(4)=0.
      SW(5)=1.
      SW(6)=0.
      SW(7)=0.
      SW(8)=0.
      SW(9)=1.
      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,M2CC,M2CD,M115,IO,K
      DIMENSION A0(01)
      DIMENSION AR(01)
      DIMENSION BETA(3,1)
      DIMENSION D3(09)
      DIMENSION D33(040,040)
      DIMENSION XBETA(040)
      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,A0,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           = A0+AR*R+ALPHA1*SIN(OMT)+ALPHA2*COS(OMT)
C     ALPHA0 = 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,ALPHA0,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)
      MOO=R2DR1*RX*A1    +R3DR2*(VOM*A1AS-OM*WI-A2*RX   )+R4DR3*OMSQ*A1-R5
     1*OMSQ*A2
      MO1=R2DR1*(.75*VV*ALPHA1-V*WI+VV*AS)+R3DR2*VOM*A1*2.+R4DR3*(OMSQ*
     1ALPHA1-2.*OM*A2*V)
      MO2=(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 MODRM (NTV,NTVM,RCAP,RMOD,AFM,NNROT,NNTVM,NTVMX,M,NWKR)
C
      DIMENSION RCAP(NNROT,NWKR ),RMOD(NWKR ),AFM(NTVMX,NNTVM)
C
      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 GMS (I,NROT,NIB,NTVM,BFM,GAMMA,GAMMAM,NTVM1,NANR,NGJR,
     1 NMODR,NMODC)
      DIMENSION BFM(4,10)
      DIMENSION GAMMA(5,40)
      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(4,11)
      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(4,11)
      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
 1002 FORMAT (7HOMODCOX,9(1X,G11.4))
      RETURN
      END
```

```
      OVERLAY (WKOVL,3,0)
      PROGRAM 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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHAO(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION ATMP(11)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION BTMP(11)
      DIMENSION C(09)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION DSQ(040)
      DIMENSION DTMP(040)
      DIMENSION DUMX(108)
      DIMENSION DUMY(108)
      DIMENSION DUMZ(108)
      DIMENSION GAMMA(05,040)
      DIMENSION GAMMK(1,040)
      DIMENSION INDXG(40)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LSQ(044)
      DIMENSION LTMP(044)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION R(11)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
```

```
      DIMENSION SGMA1(044,044)
      DIMENSION SIGBL(360)
      DIMENSION SIGMZ(10)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XMM(54)
      DIMENSION XROT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YMM(54)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZMM(54)
      DIMENSION ZROT(01)
C
      COMMON /STPS7/ NRATIO,NAA,LRGWKS,LIMLSS,LSWW
      COMMON /STPDUM/ DUMX,DUMY,DUMZ
      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
      COMMON /MODWK3/ AFM(4,11),BFM(4,10)
      COMMON /STEPXA/ WKPT,WW,IOUT,NOTTP1,KAT,NBC
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 (NPER.EQ.4.AND.NRATIO.GT.1)
     1 CALL MODCOR(NROT,NIB,NTV,NTVM,X,Y,Z,XM,YM,ZM,
     2 AFM,NEXPWK,NNTV,NMODR,NANRM-NRATIO,NMODC)
      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.NE.1) GO TO 138
      N111=NWSTRE
      IF (NPER.EQ.4) N111=LRGWKS
      DO 152 IND=1,N111
      DUMX(IND)=XM(IND,JA)
      DUMY(IND)=YM(IND,JA)
  152 DUMZ(IND)=ZM(IND,JA)
      LIMLSS=NWSTRE
      IF (NPER.NE.4) GO TO 151
      IF(NRATIO.EQ.1) GO TO 151
      DO 153 IND=LRGWKS,NWSTRE
      XMM(IND)=XM(IND,JA)
      YMM(IND)=YM(IND,JA)
  153 ZMM(IND)=ZM(IND,JA)
      IND=LRGWKS-1
      IND1=LRGWKS-NRATIO
      XMM(IND)=XM(IND1,JA)
      YMM(IND)=YM(IND1,JA)
      ZMM(IND)=ZM(IND1,JA)
      CALL INTERP(NRATIO,LRGWKS,NWSTRE,LIMLSS,XMM,YMM,ZMM,DUMX,DUMY
     1 ,DUMZ)
  151 IF (MODWK.EQ.1) CALL SUBII(NPER,NA,INDX,NR,JAC,NIB,NROT,NTV1,VZ,
     1 JA,J,JB,LIMLSS)
  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(MSFT)*SGRATC
      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(6,9887) MSET
 9887 FORMAT(1X,8H MSET = ,I5)
      IF(NBC.NE.-2) GO TO 139
      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)
      RETURN
      END
```

```
      SUBROUTINE INTERP(NRATIO,NFIR,NLST,I1  ,X,Y,Z,XN,YN,ZN)
      DIMENSION X(54),Y(54),Z(54),XN(108),YN(108),ZN(108)
      DATA RM1,RP103,RP97,RM19/-.5555555E-2,.5722222,.5388889,-.1055555/
      IF(NRATIO.NE.2.AND.NRATIO.NE.3)GO TO 998
      IF (NFIR.LT.2) GO TO 997
      NLIM=NLST-NFIR-1
      XN(NFIR)=X(NFIR)
      YN(NFIR)=Y(NFIR)
      ZN(NFIR)=Z(NFIR)
      ISSMNR=NFIR-2
      I1=NFIR
      GO TO (999,20,30),NRATIO
   20 DO 25 I=1,NLIM
      I1=I1+1
      ISSMNR=ISSMNR+1
      ISS=ISSMNR+1
      ISSP1=ISSMNR+2
      ISSP2=ISSMNR+3
      XN(I1)=-.0625*(X(ISSMNR)+X(ISSP2))+.5625*(X(ISS)+X(ISSP1))
      YN(I1)=-.0625*(Y(ISSMNR)+Y(ISSP2))+.5625*(Y(ISS)+Y(ISSP1))
      ZN(I1)=-.0625*(Z(ISSMNR)+Z(ISSP2))+.5625*(Z(ISS)+Z(ISSP1))
      I1=I1+1
      XN(I1)=X(ISSP1)
      YN(I1)=Y(ISSP1)
   25 ZN(I1)=Z(ISSP1)
      GO TO 999
   30 DO 35 I=1,NLIM
      I1=I1+1
      ISSMNR=ISSMNR+1
      ISS=ISSMNR+1
      ISSP1=ISSMNR+2
      ISSP2=ISSMNR+3
      XN(I1)=RM1*X(ISSMNR)+RP103*X(ISS)+RP97*X(ISSP1)+RM19*X(ISSP2)
      YN(I1)=RM1*Y(ISSMNR)+RP103*Y(ISS)+RP97*Y(ISSP1)+RM19*Y(ISSP2)
      ZN(I1)=RM1*Z(ISSMNR)+RP103*Z(ISS)+RP97*Z(ISSP1)+RM19*Z(ISSP2)
      I1=I1+1
      XN(I1)=RM19*X(ISSMNR)+RP97*X(ISS)+RP103*X(ISSP1)+RM1*X(ISSP2)
      YN(I1)=RM19*Y(ISSMNR)+RP97*Y(ISS)+RP103*Y(ISSP1)+RM1*Y(ISSP2)
      ZN(I1)=RM19*Z(ISSMNR)+RP97*Z(ISS)+RP103*Z(ISSP1)+RM1*Z(ISSP2)
      I1=I1+1
      XN(I1)=X(ISSP1)
      YN(I1)=Y(ISSP1)
   35 ZN(I1)=Z(ISSP1)
      GO TO 999
  997 WRITE(6,9970)NFIR,NRATIO
 9970 FORMAT(1H0,5HISS =,I3,13H FOR NRATIO =,I3,13H IS INCORRECT)
      GO TO 999
  998 WRITE(6,9980)NRATIO
 9980 FORMAT(1H0,13HGIVEN RATIO =,I3,
     1 36H IS INCORRECT FOR SUBROUTINE INTERP )
```

```
999 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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHAO(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION AT(11)
      DIMENSION ATMP(11)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION BT(11)
      DIMENSION BTMP(11)
      DIMENSION C(09)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DLNTH(040)
      DIMENSION DNTH(011,040)
      DIMENSION DSQ(040)
      DIMENSION DTMP(040)
      DIMENSION GAMMA(05,040)
      DIMENSION GAMMK(1,040)
      DIMENSION INDXG(40)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LLNTH(044)
      DIMENSION LNTH(05,44)
      DIMENSION LSQ(044)
      DIMENSION LTMP(044)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION R(11)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
```

```
      DIMENSION SGMA1(044,044)
      DIMENSION SGMA2(05,44)
      DIMENSION SIGBL(360)
      DIMENSION SIGMZ(10)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
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,
     1 NWSTRE)
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 VZ(05,44)
C
      COMMON /STPSZ/ NROLD,NAA,LRGWKS,LIMLSS,LSWW
      COMMON /MODCNT/ NTVM,NXSTRE,NWR,NANRM,NIBRVM,NIBV
      COMMON /MODWK1/ GAMMA (54,16),R (1),A (05,16),VXM(54,16),
     1 VYM(54,16),VZM(54,16),XM(54,16),YM(54,16),ZM(54,16)
      COMMON /STPDUM/ X(108),Y(108),Z(108)
      COMMON /MODWK3/ AFM(4,11),BFM(4,10)
      COMMON /VLNTHS/ NALIM,VLL(40)
      COMMON /WK2B/ XA,YA,ZA,XB,YB,ZB,XC,YC,ZC
      COMMON /WK2C/SIGBL(360)
C
      EL=VLL(JA)
      TWOEL=2.*EL
      RSQ=R(1)*R(1)
      NSW=-1
      NWSTM1=NWSTRE-1
      I=NANRM-1
      NRATIO=1
   70 I=I+1
      JZ=I
      IF(NPER.NE.4)       GO TO 71
      IF(I.LT.LRGWKS)     GO TO 71
      NRATIO=NROLD
      JZ=(I-LRGWKS)/NRATIO+LRGWKS
   71 IP1=I+1
      IF(NRATIO.GT.1) GO TO 72
      XC=XM(IP1,JA)
      YC=YM(IP1,JA)
      ZC=ZM(IP1,JA)
      GO TO 73
   72 XC=X(IP1)
      YC=Y(IP1)
      ZC=Z(IP1)
   73 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
      GO TO 135
      NSW=-1
C
C        COMPUTE SQUARE ROOT OF RT AT THE APPROPRIATE STEP.
   87 IF(NRATIO.GT.1) GO TO 88
      XB=XM(I,JA)
      YB=YM(I,JA)
      ZB=ZM(I,JA)
      GO TO 89
   88 XB=X(I)
      YB=Y(I)
      ZB=Z(I)
   89 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(JZ,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
  103 VZ(1,J)=VZ(1,J)+(SGMAZ              )*GAMMA(JZ,JA)
      IF( NPER.NE.4) GO TO 111
      SGMBL = SGMAZ
C
C        CONVENIENT LOCATION TO COMPUTE SIGMA(J,PSII,I,JJ)
C
      DO 107 JQ=1,NTV1
      MODINX=INDXL+JQ
  107 SIGBL(MODINX)=-SGMBL*BFM(JB,JQ)+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
```

```
      OVERLAY (WKOVL,4,0)
      PROGRAM 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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHA0(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION GAMMA(05,040)
      DIMENSION GAMMAG(360)
      DIMENSION GAMMK(1,040)
      DIMENSION INDXG(40)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION RCAp(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION SGMA1(044,044)
      DIMENSION SIGBL(360)
      DIMENSION SIGMZ(10)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
```

```
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XSIMQ(040,040)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZAP(40)
      DIMENSION ZROT(01)
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 (1HO,     (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)((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
```

```
      OVERLAY (WKOVL,5,0)
      PROGRAM 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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHA0(01)
      DIMENSION ALPHAR(01)
      DIMENSION A0(01)
      DIMENSION AR(01)
      DIMENSION ATMP(11)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION BTMP(11)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION DSQ(040)
      DIMENSION DTMP(040)
      DIMENSION GAMMA(05,040)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LSQ(044)
      DIMENSION LTMP(044)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION THTAX(01)
      DIMENSION THTAY(01)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
```

```
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
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
```

```
      OVERLAY (WKOVL,6,0)
      PROGRAM 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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHAO(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION B(005,40)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION GAMMA(05,040)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LOADN(044)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION RBAR(040)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION THTAX(01)
      DIMENSION THTAY(01)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XX(01)
```

```
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
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,QSZ
      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
```

```
      OVERLAY (WKOVL,7,0)
      PROGRAM 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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHAO(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION GAMMA(05,040)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VXX(01,01)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XPOT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
```

```
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
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
C
      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,
     1LTMP,LSQ,MX,MY,MZ,MUCDP,MUSDS,MUCDS
C
      DIMENSION A(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION AQ(01)
      DIMENSION AR(01)
      DIMENSION B(005,40)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DNTH(011,040)
      DIMENSION GAMMA(05,040)
      DIMENSION LNTH(05,44)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION PSIR(01)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION ATMP(11)
      DIMENSION B(005,40)
      DIMENSION BTMP(11)
      DIMENSION C(09)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DNTH(011,040)
      DIMENSION DTMP(040)
      DIMENSION GAMMA(05,040)
      DIMENSION LNTH(05,44)
      DIMENSION LTMP(044)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION PSIR(01)
      DIMENSION R(11)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION SIGMX(10)
      DIMENSION SIGMY(10)
      DIMENSION SIGMZ(10)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
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(11),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 ONLY 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)=0.
    C(3)=0.
400 CONTINUE
    RETURN
    END
```

```
      SUBROUTINE TEST5
      REAL LBIG,LSMAL,LBSLS,LBSLSQ
      DIMENSION C(01)
      DIMENSION GAMMA(005,040)
      DIMENSION LSMAL(11)
      DIMENSION RCAP(01,11)
      DIMENSION RSMAL(01,44)
      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(40)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
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(11),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
      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
```

```
      OVERLAY (WKOVL,10,0)
      PROGRAM STEPX
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(05,44)
      DIMENSION ALFA1(01)
      DIMENSION ALFA2(01)
      DIMENSION ALFAS(01)
      DIMENSION ALFAT(01)
      DIMENSION ALPHA0(01)
      DIMENSION ALPHAR(01)
      DIMENSION AO(01)
      DIMENSION AR(01)
      DIMENSION B(005,40)
      DIMENSION BETA(03,1)
      DIMENSION CCLA(01)
      DIMENSION CHORD(01)
      DIMENSION DELTA(01)
      DIMENSION DIR(1)
      DIMENSION DNTH(011,040)
      DIMENSION GAMMA(05,040)
      DIMENSION GAMMAG(360)
      DIMENSION IO(01)
      DIMENSION KXX(01)
      DIMENSION LNTH(05,44)
      DIMENSION LX(01)
      DIMENSION MB(01)
      DIMENSION MUCDS(1)
      DIMENSION MUSDS(1)
      DIMENSION NPSI(1)
      DIMENSION PSI(1)
      DIMENSION PSIR(01)
      DIMENSION RCAP(01,11)
      DIMENSION RSMLL(01,44)
      DIMENSION RZERO(1)
      DIMENSION SIGBL(360)
      DIMENSION THTAX(01)
      DIMENSION THTAY(01)
      DIMENSION TM(3,3),TV(3),Q(3),HH(3)
      DIMENSION VI(01,01)
      DIMENSION VX(05,44)
      DIMENSION VY(05,44)
      DIMENSION VZ(05,44)
      DIMENSION WKX(01,01)
      DIMENSION WKY(01,01)
      DIMENSION WKZ(01,01)
```

```
      DIMENSION X(05,44)
      DIMENSION XROT(01)
      DIMENSION XX(01)
      DIMENSION Y(05,44)
      DIMENSION YROT(01)
      DIMENSION Z(05,44)
      DIMENSION ZROT(01)
C
C
      COMMON /STPSZ/ NRATIO,NAA,LRGWKS,LIMLSS,LSWW
      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(4,11),BFM(4,10)
      COMMON /MUVXYZ/ TM,TV,DEL,VDT,RC,CAPPHI,AQ,AZ,YR,ZR,RP,IVAR
      COMMON /OUTDI/ NNTV,NEXPWK,NSIGRW,NMODR,NMODC
      COMMON /OUTDII/ NWKCLM
      COMMON /CONVGC/ GAMMAG
      COMMON /WK2C/ SIGBL
      COMMON /WK4A/ VXX
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
      IF (NAS.GE.LSWW) LRGWKS=LRGWKS+1
      IF (NAS.NE.LSWW) GO TO 416
C
C        INIT FOR SMALL STEPS
C
      FRATIO=FLOAT(NRATIO)
      DPSI=DPSI/FRATIO
      NA=NAA
      NANR=NA*NR
```

```
      NIBNA=NA/NIB
      DEL=DEL/FRATIO
      VDT=VDT/FRATIO
      GO TO (4001,4002,4003,4004,4005),IVAR
 4001 CALL TURN (TV,TM,DEL,RC,CAPPHI,AQ)
      GO TO 4005
 4002 CALL ROLL (TV,TM,AZ,DEL,YR,ZR,VDT)
      GO TO 4005
 4003 CALL SUMPUP (TV,TM,DEL,RP)
      GO TO 4005
 4004 CALL STYCLB (TV,TM,VDT,DEL)
 4005 DO 5 M=1,NROT
      MUCDS(M)=MUCDS(M)/FRATIO
    5 MUSDS(M)=MUSDS(M)/FRATIO
C
C        TRANSPORT FULL MESH
C
  416 DO 420 M=1,NROT
      IF (IVAR.EQ.5) TV(1)=MUCDS(M)
      IF (IVAR.EQ.5) TV(3)=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
      HH(1)=X(K,J)
      HH(2)=Y(K,J)
      HH(3)=Z(K,J)
      CALL GMPRD (TM,HH,Q,3,3,1,9,3,3)
      X(L,J)=TV(1)+Q(1)+VX(K,J)*DPSI
      Y(L,J)=TV(2)+Q(2)+VY(K,J)*DPSI
  418 Z(L,J)=TV(3)+Q(3)+VZ(K,J)*DPSI
      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 MODCOR (NROT,NIB,NTV,NTVM,VX,VY,VZ,VXM,VYM,VZM,AFM,NEXPWK,
     2        NWKCLM,NMODR,NANRM,NVM2)
      DO 550 M=1,NROT
      IF (IVAR.EQ.5) TV(1)=MUCDS(M)
      IF (IVAR.EQ.5) TV(3)=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
      HH(1)=XM(K,J)
      HH(2)=YM(K,J)
      HH(3)=ZM(K,J)
      CALL GMPRD (TM,HH,Q,3,3,1,9,3,3)
      XM(L,J)=TV(1)+Q(1)+VXM(K,J)*DPSI
      YM(L,J)=TV(2)+Q(2)+VYM(K,J)*DPSI
  510 ZM(L,J)=TV(3)+Q(3)+VZM(K,J)*DPSI
      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.GE.NANRM+1)
     1CALL MODCOR(NROT,NIB,NTV,NTVM,X,Y,Z,XM,YM,ZM,AFM,NEXPWK,NNTV,
     1 NMODR,NANRM-1,NVM2)
C
C         OUTPUT BRANCH CONTROL
C
C     START NEW COMPUTATION WITH NO OUTPUT IF NBC.EQ.1
C
      WRITE (6,3) NAS,NW,NWSTRE
      IF(NBC.NE.-1) GO TO 490
      WRITE(7,900) NAS,PSI(1)
      WRITE(7,901) (( X(I,J),Y(I,K),Z(I,K),I=1,NW),J=NTVM,NIBRV,NTVM)
  900 FORMAT(I3,G14.4)
  901 FORMAT(16F5.2)
  490 CONTINUE
      IF (NAS.GE.WW) NWKPD=4
      IF (NWKPD.EQ.4) NPER=4
      II=0
      IF (KAT.EQ.1) STOP
      IF (NAS.GE.WW+NIBNA) STOP
      IF(NAS.GT.(2*NANRM).AND.NBC.EQ.-10) STOP
      IF(NBC.LE.1)GO TO 440
  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
```

```
  3 FORMAT (30HOTHE NUMBER OF WAKE POSITIONS ,3I5)
902 FORMAT (2I3/(8F10.7))
    END
```

Machine Compatibility

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 3.0. The program is standard FORTRAN IV and is also WATFIV compatible.

Recommended CDC 6600 Overlay Structure

The recommended CDC 6600 overlay structure is that contained in this listing. For execution on an IBM 360/65 the overlay statements may be replaced by subroutines by the following types of statements:

```
PROGRAM XXX by SUBROUTINE XXX
```

and

```
CALL OVERLAY (5LSKOVL,N,M,6HRECALL) by CALL XXX
```

where XXX is the appropriate subprogram or subroutine name.

```
      PROGRAM MAIN
C     MAIN PROGRAM BLD34
      DIMENSION COSARY(36)
      DIMENSION SNCSIA(36),SNCSIB(36),VBINT(10,36)
      DIMENSION SCTRMA(10,36)
      DIMENSION ASSF(20),ASSL(20),ASST(20)
      DIMENSION VX(18,37),WX(18,37)
      DIMENSION CSIX(740),CSIA(740),CSIB(740)
      DIMENSION DUMBD4(4286),DOMBD4(0716)
      DIMENSION SIGKJA(20,20)
      DIMENSION SIGKJB(20,20)
      DIMENSION SINEBJ(36)
      DIMENSION SINEJ(36)
      DIMENSION SAVE(360),WBR(10,36),EL(10,36)
      DIMENSION HDOT(10,36),PHI(10,36),THET(10,36)
      DIMENSION ELNTA(18),ELNTB(01),EMAA(18),EMAB(01),EIXA(18),EIXB(01)
      DIMENSION EIZA(18),EIZB(01)
      DIMENSION RBL(10),RBLA(10),RBLB(10)
      DIMENSION XINA(18),XINB(01),DPHA(18),DPHB(01),EIYA(18),EIYB(01)
      DIMENSION UNKWN(360),SV3(360)
      DIMENSION EPA(18),EPB(01),DLA(18),DLB(01),ZAA(18),ZAB(01)
      DIMENSION WDOT(10,36),WPHI(10,36),WTHET(10,36)
      DIMENSION FORCX(10,36),FORCZ(10,36),EMOME(10,36),YINA(18),YINB(01)
      DIMENSION WFX(10,36),WFZ(10,36),WEMO(10,36),BD1(15),BD2(15)
      DIMENSION BI(10),BET(10)
      DIMENSION VDOT(18,37),XDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION OMEGA(20),SIGA(20),AVA(18,20),AWA(18,20),APHIA(18,20),
     1 ASIA(18,20), ATHEA (18,20),ATA(18,20), AMZA(18,20),AVYA(18,20),
     2 AMYA(18,20),AVZA(18,20),CSIDTA(1,36)
      DIMENSION OMEGB(20),SIGB(20),AVB(01,20),AWB(01,20),APHIB(01,20),
     1 ASIB(01,20), ATHEB(01,20),ATB(01,20),AMZB(01,20),AVYB(01,20),
     2 AMYB(01,20),AVZB(01,20),CSIDTB(1,36)
      DIMENSION DUMSDA(3404),DUMSDB(1)
      DIMENSION NACT(10),NBCT(10),NBSV(10),JCYC(10)
      DIMENSION  NVBINT (10,36)
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))
      EQUIVALENCE (DOMBD4(1), ELNTB(1))
C
```

```
      EQUIVALENCE (DUMSDA(1),VDOT(1,1))
C
      COMMON /BLD4X1/ ELNTA, EMAA, EIXA, RBLA, XINA, DPHA, EIYA, EPA,
     1DLA, ZAA, YINA, OMEGA, PSIRA, SIGA, AVA, AWA, APHIA, ASIA, ATHEA,
     2ATA, AMZA, AVYA, AMYA, AVZA, BD1, RWKA, CSALA, NMA, CSIDTA,
     3SIGKJA, DAMPCA
      COMMON /DUMDCM/ ELNTB, EMAB, EIXB, RBLB, XINB, DPHB, EIYB, EPB,
     1DLB, ZAB, YINB, OMEGB, PSIRB, SIGB, AVB, AWB, APHIB, ASIB, ATHEB,
     2ATB, AMZB, AVYB, AMYB, AVZB, BD2, RWKB, CSALB, NMB, CSIDTB,
     3SIGKJB, DAMPCB
      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 /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
      COMMON /VORTEX/ VBINT
      COMMON /MANV1/ SCTRMA
      COMMON /MANV2/ SNCSIA,FGF
      COMMON /MANV3/ NAIR
      COMMON /MANV4/ ASSF,ASSL,ASST
      COMMON /MANV5/ ALFDTM, PHIDTM
      COMMON /VORINT/ NACT,NBCT,NBSV,CLMAX,DELCL2,WCOM,CMST2,JCYCLE,JCYC
     1,APMAX
C
C        ARRAY DIMENSIONING VARIABLES
C
      NDUMRY=4286
      NDIMRY=716
      NCSIRY=740
      NSOBRY=3404
      NSNCRY=36
      NVBIRY=10
C
C     INIT A AND B ROTOR SWAP STORAGE
C
      EPSV=0.
```

```
      ISW=2
      ISW=1
      DO 2384 I=1,NDUMRY
 2384 DUMBD4(I)=0.
      DO 5000 I=1,NDOMRY
 5000 DOMBD4(I)=0.
      DO 2344 I=1,NCSIRY
      CSIA(I)=0.
      CSIB(I)=0.
 2344 CSIX(I)=0.
      DO 2389 I=1,NSDBRY
C     DUMSDB(I)=0.
 2389 DUMSDA(I)=0.
      DO 2390 I=1,NSNCRY
      SNCSIA(I)=0.
 2390 SNCSIB(I)=0.
      DO 2391 I=1,NVBIRY
      DO 2391 J=1,NSNCRY
      NVBINT(I,J)=0
 2391 VBINT(I,J)=0.
      DO 9934   I=1,NVBIRY
 9934 NACT(I)=0
C
      IN=5
      NOUT=6
      IT7=8
      IT8=7
C
C     READ MODEL PARAMETERS
C
      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,NIP
      READ (IN,900) NVORTX,NAIR
      READ(IN,901)AMU,ALL1,ALL2,ALL2R,ATIME
      READ(IN,901) CPOMG,ROAIR
      READ (IN,901) XINPT,FINPT
      READ(IN,901) SIGLM,WBRLM
      READ (IN,901) WCLIMB,ALFDTM,PHIDTM,ALFRM,FGF
      IF(NVORTX.EQ.1)  READ( IN ,901)((VBINT(I,J),I=1,NR),J=1,NA)
      IF(NVORTX.EQ.1) READ(IN,900) JCYCLE
      IF(NVORTX.EQ.1) READ(IN,901) CLMAX,DELCL2,WOOM,CMST2,APMAX
      READ (IN,901) (ASSF(I),ASSL(I),ASST(I),I=1,MAXMO)
C
      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 OUT INPUTS
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
      WRITE(NOUT,9217) WCLIMB,ALFDTM,PHIDTM,ALFRM,FGF
      IF(NVORTX.EQ.1) WRITE(6,9218)  ((VBINT(I,J),I=1,NR),J=1,NA)
      IF(NVORTX.EQ.1) WRITE(6,900)  JCYCLE
      IF(NVORTX.EQ.1) WRITE(6,200)  CLMAX,DELCL2,WOGM,CMST2,APMAX
      WRITE(NOUT,9219)(I,ASSF(I),ASSL(I),ASST(I),I=1,MAXMO)
      WRITE(NOUT,9211)
C
C
      PI=3.1415926
      TWOPI=2.*PI
      DSI=TWOPI/NA
      NRPI=NR+1
      NAPI=NA+1
      NWPI=NW+1
      IJ=NA*NR
      IT3=1
      I1=1
      I2=NR1
      JIP=1
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
C
 9214 CNVRT=PI/180.0
      CALL CONV(CNVRT,THETA,ACA,BCA,ALPTA,DPHA,NMA,NMAS)
      WBARA=WBAR(AMU,ALPTA,CTA)
      WCLIMB=WCLIMB/(CPOMG*RA)
      ALFDTM=ALFDTM/CPOMG
      PHIDTM=PHIDTM/CPOMG
      ALFRM=COS(ALFRM*CNVRT)
      FGF=-FGF*32.2
      AMSNA=AMU*SIN(ALPTA)
      CSALA=COS(ALPTA)
      AMCSA=AMU*CSALA
      AMSNB=0.
C
```

```
      IF (NBL.NE.2) GO TO 38
      CALL CONV(CNVRT,THETB,ACB,BCB,ALPTB,DPHB,NMB,NMAS)
      WBARB=WBAR(AMU,ALPTB,CTB)
      AMSNB=AMU*SIN(ALPTB)
      CSALB=COS(ALPTB)
      AMCSB=AMU*CSALB
C
C     COMPUTE DISTANCES L SUB I J ACCORDING TO FORMULA PAGE III-3
C
   38 IX1=NR*NBL
      CSI=-DSI*DIRA+PSIRA
      SINALS=SIN(ALPTA)
      DO 40 J=1,NA
      CSI=CSI+DSI*DIRA
      SINEJ(J)=SIN(CSI)
      COSARY(J)=COS(CSI)
   40 SNCSIA(J)=SINEJ(J)*SINALS
C
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
      SCTRMA(I,J)=-WCLIMB+RBL(I)*(ALFDTM*COSARY(J)+PHIDTM*SINEJ(J)*
     1 ALFRM)
   50 EL(I,J)=DSI*(RBL(I)+AMCSA*SINEJ(J)*DIRA)
C
      IF(NBL-1) 143,143,144
  144 CSI=-DSI*DIRB+PSIRB
      SINALS=SIN(ALPTB)
      DO 55 J=1,NA
      CSI=CSI+DSI*DIRB
      SINEBJ(J)=SIN(CSI)
      COSARY(J)=COS(CSI)
   55 SNCSIB(J)=SINEBJ(J)*SINALS
      DO 59 I=NR2,NR
      BET(I)=BET(I)*CNVRT
      DO 59 J=1,NA
      SCTRMA(I,J)=-WCLIMB+RBL(I)*(ALFDTM*COSARY(J)+PHIDTM*SINEBJ(J)*
     1 ALFRM)
   59 EL(I,J)=DSI*(RBL(I)+AMCSB*SINEBJ(J)*DIRB)
   10 JIP=JIP+1
C
C     INITIALIZE COLUMN VECTOR OF UNKNOWNS FOR OVERALL ITERATION
C
  143 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
C
        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  ZY=RA
        ZQ=RA
C
C       ENTRY POINT FOR NEXT ITERATION
C
    85  CT1=CTA
        ALPT1=ALPTA
        GO TO (2004,2000),ISW
 2004   CALL BLD3
 2000   CONTINUE
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)
    93  WEMO(I,J)=EMOME(I,J)
        DO 931 I=1,NR1
931     RBLA(I)=RBL(I)
        RER=0.
        CSQ=0.
C
C  ENTER BLD4 WITH PARAMETERS FOR FIRST ROTOR
C
        DO 2345 I=1,NCSIRY
 2345   CSIX(I)=CSIA(I)
C
```

```
      CALL BLD4
C
C ASSIGN CALCULATED VALUES FROM BLD4 TO CORRECT ARRAYS
C
      DO 2346 I=1,NCSIRY
      RER=RER+(CSIX(I)-CSIA(I))*(CSIX(I)-CSIA(I))
      CSQ=CSIX(I)*CSIX(I)+CSQ
 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)
   94 THET(I,J)=WTHET(I,J)
C
C IF ONLY ONE ROTOR IS USED GO TO 85
C
      IF (NBL-1) 9411,9411,86
C
C ASSIGN CORRECT FORCES FOR INPUT TO BLD4
C
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)
   95 WEMO(L,J)=EMOME(I,J)
      DO 951 I=NR2,NR
      L=I-NR1
951   RBLB(L)=RBL(I)
C
C ENTER BLD4 WITH PARAMETERS FOR SECOND ROTOR
C
      DO 2347 I=1,NCSIRY
 2347 CSIX(I)=CSIB(I)
      DO 2387 I=1,NSDBRY
      TEMPX=DUMSDA(I)
      DUMSDA(I)=DUMSDB(I)
 2387 DUMSDB(I)=TEMPX
      DO 2386 I=1,NDUMRY
      TEMPX=DUMBD4(I)
      DUMBD4(I)=DOMBD4(I)
 2386 DOMBD4(I)=TEMPX
      DO 3000 I=1,NSNCRY
      TEMPX=SNCSIB(I)
      SNCSIB(I)=SNCSIA(I)
 3000 SNCSIA(I)=TEMPX
C
      CALL BLD4
C
C ASSIGN CALCULATED VALUES FROM BLD4 TO CORRECT ARRAYS
```

```
C
      DO 2385  I=1,NDUMRY
      TEMPX=DUMBD4(I)
      DUMBD4(I)=DUMBD4(I)
 2385 DUMBD4(I)=TEMPX
      DO 2388 I=1,NSDBRY
      TEMPX=DUMSDA(I)
      DUMSDA(I)=DUMSDB(I)
 2388 DUMSDB(I)=TEMPX
      DO 2348 I=1,NCSIRY
      RER=RER+(CSIX(I)-CSIB(I))*(CSIX(I)-CSIB(I))
      CSQ=CSIX(I)*CSIX(I)+CSQ
 2348 CSIB(I)=CSIX(I)
      DO 3001 I=1,NSNCRY
      TEMPX=SNCSIB(I)
      SNCSIB(I)=SNCSIA(I)
 3001 SNCSIA(I)=TEMPX
      I1=0
      DO 941 I=NR2,NR
      I1=I1+1
      DO 941 J=1,NA
      WDOT(I,J)=WDOT(I1,J)
      PHI(I,J)=WPHI(I1,J)
  941 THET(I,J)=WTHET(I1,J)
 9411 EPSR=SQRT(RER/CSQ)
      IF(IT3-3) 3028,3026,3021
 3021 IF(EPSR.GT.EPSV) GO TO 3030
 3026 EPSV=EPSR
 3028 WRITE(NOUT,940) IT3,EPSR
      IF(EPSR.GT.ALL2R) GO TO 3010
 3030 IF(IT3.EQ.1000) GO TO 4000
      IT3=-500
      WRITE(NOUT,940) IT3,EPSR
 3010 IT3=IT3
      IF(IT3.GT.NIT3) IT3=1000
  940 FORMAT(/,31X,5HIT3 =,I5,19X,18HRESPONSE    ERROR =,G15.6/)
      IF(IT3.EQ.1000) GO TO 4000
      GO TO 85
 4000 IF (JIP.GE.NIP) STOP
      READ (IN,901) THETO,AC,BC
      THETA=CNVRT*THETO
      ACA=CNVRT*AC
      BCA=CNVRT*BC
      WRITE (NOUT,4200) THETO,AC,BC
      IF (NBL-1) 4010,4010,4005
 4005 READ (IN,901) THETO,AC,BC
      THETB=CNVRT*THETO
      ACB=CNVRT*AC
      BCB=CNVRT*BC
      WRITE (NOUT,4200) THETO,AC,BC
```

```
 4010 IT3=1
      DO 1 I=1,NCSIRY
      CSIA(I)=0.
      CSIB(I)=0.
  1   CSIX(I)=0.
      GO TO 10
C
  200 FORMAT (10(1X,E12.5)/)
  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,16HADVANCE 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(/))
 9216 FORMAT(39X,31HLIMIT ON OFF-DIAGONAL SIGMAS = ,F7.4,/,
     1       35X,35HLIMIT ON WAKE-INDUCED VELOCITIES = ,F7.4,//)
 9217 FORMAT (14HOCLIMB RATE = ,E14.7/14H PITCH RATE = ,E14.7/
     1 13H ROLL RATE = ,E14.7/10H ALPHAR = ,E14.7/18H GRAVITY FACTOR = ,
     2 E14.7)
 9218 FORMAT (32HOVORTEX BLADE INTERACTION PHASES//(8E15.6))
 7994 FORMAT (6HOFORCZ//)
 7995 FORMAT (6HOFORCX//)
 7996 FORMAT (6HOEMOME//)
 9219 FORMAT(2X,2H K,9X,5H ASSF,15X,5H ASSL,15X,5H ASST/(I5,3E20.10/))
 4200 FORMAT (20X,3(2X,G15.7))
      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(10)
      DIMENSION SIGKJ(20,20)
      DIMENSION BI(10),BET(10),ELNTH(18),EIX(18),EIY(18),XINR(18),
     1 YINR(18),EMAS(18),DPHI(18),EPS(18),DLZ(18),ZA(18),OMEGA(20),
     2 EIZ(18),SIG(20)
      DIMENSION AMY(18,20),AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),AVZ(18,20)
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
      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
C
   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
C
   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
C     THIS READ COULD BE REPLACED WITH A IT8 UNFORMATED READ
C
      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
C     REPLACE THIS READ WITH IT7
C
      READ(5,589)(RBL(I),I=I1,I2)
      RETURN
C
  900 FORMAT (16I5)
  901 FORMAT (8F10.7)
  902 FORMAT(10E8.7)
589   FORMAT(10F8.0)
588   FORMAT(5(G12.7,3X),/,5G15.7)
      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(10)
      DIMENSION SIGKJ(20,20)
      DIMENSION BI(10),BET(10),ELNTH(18),EIX(18),EIY(18),XINR(18),
     1 YINR(18),EMAS(18),DPHI(18),EPS(18),DLZ(18),ZA(18),OMEGA(20),
     2 EIZ(18),SIG(20)
      DIMENSION AMY(18,20),AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),AVZ(18,20)
C
      COMMON /IO/ IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
C
      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
C
  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=,F12.5)
  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,16F8.2)
       END
```

```
      SUBROUTINE CONV(CNVRT,THETO,AC,BC,ALPHT,DPHI,NM,NMAS)
C
      DIMENSION DPHI(18)
C
      COMMON /A1/ NBL,NB,NR,Nw,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
C
      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
C     WBAR RETURNS EITHER WBARA OR WBARB FOR USE IN SUBROUTINE START
C
      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,THETO,DSI,CSI,
     1 BI,BET,RBL,UMSQ,L,M,AV,CSIDT,R,PSIR,DIR,HDOT,PHI,THET,NA,NRNT)
C
      DIMENSION BI(10),BET(10),RBL(10)
      DIMENSION HDOT(10,36),PHI(10,36),THET(10,36),CSIDT(1,36),
     1CSI(20,37)
C
      COMMON /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPOMG,IT3
C
C     START SETS INITIAL VALUES FOR HDOT, PHI, THET
C
      FNA=NA
      HOLD=WBARX+AMU*ALPHT
      AMUSQ=AMU*AMU
      AO=(.25*(1.+AMUSQ)*THETO+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*THETO/3.+(.25+.375*AMUSQ)*AC-BC*BI(L)/3.-.5*AMU*HOLD+
     1 .5*AMU*BET(M ))*AKI
      CO=.25*AKI
      C1=AMU*AKI/3.
      DO=UMSQ-BI(L)*AKI/3.
      D1=-.5*AMU*BI(L)*AKI
      D2=.25*AMUSQ*AKI
      ETA1=((DO-1.)*(BCIR*DO-D1*AO)-(AI*DO-C1*AO)*(.5*D2-CO))/
     1 ((DO-1.)*(DO*(DO-1.)-.5*D1*D1)-(.5*D2-CO)*(DO*(.5*D2+CO)
     2 -.5*D1*C1))
      CSI1=(AI-C1*AO/DO-(.5*D2+CO-D1*C1/(2.*DO))*ETA1)/(DO-1.)
      CSI0=(AO-.5*D1*ETA1)/DO
      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)=THETO+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.
  100 CSIDT(1,J)=0.
C
      WRITE(6,22)
      WRITE(6,200)((THET (I,J),J=1,NA),I=L,M)
      RETURN
```

```
C
20      FORMAT(    6HOHDOT ,//)
200     FORMAT(10(1X,E12.5),/)
21      FORMAT(    6HOPHI  ,//)
22      FORMAT(    6HOTHET ,//)
23      FORMAT (6HOCSIOT//)
 9876   FORMAT (1HO,6E15.7)
        END
```

```
      SUBROUTINE HARMN (NM,NA,DSI,F,MTYP,NMAS,NROW,NCOL)
C
      DIMENSION A(18,19),B(18,19),F(NROW,NCOL)
C
      WRITE(6,900)
      NCNT=1
      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=NAU2
  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,NAU2
      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)
      NCNT=NCNT+1
      IF (NCNT.LT.2) GO TO 270
      RETURN
C
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))
  920 FORMAT (1H ,///,51X,29HHARMONIC ANALYSIS, POLAR FORM)
      END
```

```
      SUBROUTINE BLD3
C
C
C
C     THIRD PROGRAM IN BLADE LOADS SEQUENCE
C     SOLUTION OF EQUATIONS BY ITERATION
C
      DIMENSION USAVE(360),CO(10,36),CO1(10,36),SIGMA(360)
      DIMENSION UNKWN(360),SV3(360),RBL(10),FORCX(10,36),FORCZ(10,36)
      DIMENSION EMOME(10,36),HDOT(10,36),PHI(10,36),THET(10,36),BI(10)
      DIMENSION BET(10),SAVE(360),WBR(10,36),EL(10,36)
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 ,CPCMG,IT3
      COMMON /B3/ DSI,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLAD3/ CMPO,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
 3396 IF(IT3.NE.1) GO TO 370
3398   DO 3397 I=1,NANR
3397  SV3(I)=0.0
370   IF(KTEST)373,373,372
  373 DO 3333 M=1,NRNA
 3333 SIGMA(M)=0.
  372 IF (KTEST) 20,20,2
2     REWIND IT7
      REWIND IT8
C
C     DEFINE NECESSARY CONSTANTS
C
   20 CNOTP=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
C
C     TEST ON OUTER AND MIDDLE ITERATION
C
      IF(IT3.NE.1) GO TO 50
40    IF(IT2-1)41,41,50
```

```
41      IF(KTEST)42,42,43
   43 READ (IT8) (UNKWN(K),K=1,NANR)
      IF(KTEST.GT.1) READ(5,901) (UNKWN(K),K=1,NANR)
 901  FORMAT(8F10.6)
      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
C     CHECK FOR CONVERGENCE ON OVERALL ITERATION
C
      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
C     CUTOFF ON OVERALL ITERATION FOR SEVERE DIVERGENCE
C
3021  IF(EPLON-ERRSV)3026,3026,303
3026  ERRSV=EPLON
3028  WRITE(NOUT,940)IT3,EPLON
      IF(EPLON-ALL2)303,303,310
 303  IF(IT3.LT.0) IT3=1000
      WRITE(NOUT,940)IT3,EPLON
  310 IT3=IT3+1
      IF (IT3.GT.NIT3) IT3=1000
      IF(IT3.EQ.1000) WRITE(NOUT,993) ((WBR  (I,J),I=1,NR),J=1,NA)
 993  FORMAT(3X,3HWBR//,(8G15.5))
      IF (IT3.EQ.1000) CALL HARMN (NR,NA,DS1,WBR,6,0,10,36)
      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 SQ(10,36),AP(10,36)
      DIMENSION SCTRMA(10,36),VBINT(10,36)
      DIMENSION USAVE(360),CO(10,36),CO1(10,36),SIGMA(360)
      DIMENSION UNKWN(360),SV3(360),RBL(10),FORCX(10,36),FORCZ(10,36)
      DIMENSION EMOME(10,36),HDOT(10,36),PHI(10,36),THET(10,36),BI(10)
      DIMENSION BET(10),SAVE(360),WBR(10,36),EL(10,36)
C
      COMMON /APSQ/ AP,SQ
      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/ CMPO,AMNA2,AMNB2,IT2,ON2OS,IT1,CMPAR
      COMMON /SGSLS/ SIGLM,WBRLM
      COMMON/PRNT/NPRNT
      COMMON /VORTEX/ VBINT
      COMMON /MANV1/ SCTRMA
      COMMON /MANV3/ NAIR
C
C     DO LOOPS TO COMPUTE CERTAIN COEFFICIENTS AS FUNCTIONS OF RADIAL
C     POSITION AND AZIMUTH WHICH APPEAR IN EXPRESSIONS FOR UNKNOWNS,
C
      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
      WBR(I,J)=WBAR(AMU,ALPTA,CTA)
      GO TO 3139
```

```
3138  AMUSN=AMSNB
      AMSN2=AMNB2
      EMT=EMTB1
      WBR(I,J)=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 INDEX=(J-1)*NR+I
   73 INJM1=(JM1-1)*NR+I
      IF(KTEST.EQ.0) GO TO 80
      WBR(I,J)=0.
C
C     RECOMPUTE DOWNWASH VALUES ACCORDING TO PAGE III-4, IF IT2=1.
C
      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 80
      WBR(I,J)=WBR(I,J)*WBRLM/WABS
C
C     COMPUTE U ACCORDING TO PAGE III-3 AND V ACCORDING TO PAGE III-4
C
   80 U=EL(I,J)/DSI
      V=HDOT(I,J)-AMUSN-WBR(I,J)+SCTRMA(I,J)
      SQTUV=SQRT(U*U+V*V)
      SQ(I,J)=SQTUV
      ARG=0.
      IF (SQTUV.NE.0.) ARG=ATAN2(V,U)
      APHIJ=BET(I)+THET(I,J)+ARG
      AP(I,J)=APHIJ
C
C     SUBROUTINE SERIES COMPUTES VALUES OF CLIFT=LIFT COEFFICIENT,
C     ASLOP=LIFT CURVE SLOPE, CMOME=MOMENT COEFFICIENT, CDRAG=DRAG
C     COEFFICIENT
C
      IF (NAIR.EQ.1) CALL NACA15 (APHIJ,CLIFT,CDRAG,CMOME,ASLOP,EMT,U)
      IF (NAIR.EQ.1) GO TO 81
      CALL SERIES (I,J,1,EMT,U,V,APHIJ,CLIFT,ASLOP,CMOME,CDRAG)
C
   81 IF (VBINT(I,J).NE.0.) CALL INTRAT (I,J,CLIFT,CMOME,NA,DSI,APHIJ)
```

```
      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))
C
   100 CONTINUE
C
      IF(NPRNT.LT.1) GO TO 1000
      WRITE (6,9998) ((SQ(I,J),J=1,NA),I=1,NR)
      WRITE (6,9997) ((AP(I,J),J=1,NA),I=1,NR)
C
      IF(NPRNT.LT.1) GO TO 1000
      WRITE(6,9999)((WBR(I,J),J=1,NA),I=1,NR)
 1000 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
C
      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
C     END UPDATE OF GAMMAS
C
      IF(KTEST)304,304,113
  113 REWIND IT7
C
C     CHECK FOR CONVERGENCE ON INNERMOST ITERATION
C
  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     CHECK FOR CONVERGENCE ON SECOND LEVEL ITERATION
C
  360 IT1=0
      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 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/)
 9997 FORMAT(6HOAPHIJ/(9G13.5))
 9998 FORMAT(6HOSQTUV/(9G13.5))
 9999 FORMAT(/1HO,3HWBR/(9G13.5))
      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 SQ(10,36),AP(10,36)
      DIMENSION SCTRMA(10,36),VBINT(10,36)
      DIMENSION USAVE(360),CO(10,36),CO1(10,36),SIGMA(360)
      DIMENSION UNKWN(360),SV3(360),RBL(10),FORCX(10,36),FORCZ(10,36)
      DIMENSION EMOME(10,36),HDOT(10,36),PHI(10,36),THET(10,36),BI(10)
      DIMENSION BET(10),SAVE(360),WBR(10,36),EL(10,36)
C
      COMMON /PUNCH/ NPCH
      COMMON /APSQ/ AP,SQ
      COMMON /MANV1/ SCTRMA
      COMMON /MANV3/ NAIR
      COMMON /VORTEX/ VBINT
      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/ DST,IJ,KTEST,AMSNA,AMSNB,EMTA1,EMTB1,ALL1,ALL2,ROAIR,
     1 RA1,RB1,ERRSV
      COMMON /BLAD3/ CMPO,AMNA2,AMNB2,IT2,ON2DS,IT1,CMPAR
C
C     COMPUTATION FOR BLADE LOADS
C
      WRITE (6,9872) (UNKWN(INDEX),INDEX=1,IJ)
      IF(IT3.LT.500) GO TO 58
      IF(NPCH.NE.2) GO TO 58
      DO 57 J=1,NA
      M=(J-1)*NR
      DO 57 I=1,NR
      K=M+I
   57 UNKWN(K)=UNKWN(K)*BI(I)
      WRITE(3,901) (UNKWN(I),I=1,IJ)
  901 FORMAT(8F10.6)
   58 CONTINUE
 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 BO2DS=ON2DS*BI(I)
      TPIB2=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)+SCTRMA(I,J)
      SQTUV=SQRT(U*U+V*V)
      SQ(I,JC)=SQTUV
      ARG=0.
      IF (SQTUV.EQ.0.) GO TO 605
      ARG=ATAN2(V,U)
      APHIJ=BET(I)+THET(I,JC)+ARG
  605 AP(I,JC)=APHIJ
      IF (IT3.EQ.1000) GO TO 700
C
C     RECOMPUTE COEFFICIENTS FROM SERIES SUBROUTINE
C
      IF (NAIR.EQ.1) CALL NACA15 (APHIJ,CLIFT,CDRAG,CMOME,ASLOP,EMT,U)
      IF (NAIR.EQ.1) GO TO 606
      CALL SERIES (I,J,2,EMT,U,V,APHIJ,CLIFT,ASLOP,CMOME,CDRAG)
C
  606 IF (VBINT(I,J).NE.0.) CALL INTRAT (I,J,CLIFT,CMOME,NA,DSI,APHIJ)
C
C     COMPUTE CAP I FUNCTIONS GIVEN ON PAGE III-8
C
      CO(I,JC)= 2.*802DS*CMOME*SQTUV
C
C     COMPUTE BLADE LOADS ACCORDING TO PAGE III-7
C
      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
      IF (IT3.EQ.1000) GO TO 825
C
C     CORRECT BLADE LOADS FOR J+1 AND J-1 TERMS
C
      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))
  800 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))
  825 CONTINUE
C
C
      IF (IT3.EQ.1000) WRITE (6,9878) ((SQ(I,JC),JC=1,NA),I=1,NR)
      IF (IT3.EQ.1000) WRITE (6,9877) ((AP(I,JC),JC=1,NA),I=1,NR)
  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
C
 9872 FORMAT (7H GAMMAS//(10G12.4))
 9877 FORMAT  ( 6HOAPHIJ/(18F7.3))
 9878 FORMAT  ( 6HOSQTUV/(18F7.3))
      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(360),CO(10,36),CO1(10,36),SIGMA(360)
      DIMENSION UNKWN(360),SV3(360),RBL(10),FORCX(10,36),FORCZ(10,36)
      DIMENSION EMOME(10,36),HDOT(10,36),PHI(10,36),THET(10,36),BI(10)
      DIMENSION BET(10),SAVE(360),WBR(10,36),EL(10,36)
C
      COMMON /CIR/PI,TWOPI,DIS
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /A1/ NBL,NB,NR,NW,NCV,MR2,MAXMU,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
C
      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)
850   WRITE(NOUT,981)(I,FORCZ(I,J),FORCX(I,J),EMOME(I,J),I=2,NP)
    9 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
   87 FORCX(M,N)=FORCX(M,N)*FACTR
C
      IF(IT3-1000) 895,891,891
  891 WRITE(NOUT,983)
      NAOVTO=NA/2+1
      NROW=10
      NCOL=36
      WRITE(NOUT,993) ((FORCZ(I,J),I=1,NR),J=1,NA)
      CALL HARMN(NR,NA,DSI,FORCZ,6,NMAS,NROW,NCOL)
      WRITE(NOUT,984)
      WRITE(NOUT,993) ((FORCX(I,J),I=1,NR),J=1,NA)
```

```
      CALL HARMN(NR,NA,DSI,FORCX,6,NMAS,NROW,NCOL)
      WRITE(NOUT,985)
      WRITE(NOUT,993) ((EMOME(I,J),I=1,NR),J=1,NA)
 993  FORMAT(8G15.5)
      CALL HARMN(NR,NA,DSI,EMOME,6,NMAS,NROW,NCOL)
  895 RETURN
8961  FORMAT(28X,2I5,3E20.7)
981   FORMAT(28X,I5,5X,3E20.7)
 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    )
      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
      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)/C2
      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 INTRAT (I,J,CLIFT,CMOME,NA,DSI,APHIJ)
      DIMENSION NACT(10),NBCT(10),NBSV(10),JCYC(10)
      DIMENSION   VBINT(10,36)
      COMMON /VORINT/ NACT,NBCT,NBSV,CLMAX,DELCL2,WOOM,CMST2,JCYCLE,JCYC
     1,APMAX
      COMMON /VORTEX/ VBINT
      IF (NACT(I).NE.0) GO TO 3
      IF (APHIJ.LT.APMAX) RETURN
      JCYC(I)=0
      JC=JCYCLE+J
      DO 1 K=J,JC
      IF (K.GT.NA) GO TO 2
      JCYC(I)=JCYC(I)+1
1      VBINT(I,K)=VBINT(I,K)+2.
    2 NACT(I)=1
      NBSV(I)=J
      NBCT(I)=0
      A=(CLIFT-CLMAX)/DELCL2+1.
      IF(ABS(A).GT.1.0)  A=A/ABS(A)
      PSI=1./WOOM*ASIN(A)
    3 NBCT(I)=NBCT(I)+1
      IF (NBCT(1).GT.JCYC(I)) GO TO 6
      PSIJVI=J*DSI
      PSIPSI=PSIJVI-PSI
      CLIFT=CLMAX+DELCL2*(SIN(WOOM*PSIPSI)-1.)
      CMOME=CMST2*(1.-COS(WOOM*PSIPSI))
      CMOME=(CMOME+CLIFT*.5)*.5
      GO TO 5
    6 JCY=JCYC(I)+NBSV(I)-1
      NBV=NBSV(I)
      DO 4 K=NBV,JCY
4      VBINT(I,K)=VBINT(I,K)-2.
      NACT(I)=0
    5 RETURN
      END
```

```
      SUBROUTINE NACA15 (ALPHA,CL,CD,CM,CLA,EMT,U)
      REAL MACH
      DATA PI,TWOPI/3.141593,6.283185/
      NS=1
      MACH=EMT*ABS(U)
    1 IF (ALPHA.GE.0.) GO TO 2
      ALPHA=-ALPHA
      NS=-1*NS
    2 IF (ALPHA.LE.PI) GO TO 3
      ALPHA=ALPHA-TWOPI
      GO TO 1
    3 ALSQ=ALPHA*ALPHA
      IF (ALPHA.GT..17453) GO TO 4
      CL=5.4425665*ALPHA
      CD=.006+1.2578279*ALSQ
      CM=1.44674322*ALPHA
      CLA=5.4425665
      GO TO 10
    4 IF (ALPHA.LE.3.05433) GO TO 5
      CL=-27.057746+8.6127481*ALPHA
      CD=1.1263058-.1100658*ALSQ
      CM=-13.846155+4.4073682*ALPHA
      CLA=8.6127481
      GO TO 10
    5 ALCUBE=ALSQ*ALPHA
      ALFOUR=ALCUBE*ALPHA
      IF (ALPHA.GT..59305) GO TO 6
      CL=-5.17552006+74.39225790*ALPHA-305.08426727*ALSQ+
     1 510.05783752*ALCUBE-300.69898135*ALFOUR
      CD=1.00809827-13.21615930*ALPHA+60.24338928*ALSQ-
     1 105.33915863*ALCUBE+66.65706750*ALFOUR
      CM=-1.3120201+19.4541356*ALPHA-82.0386380*ALSQ+
     1 139.6846800*ALCUBE-83.7789094*ALFOUR
      CLA=74.39225790-610.1685344*ALPHA+1530.1735125*ALSQ-
     1 1202.7959252*ALCUBE
      GO TO 10
    6 IF (ALPHA.GT.2.79253) GO TO 7
      CL=-1.1042825+5.9103327*ALPHA-5.2223610*ALSQ+
     1 1.3844760*ALCUBE-0.08952477*ALFOUR
      CD=-0.36898575+1.2001668*ALPHA+1.4199886*ALSQ-
     1 1.0988522*ALCUBE+.16709722*ALFOUR
      CM=-.1810135+1.13690885*ALPHA-1.1730461*ALSQ+
     1 .30186757*ALCUBE-.01199924*ALFOUR
      CLA=5.9103327-10.444722*ALPHA+4.153428*ALSQ-
     1 .35809908*ALCUBE
      GO TO 10
    7 CL=19073.5635-26905.6199*ALPHA+14223.9149*ALSQ-
     1 3340.04534*ALCUBE+293.9323*ALFOUR
      CD=4324.4195-6164.22236*ALPHA+3288.5370*ALSQ-
     1 778.11679*ALCUBE+68.89728*ALFOUR
```

```
      CM=20060.7293-27914.6308*ALPHA+14560.552*ALSQ-
     1 3374.20824*ALCUBE+293.10403*ALFOUR
      CLA=-26905.6199+28447.8298*ALPHA-10020.13602*ALSQ+
     1 1175.7292*ALCUBE
   10 EFMACH=SQRT(1.-MACH*MACH)
      CL=CL/EFMACH
      CD=CD/EFMACH
      CM=CM/EFMACH
      CLA=CLA/EFMACH
      IF (NS.EQ.1) GO TO 11
      CL=-CL
      CM=-CM
      ALPHA=-ALPHA
   11 RETURN
      END
```

```
      SUBROUTINE BLD4
C
      DIMENSION DALPIA(36),CSIX(20,37),              SIGKJ(20,20)
      DIMENSION ELNTH(18),EMAS(18),EIX(18),EIZ(18),RBL(10),
     1 XINR(18),DPHI(18),EIY(18),EPS(18),DLZ(18),ZA(18),
     2 YINR(18),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),
     2 AMY(18,20),AVZ(18,20),HDOT(10,36),PHI(10,36),
     3 THET(10,36),FX(10,36),FZ(10,36),EMO(10,36)
      DIMENSION VDOT(18,37),WDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION FV(18,37),FW(18,37),EMOME(18,37),CSI(20,37),CS2DT(20,37)
     1 ,CH(12,18),CCP(37),SSP(37),CCPS(18),SCPS(18),SUMA(18),H(18),X(19)
     2 ,RADIS(18),PHIV(18,37),SI(18,37),XCSIDT(1,36),THETA(18,37)
      DIMENSION CSPH(18),CAPHI(18),CC(37),SC(37),EX(37),EXMT(37),SN(11),
     1 SMLAZ(18),SMLAX(18),SMLAM(18),ALAM(37),SFZ(37),SFX(37),SFM(37),
     2 FORC(37)
      DIMENSION JM(10),JMP1(10),HDT(10),APH(10),ATH(10)
C
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/ ELNTH, EMAS, EIX , RBL , XINR, DPHI, EIY , EPS,
     1 DLZ, ZA , YINR, OMEGA, PSIR, SIG , AV , AW , APHI , ASI , ATHET,
     2AT , AMZ , AVY , AMY , AVZ , BD , RWK , CSAL , NM , XCSIDT,
     3SIGKJ, DAMPC
      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 /MANV2/ DALPIA,FGF
      COMMON /CIR/PI,TWOPI,DIS
C
C     BLADE DYNAMICS PROGRAM
C
C     SAVE NR  AND SET NEW NR FOR BLADE RESPONSE SUBPROGRAM
C
```

```
C       DEFINE CONSTANTS
C
        NPAGE=1
        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
        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)
        CMSQ=BD(13)
C
C       INITIALIZE DYNAMIC EFFECTS TO ZERO IF IN FIRST OVERALL ITERATION
C
        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
        ENA=NA
        ENR=NR
        CNOCP=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
C     COMPUTE TABLES OF COSINES AND OFFSET DISTANCES EXACTLY THE SAME AS
C     IN THE NATURAL FREQUENCY PROGRAM
C
      DO 40 I=2,NM
      IM1=I-1
C
C     CUMULATIVE PHI ANGLE ALONG BLADE
C
      CAPHI(I)=CAPHI(IM1)+DPHI(I)
      CCPS(I)=COS(CAPHI(I))
      SCPS(I)=SIN(CAPHI(I))
      CSPH(I)=COS(CAPHI(I)-THETO)
C
C     TOTAL DISTANCE TO RIGHT SIDE OF SECTION FROM CENTER OF ROTATION
C
      RADIS(I)=XROOT+ELNTH(I)
C
C     AVERAGE MASS USED IN COMPUTATION
C
   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)
   35 RADIS(I)=RADIS(I)+ELNTH(J)
   40 CONTINUE
C
C     SINE AND COSINE OF OMEGA * T
C
      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(12),FXX(12),FZZ(12),EMOO(12)
C
      DIMENSION                                          SIGKJ(20,20)
      DIMENSION ELNTH(18),EMAS(18),EIX(18),EIZ(18),RBL(10),
     1 XINR(18),DPHI(18),EIY(18),EPS(18),DLZ(18),ZA(18),
     2 YINR(18),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),
     2 AMY(18,20),AVZ(18,20),HDOT(10,36),PHI(10,36),
     3 THET(10,36),FX(10,36),FZ(10,36),EMO(10,36)
      DIMENSION VDOT(18,37),WDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION FV(18,37),FW(18,37),EMOME(18,37),CSI(20,37),CS2DT(20,37)
     1 ,CH(12,18),CCP(37),SSP(37),CCPS(18),SCPS(18),SUMA(18),H(18),X(19)
     2 ,RADIS(18),PHIV(18,37),SI(18,37),XCSIDT(1,36),THETA(18,37)
      DIMENSION CSPH(18),CAPHI(18),CC(37),SC(37),EX(37),EXMT(37),SN(11),
     1 SMLAZ(18),SMLAX(18),SMLAM(18),ALAM(37),SFZ(37),SFX(37),SFM(37),
     2 FORC(37)
      DIMENSION JM(10),JMP1(10),HDT(10),APH(10),ATH(10)
C
C
      COMMON /BLD4X1/ ELNTH, EMAS, EIX , RBL , XINR, DPHI, EIY , EPS,
     1 DLZ, ZA , YINR, OMEGA, PSIR, SIG , AV , AW , APHI , ASI , ATHET,
     2AT , AMZ , AVY , AMY , AVZ , BD , RWK , CSAL , NM , XCSIDT,
     3SIGKJ, DAMPC
C
      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 /CONL1/ ENR,PID2,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)
```

```
  150 EMOME(I,JA)=SFM(I)+ZA(I)*FV(I,JA)
  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)
  210 SIDT(I,NAP1)=SIDT(I,1)
      IF(IT3.EQ.1000) GO TO 100
      IF (NPRNT.LT.2) GO TO 1117
  100 WRITE (6,211)
      WRITE(6,9875)       ((   FV(I,JX),JX=1,NAP1),I=1,NM)
      WRITE (6,212)
      WRITE(6,9875)       ((   FW(I,JX),JX=1,NAP1),I=1,NM)
      WRITE (6,213)
      WRITE(6,9875)       ((EMOME(I,JX),JX=1,NAP1),I=1,NM)
      WRITE (6,214)
      WRITE(6,9875)       (( SIDT(I,JX),JX=1,NAP1),I=1,NM)
C
C     END CONVERSION OF AERODYNAMIC LOADS
C
  211 FORMAT (3HOFV//)
  212 FORMAT (3HOFW//)
  213 FORMAT (6HOEMOME//)
  214 FORMAT (5HOSIDT//)
 9875 FORMAT (       (1H ,10G12.4))
 1117 CONTINUE
      RETURN
      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 CSITMP(37),CSITXP(37)
      DIMENSION ASSF(20),ASSL(20),ASST(20)
      DIMENSION DALPIA(36),                                SIGKJ(20,20)
      DIMENSION ELNTH(18),EMAS(18),EIX(18),EIZ(18),RBL(10),
     1 XINR(18),DPHI(18),EIY(18),EPS(18),DLZ(18),ZA(18),
     2 YINR(18),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),
     2 AMY(18,20),AVZ(18,20),HDOT(10,36),PHI(10,36),
     3 THET(10,36),FX(10,36),FZ(10,36),EMO(10,36)
      DIMENSION VDOT(18,37),WDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION FV(18,37),FW(18,37),EMOME(18,37),CSI(20,37),CS2DT(20,37)
     1 ,CH(12,18),CCP(37),SSP(37),CCPS(18),SCPS(18),SUMAS(18)
     2 ,RADIS(18),PHIV(18,37),SI(18,37),XCSIDT(1,36),THETA(18,37)
      DIMENSION H(18),X(19)
      DIMENSION CSPH(18),CAPHI(18),CC(37),SC(37),EX(37),EXMT(37),SN(11),
     1 SMLAZ(18),SMLAX(18),SMLAM(18),ALAM(37),SFZ(37),SFX(37),SFM(37),
     2 FORC(37)
      DIMENSION JM(10),JMP1(10),HDT(10),APH(10),ATH(10)
      DIMENSION CH13(18)
C
C
      COMMON /BLD4X1/ ELNTH, EMAS, EIX , RBL , XINR, DPHI, EIY , EPS,
     1 DLZ, ZA , YINR, OMEGA, PSIR, SIG , AV , AW , APHI , ASI , ATHET,
     2AT , AMZ , AVY , AMY , AVZ , BD , RWK , CSAL , NM , XCSIDT,
     3SIGKJ, DAMPC
      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
      COMMON /MANV2/ DALPIA,FGF
      COMMON /MANV4/ ASSF,ASSL,ASST
      COMMON /MANV5/ ALFDTM, PHIDTM
C
```

```
C       CERTAIN COEFFICIENTS FO COMPUTATION OF TOTAL GENERALIZED FORCES
C       AND MOMENTS
C
        JKK= NA/2
        SNTHO=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))
        CH13(I)=(XINR(I)+SUMAS(I)*RADIS(I)*RADIS(I))*CPOMG
  300   CH(11,I)=CH(12,I)*SCPS(I)-2.*CPOMG*SUMAS(I)*EPS(I)*H(I)*SNTHO
     1           /CCPS(I)
C
        IF(NPRNT.LT.2) GO TO 9
        WRITE (6,301)
        WRITE (6,9875) ((CH(IX,I),IX=1,11),I=1,NM)
C
C       COMPUTE SUPERPOSITION INTEGRALS IN EACH KTH MODE
C
    9   DO 550 K=1,MAXMO
        SIGOM=SIG(K)*OMEGA(K)
        OMK2=OMEGA(K)*OMEGA(K)
        TEST=SIG(K)*SIG(K)
        IF (TEST.LE.1.) GO TO 5
        OMBR=OMEGA(K)*SQRT(TEST-1.)
        OMT=SIGOM-OMBR
        OMT2=SIGOM+OMBR
        DO 4 JA=1,NAP1
        CSITMP(JA)=CSI(K,JA)
        CSITXP(JA)=CSIDT(K,JA)
        T=(JA-1.)*DT
        FORC(JA)=0.0
        EXMT(JA)=EXP(OMT*T)
    4   EX(JA)=EXP(OMT2*T)
        GO TO 311
    5   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
   10   SIKDT=SIN(OMT)
```

```
      CSKDT=COS(DMT)
    6 DMT2=DMT*DMT
      DMT3=DMT2*DMT
      IF(DMT-.05) 303,307,307
  303 DMT4=DMT2*DMT2
      DMT6=DMT4*DMT2
C
C     ALPHA,BETA,GAMMA COEFFICIENTS FOR FILONS RULE OF INTEGRAION .
C     SEE APPENDIX OF BOOK ON INTEGRAL TRANSFORMS BY TRANTER
C
      FILA=DMT3/22.5-(DMT4*DMT)/157.5+(DMT6*DMT)/2362.5
      FILB=.6666666666666667+DMT2/7.5-DMT4/26.25+DMT6/283.5
      FILG=1.3333333333333-DMT2/7.5+DMT4/210.-DMT6/11340.
      GO TO 308
  307 FILA=(DMT2+DMT*SIKDT*CSKDT-2.*SIKDT*SIKDT)/DMT3
      FILB=2.*(DMT*(1.+CSKDT*CSKDT)-2.*SIKDT*CSKDT)/DMT3
      FILG=4.*(SIKDT-DMT*CSKDT)/DMT3
  308 ADT=FILA*DT
      BDT=FILB*DT
      GDT=FILG*DT
      DO 310 JA=1,NAP1
      T=(JA-1.)*DT
      FORC(JA)=0.
      CMT=CMBR*T
   20 CC(JA)=COS(CMT)
      SC(JA)=SIN(CMT)
  310 EXMT(JA)=EXP(-SIGOM*T)
  311 DO 360 JA=1,NA
      THC1=CPOMG*(AC*CCP(JA)-BC*SSP(JA))
      THC2=-CPSQ*(AC*SSP(JA)+BC*CCP(JA))
C
C     COMPUTE TOTAL GENERALIZED FORCES, PAGE IV-3,IV-4
C
      DO 350 I=1,NM
      QVNEW=(-CCPS(I)*CSAL+SCPS(I)*DALPIA(JA))*SUMAS(I)*FGF
      QWNEW=(SCPS(I)*CSAL+CCPS(I)*DALPIA(JA))*SUMAS(I)*FGF
      QPHI1=-EPS(I)*QVNEW
      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)
      QPHI =QPHI - CPOMG*(CCP(JA)*ALFDTM + SSP(JA)*PHIDTM)*CH(8,I)*CPOMG
      QSI=CH(10,I)+CH(11,I)*THC1+CH(12,I)*PHIDT(I,JA)-CH(3,I)*
     1VDOT(I,JA)-CH(6,I)*WDOT(I,JA)
      QTHET=-(-SSP(JA)*ALFDTM + CCP(JA)*PHIDTM)*CH13(I)*CPOMG
      QV=QV+QVNEW
      QW=QW+QWNEW
      QPHI=QPHI+QPHI1
      IF(JA-JKK) 312,312,314
  312 JB=JA+JKK
      GO TO 316
```

```
  314 JB=JA-JKK
  316 THO1=CPOMG*(AC*CCP(JB)-BC*SSP(JB))
      THO2=-CPSQ*(AC*SSP(JB)+BC*CCP(JB))
      QVONEW=(-CCPS(I)*CSAL+SCPS(I)*DALPIA(JB))*SUMAS(I)*FGF
      QWONEW=(SCPS(I)*CSAL+CCPS(I)*DALPIA(JB))*SUMAS(I)*FGF
      QPHIO1=-EPS(I)*QVONEW
      QVO=CH(1,I)+CH(2,I)*THO2+CH(3,I)*SIDT(I,JB)+FV(I,JB)
      QWO=CH(4,I)+CH(5,I)*THO2+CH(6,I)*SIDT(I,JB)+FW(I,JB)
      QPHIO=CH(7,I)+CH(8,I)*THO2+CH(9,I)*SIDT(I,JB)+EMOME(I,JB)
      QPHIO=QPHIO- CPOMG*(CCP(JB)*ALFDTM + SSP(JB)*PHIDTM)*CH(8,I)*CPOMG
      QSIO=CH(10,I)+CH(11,I)*THO1+CH(12,I)*PHIDT(I,JB)-CH(3,I)*
     1VDOT(I,JB)-CH(6,I)*WDOT(I,JB)
      QTHETO=-(-SSP(JB)*ALFDTM + CCP(JB) * PHIDTM)*CH13(I)*CPOMG
      QVO=QVO+QVONEW
      QWO=QWO+QWONEW
      QPHIO=QPHIO+QPHIO1
  318 IF(I-2) 330,320,330
  320 G1=AKL/ELNTH(2)*SIDT(1,JA)
      G12=AKL/ELNTH(2)*SIDT(1,JB)
      QV=QV+SCPS(2)*G1
      QW=QW+CCPS(2)*G1
      QVO=QV+SCPS(2)*G12
      QWO=QW+CCPS(2)*G12
C
C       GENERALIZED FORCE ACTING IN EACH NORMALIZED MODE
C
  330 FORCO=QVO*AV(I,K)*ASSF(K)+(QWO*AW(I,K)+QSIO*ASI(I,K))*
     1ASSL(K)+QPHIO*APHI(I,K)*ASST(K)
      FORCO = FORCO + QTHETO*ATHET(I,K)*ASST(K)
      FORC(JA)=FORC(JA)+QV*AV(I,K)+QW*AW(I,K)+QPHI*APHI(I,K)
     1+QSI*ASI(I,K)+FORCO
      FORC(JA)=FORC(JA) + QTHET*ATHET(I,K)
  350 CONTINUE
      DO 355 KJ=1,MAXMO
  355 FORC(JA)=FORC(JA)-SIGKJ(K,KJ)*(CSIDT(KJ,JA)+CSIDT(KJ,JB)*ASST(K))
      FORC(JA)=FORC(JA)*(1.-.5*ASST(K))
  360 CONTINUE
      FORC(NAP1)=FORC(1)
      WRITE(6,361)
      WRITE(6,9875)          (FORC(JA),JA=1,NA)
      IF (TEST.GT.1.) GO TO 371
      DO 370 JA=1,NAP1
  370 EX(JA)=(1./EXMT(JA))*FORC(JA)
      JGO=1
      GO TO 379
C
C     COMPUTE SUPERPOSITION INTEGRALS AT EACH AZIMUTH POSITION
C     INTEGRALS ON PAGE IV-6
C
  371 SC(1)=0.0
```

```
      CC(1)=0.0
C
C        COMPUTATION OF REDUCED INTEGRAL PARTS OF SUPPOSITION
C       INTEGRALS AT EACH AZIMUTH POSITION FOR OVERDAMPED CASE
C
      DO 372 JA=2,NAP1
      JAA=JA-1
      CC(JA)=CC(JAA)+DT02*(FORC(JA)*EXMT(JA)+FORC(JAA)*EXMT(JAA))
  372 SC(JA)=SC(JAA)+DT02*(FORC(JA)*EX(JA)+FORC(JAA)*EX(JAA))
      OMT3=1.0/EXMT(NAP1)
      OMT4=1.0/EX(NAP1)
      CK=1.0-OMT4
      SK=1.0-OMT3
      S1=(OMT3*CC(NAP1))/SK
      S2=(OMT4*SC(NAP1))/CK
C        CALCULATION OF CSI(K,JA),CSIDT(K,JA),CS2DT(K,JA)
      DO 375 JA=1,NAP1
      CMT3=.5*(CC(JA)+S1)/EXMT(JA)
      OMT4=.5*(SC(JA)+S2)/EX(JA)
      CSI(K,JA)=(OMT3-OMT4)/OMBR
      CSIDT(K,JA)=-SIGOM*CSI(K,JA)+OMT3+OMT4
      CS2DT(K,JA)=FORC(JA)-OMK2*CSI(K,JA)-2.*SIGOM*CSIDT(K,JA)
      CSI(K,JA)=(CSI(K,JA)*FINPT+CSITMP(JA)*(1.-FINPT))
  375 CSIDT(K,JA)=(CSIDT(K,JA)*FINPT+CSITXP(JA)*(1.-FINPT))
      GO TO 550
  379 DO 500 JAA=1,NAP1
      CSITMP(JAA)=CSI(K,JAA)
      CSITXP(JAA)=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=DT02*EX(2)
      CSINT=SAVE*CC(2)+DT02*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)
   40 S=CK*CK+SK*SK
      S1=(SK*CKINT+CK*SKINT)/S
      S2=(CK*CKINT-SK*SKINT)/S
  457 CONTINUE
  500 CONTINUE
C
C     COMPUTE QUANTITIES ZETA (CSI) AND ZETA DOT (CSI DOT) FROM
C     SUPERPOSITION INTEGRALS
C     COMPUTATION OF ZETA,ZETA-DOT, SEE PAGE IV-6
C
      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
      DO 540 JAA=1,NAP1
      CSI(K,JAA)=(CSI(K,JAA)*FINPT+CSITMP(JAA)*(1.-FINPT))
  540 CSIDT(K,JAA)=(CSIDT(K,JAA)*FINPT+CSITXP(JAA)*(1.-FINPT))
C
  550 CONTINUE
      IF(NPRNT.LT.2) RETURN
C
      WRITE (6,551)
      WRITE(6,9875)        ((  CSI(K,JAA),K=1,MAXMO),JAA=1,NAP1)
      WRITE (6,552)
      WRITE(6,9875)        ((CSIDT(K,JAA),K=1,MAXMO),JAA=1,NAP1)
      WRITE (6,553)
      WRITE(6,9875)        ((CS2DT(K,JAA),K=1,MAXMO),JAA=1,NAP1)
C
```

```
      RETURN
 9875 FORMAT  (        (1H ,09G13.5))
  301 FORMAT (3HOCH//)
 6001 FORMAT(1HO,4HK = ,I3,5X,8HSIG =    ,G10.3,2X,8HOMEGA = ,G10.3,2X,
     1                          8HDT =     ,G10.3)
  361 FORMAT (5HOFORC//)
  551 FORMAT (4HOCSI//)
  552 FORMAT (6HOCSIDT//)
  553 FORMAT (6HOCS2DT//)
      END
```

```
      SUBROUTINE RSPNS
C
      DIMENSION VDTDT(18,37),VX(18,37),WX(18,37)
      DIMENSION                                      SIGKJ(20,20)
      DIMENSION ELNTH(18),EMAS(18),EIX(18),EIZ(18),RBL(10),
     1 XINR(18),DPHI(18),EIY(18),EPS(18),DLZ(18),ZA(18),
     2 YINR(18),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(18,20),Aw(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),
     2 AMY(18,20),AVZ(18,20),HDOT(10,36),PHI(10,36),
     3 THET(10,36),FX(10,36),FZ(10,36),EMO(10,36)
      DIMENSION VDOT(18,37),WDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION FV(18,37),FW(18,37),EMOME(18,37),CSI(20,37),CS2DT(20,37)
     1 ,CH(12,18),CCP(37),SSP(37),CCPS(18),SCPS(18),SUMA(18),H(18),X(19)
     2 ,RADIS(18),PHIV(18,37),SI(18,37),XCSIDT(1,36),THETA(18,37)
      DIMENSION CSPH(18),CAPHI(18),CC(37),SC(37),EX(37),EXMT(37),SN(11),
     1 SMLAZ(18),SMLAX(18),SMLAM(18),ALAM(37),SFZ(37),SFX(37),SFM(37),
     2 FORC(37)
      DIMENSION JM(10),JMP1(10),HDT(10),APH(10),ATH(10)
C
C
      COMMON /BLD4X1/ ELNTH, EMAS, EIX , RBL , XINR, DPHI, EIY , EPS,
     1 DLZ, ZA , YINR, OMEGA, PSIR, SIG , AV , AW , APHI , ASI , ATHET,
     2AT , AMZ , AVY , AMY , AVZ , BD , RWK , CSAL , NM , XCSIDT,
     3SIGKJ, DAMPC
      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,NAPSGN,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
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      COMMON/PRNT/NPRNT
C
C     COMPUTE RESPONSES FROM MODE SHAPES AND ZETAS
C
      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.
      VDTDT(I,JA)=0.
  575 THETA(I,JA)=0.
C
C     RESPONSE VARIABLES OF INTEREST, PAGE IV-7
C
      DO 580 JA=1,NA1
      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
C
      IF(IT3.EQ.1000) GO TO 9
      IF(NPRNT.LT.2) GO TO 3
    9 NAP1=NA
      WRITE (6,9988) ((VX(I,JA),JA=1,NAP1),I=1,NM)
      IF(NPRNT.LT.1) GO TO 12
      WRITE (6,9989) ((WX(I,JA),JA=1,NAP1),I=1,NM)
      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)
  12   CONTINUE
      GO TO (3,4),NGOTO
C
C
C     CONVERT RESPONSE TO FORM FOR LOADS COMPUTATIONS
C
    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=UNOCP*(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
    4  CONTINUE
      IF(IT3.EQ.1000) GO TO 10
      IF(NPRNT.LT.2)GO TO 6
   10 WRITE(6,24)
      WRITE(6,29)((HDOT (I,J),J=1,NA),I=1,NR)
      WRITE(6,22)
      WRITE(6,29)((PHI  (I,J),J=1,NA),I=1,NR)
      WRITE(6,23)
```

```
      WRITE(6,29)((THETA(I,J),J=1,NA),I=1,NR)
      WRITE (6,28)
      WRITE (6,29) ((THET(I,J),J=1,NA),I=1,NR)
    6 WRITE(NOUT,2935)
      WRITE(NOUT,939)(II,II=1,MAXMO)
  939 FORMAT(1X,14HAZIMUTH (DOWN)/6H MODE ,2X,I1,    13(7X,I2))
      DO 5 JJ=1,NA
    5 WRITE(NOUT,938) JJ,(CSI(II,JJ),II=1,MAXMO)
      IF(IT3-1000)890,750,750
  750 CONTINUE
C
      NAOVTO=18
      NRNRA=37
      WRITE(6,8995)
      CALL HARMN(NM,NAP1,DSI,PHIV ,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)
      NAP1=NA+1
  890 NR=SNR
      RETURN
 9988 FORMAT (/1HO,3HVX //(9G13.5))
 9989 FORMAT (/1HO,3HWX //(9G13.5))
 9990 FORMAT(/1HO,5HVDOT //(9G13.5))
 9991 FORMAT(/1HO,5HWDOT //(9G13.5))
 9992 FORMAT(/1HO,5HPHIV //(9G13.5))
 9993 FORMAT(/1HO,5HPHIDT//(9G13.5))
 9994 FORMAT(/1HO,5HSI   //(9G13.5))
 9995 FORMAT(/1HO,5HSIDT //(9G13.5))
 9996 FORMAT(/1HO,5HTHETA//(9G13.5))
 9997 FORMAT (6HOVDTDT//(9G13.5))
   24 FORMAT (5HOHDOT//)
   29 FORMAT(10(1X,E12.5),/)
   22 FORMAT (4HOPHI//)
   23 FORMAT (6HOTHETA//)
   28 FORMAT (5HOTHET//)
 2935 FORMAT (2(/),55X,10HCSI VALUES)
  938 FORMAT(1X,I2,14(F9.4))
  900 FORMAT(1X,5E20.7)
 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)
      END
```

```
      SUBROUTINE SHEAR
C
      DIMENSION CT(18,37),CMZ(18,37),CMY(18,37),CVZ(18,37),CVY(18,37)
      DIMENSION                                          SIGKJ(20,20)
      DIMENSION ELNTH(18),EMAS(18),EIX(18),EIZ(18),RBL(10),
     1 XINR(18),DPHI(18),EIY(18),EPS(18),DLZ(18),ZA(18),
     2 YINR(18),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),
     2 AMY(18,20),AVZ(18,20),HDOT(10,36),PHI(10,36),
     3 THET(10,36),FX(10,36),FZ(10,36),EMO(10,36)
      DIMENSION VDOT(18,37),WDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION FV(18,37),FW(18,37),EMOME(18,37),CSI(20,37),CS2DT(20,37)
     1 ,CH(12,18),CCP(37),SSP(37),CCPS(18),SCPS(18),SUMA(18),H(18),X(19)
     2 ,RADIS(18),PHIV(18,37),SI(18,37),XCSIDT(1,36),THETA(18,37)
      DIMENSION CSPH(18),CAPHI(18),CC(37),SC(37),EX(37),EXMT(37),SN(11),
     1 SMLAZ(18),SMLAX(18),SMLAM(18),ALAM(37),SFZ(37),SFX(37),SFM(37),
     2 FORC(37)
      DIMENSION JM(10),JMP1(10),HDT(10),APH(10),ATH(10)
C
C
      COMMON /BLD4X1/ ELNTH, EMAS, EIX , RBL , XINR, DPHI, EIY , EPS,
     1 DLZ, ZA , YINR, OMEGA, PSIR, SIG , AV , AW , APHI , ASI , ATHET,
     2AT , AMZ , AVY , AMY , AVZ , BD , RWK , CSAL , NM , XCSIDT,
     3 SIGKJ, DAMPC
      COMMON /GCORD1/ THETO,CPSQ,DT,AC,BC,AKL,DTQ2,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,NRI1
      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 /A1/ NBL,NB,NR,NW,NCV,NR2,MAXMO,NIT1,NIT2,NIT3
     1 ,CPCMG,IT3
      COMMON /IO/IN,NOUT,IT7,IT8
      COMMON /SHEAR1/ NAOVTO
C
      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
  100 DVY=DVY+AVY(I,K)*CSIX
      CT(I,JA)=DT
      CMZ(I,JA)=DMZ
      CMY(I,JA)=DMY
      CVZ(I,JA)=DVZ
      CVY(I,JA)=DVY
  200 CONTINUE
C
      NSAVE=NA
      NAOVTO=18
      NRNRNA=37
      CALL HARMN (NM,NA,DSI,CT,1,NMAS,NAOVTO,NRNRNA)
      WRITE (6,900)
      WRITE (6,901) ((CT(I,J),J=1,NA),I=1,NM)
      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
  900 FORMAT (///,56X,19HRADIAL VS AZIMUTHAL)
  901  FORMAT(///(9G14.6))
      END
```

```
      SUBROUTINE RSPZZ
C
      DIMENSION RCAP(11),ZWK(11,36),VX(18,37),WX(18,37)
      DIMENSION KM(11),KMP1(11)
      DIMENSION                                          SIGKJ(20,20)
      DIMENSION ELNTH(18),EMAS(18),EIX(18),EIZ(18),RBL(10),
     1 XINR(18),DPHI(18),EIY(18),EPS(18),DLZ(18),ZA(18),
     2 YINR(18),OMEGA(20),SIG(20),BD(15)
      DIMENSION AV(18,20),AW(18,20),APHI(18,20),ASI(18,20),
     1 ATHET(18,20),AT(18,20),AMZ(18,20),AVY(18,20),
     2 AMY(18,20),AVZ(18,20),HDOT(10,36),PHI(10,36),
     3 THET(10,36),FX(10,36),FZ(10,36),EMO(10,36)
      DIMENSION VDOT(18,37),WDOT(18,37),PHIDT(18,37),SIDT(18,37),
     1 CSIDT(20,37)
      DIMENSION FV(18,37),FW(18,37),EMOME(18,37),CSI(20,37),CS2DT(20,37)
     1 ,CH(12,18),CCP(37),SSP(37),CCPS(18),SCPS(18),SUMA(18),H(18),X(19)
     2 ,RADIS(18),PHIV(18,37),SI(18,37),XCSIDT(1,36),THETA(18,37)
      DIMENSION CSPH(18),CAPHI(18),CC(37),SC(37),EX(37),EXMT(37),SN(11),
     1 SMLAZ(18),SMLAX(18),SMLAM(18),ALAM(37),SFZ(37),SFX(37),SFM(37),
     2 FORC(37)
      DIMENSION JM(10),JMP1(10),HDT(10),APH(10),ATH(10)
C
C
      COMMON /BLD4X1/ ELNTH, EMAS, EIX , RBL , XINR, DPHI, EIY , EPS,
     1 DLZ, ZA , YINR, OMEGA, PSIR, SIG , AV , AW , APHI , ASI , ATHET,
     2AT , AMZ , AVY , AMY , AVZ , BD , RWK , CSAL , NM , XCSIDT,
     3 SIGKJ, DAMPC
      COMMON /GCURD1/ THETO,CPSW,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,NAPSGN,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/ NGUTO
      COMMON /SAD2/ VDOT,WDOT,PHIDT,SIDT,CSIDT
      COMMON /SAD6/ VX,WX
C
C     COMPUTE Z S FOR CASE 12 WAKE AND LOADS RUNS
C
    4 READ(5,9990)(RCAP(I),I=1,NRP1)
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
C
      WRITE(6,24)
      WRITE(3,29)(( ZWK (I,J),I=1,NRP1),J=1,NA)
      WRITE(6,30)(( ZWK (I,J),I=1,NRP1),J=1,NA)
    3 CONTINUE
 9990 FORMAT (8F10.9)
   24 FORMAT(   5HOZWK ,//)
   29  FORMAT( 29X,F10.7)
   30 FORMAT (1X,8E16.7)
      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 CDC 6600 under Scope 3.0. 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 the two calls
   to CALL BLD4
```

Subroutine BLD3:

```
OVERLAY(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 two 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
```

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D.C. 20546

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300

FIRST CLASS MAIL

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION
451

U.S.MAIL

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Return

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons. Also includes conference proceedings with either limited or unlimited distribution.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include final reports of major projects, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546