NASA CR 167952

Turbofan Noise Generation
Volume 2: Computer Programs

by
C.S. Ventres
M.A. Theobald
W.D. Mark

(NASA-CR-167952) TURBOFAN NOISE GENERATION. VOLUME 2: COMPUTER PROGRAMS Final Report (Bolt, Beranek, and Newman, Inc.) 206 p HC A10/MF A01 CSCL 20A
N85-11791
Unclas
G3/71 24800

July 1982

Prepared Under Contract No. NAS3-21252
by
Bolt Beranek and Newman Inc.
Cambridge, Mass. 02138

for

NASA

National Aeronautics and
Space Administration
Lewis Research Center
Cleveland, Ohio 44135

Report No. 4770

TURBOFAN NOISE GENERATION:
VOLUME 2: COMPUTER PROGRAMS

C.S. Ventres
M.A. Theobald
W.D. Mark

April 1982

Prepared for:

National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Onio 44135

SUMMARY OF VOLUME 2

The computations for three different noise mechanisms - mean rotor wake, inlet turbulence, and rotor wake turbulence - described in Volume 1 of this report are coded as three separate computer program "packages." This arrangement was deemed reasonable because of differences in the required input data (e.g., rotor vs stator geometries), because of interest in the role of each separate noise mechanism, and for economy of execution during parametric studies. The computer codes are described by means of block diagrams, tables of data and variables, and example program executions. Reference to Volume 1 will be made via equation or figure number. FORTRAN listings of the programs appear in the appendix to Volume 2.

PRECEDING PAGE BLANK NOT FILMED

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

MEAN ROTOR WAKE PROGRAM

The computer program described in Fig. 1* computes the sound power of each propagating mode for the first three harmonics of the blade passage frequency. The data required to run the program and the program results are described in Table 1.* Important FORTRAN variables used in the program are given in Table 2. Function subroutines utilized in the program are described in Table 3.

The terminal output from a sample execution of the Mean Rotor Wake Program is shown in Table 4 for the rotor/stator data provided by NASA Lewis. The mean flow velocity entering the stator is specified from measured data. The pressure amplitude and sound power of each mode for each harmonic number of the blade passage frequency are listed. Note that for a propagating mode (m,n) at harmonic s, another mode (-m,n) of equal amplitude propagates with frequency -s. Thus the total sound power for the mode shape (|m|, n) at frequency |s|x (blade passage frequency) is the sum of the two components.

*Note: Because of the number and complexity of the figures and tables, they have been collected at the end of each section of this volume. The 9 text figures and 12 tables appear before the appendix.

CHAPTER 2

TURBULENCE PROGRAMS

The calculations for two types of turbulence noise are considerably more time-consuming than for the tones of the mean-wake interaction because 1) all modes that are not cut off by the duct may be excited, i.e., there is no selection rule, as is the case for the mean-wake interaction, and 2) the noise is broadband rather than tonal, requiring the sound power at many frequencies to be calculated. For these reasons, the turbulence programs are each broken into two subsections. The method used is, first, to calculate and store, for a given number of blade radii, the pressure distributions across the rotor blades or stator vanes at a small number of frequencies over the range of interest. The power spectral density is essentially a spatial integral containing the pressure distribution and acoustic mode shape as factors in the integrand. The pressure distributions are slowly varying functions of frequency and duct radius when compared with the acoustic mode shapes. Thus it is reasonable to interpolate the pressure distribution for a specific frequency and radius from a look-up table of stored pressure values. All other variables that change with radius (or frequency) are calculated explicitly for that given radius (frequency). In this manner, convergence of the radial integration is improved by using more integration steps than there are stored pressure values. The pressure distributions are stored in binary form in a disc file for later use so that, for example, turbulence parameter values may be varied without recomputing the pressures.

2.1 Inlet Turbulence Program

The pressure distributions across the rotor blades are computed and stored as described in Fig. 2. The sound power calculation is diagrammed in Fig. 3. Input and output data for the programs are listed in Table 5. Additional program variables of possible interest to the user are given in Table 6.

Output from sample executions of the two program sections is shown in Tables 7 and 8. The input data relating to rotor geometry and performance are first entered as listed in Table 7. The pressure distributions are computed (as evidenced by the error code returned from LEQTIC; see Table 3), and the data are stored on a disc. The power spectral density for all propagating modes at a frequency of 15 times shaft rate (blade passage frequency in this 15-blade fan) is then computed, and the results are given in Table 8.

2.2 Rotor Wake Turbulence Program

The two-stage method of calculation described above is repeated for the Rotor Wake Turbulence problem. The calculation of pressure distributions across the stator vanes proceeds as shown in Fig. 4. The sound power calculation is explained in Fig. 5. Input and output data for the programs are listed in Table 9. Additional program variables are given in Table 10.

The output from sample program executions is shown in Tables 11 and 12. The mean flow into the stator may be specified as for the Mean Wake case. (Note that the printout shown in Table 12 has been stripped of all messages other than the power spectral density per mode to minimize the amount of paper delivered for high frequencies (many modes). The standard program version would contain diagnostic messages, cut-off ratios, etc., as for Table 8.

CHAPTER 3

EXECUTION PARAMETERS AND PROGRAM LIMITATIONS

A. General Remarks

1. Specifying blade or vane geometries: Geometrical data are stored in matrix VGEOM. The number of radial positions at which data may be specified (NDAT) is presently limited to 10 by DIMENSION statements. This limitation may be lifted by increasing the size of VGEOM *in all main and subprograms.* The geometrical data are specified by proceeding radially *from hub to tip,* and *must include these two end points.* These data are linearly interpolated for integration stations between the data points.

2. Specifying mean flow distributions: Variable IVOR is a switch controlling whether a free vortex distribution (IVOR=0) or input data (IVOR=1) are used in stator calculations. Input data are stored in matrix VELOCV. The number of radial positions at which data are specified (NVELO) is presently limited to 10. This limitation may be lifted by increasing the size of VELOCV *in all main and subprograms.* The velocity data are specified by proceeding radially *from hub to tip* and *must include these two endpoints.* These data are also linearly interpolated for intermediate points in the radial integration.

3. Specifying Bessel function accuracies: The absolute accuracies for all Bessel function calculations (EB) and root convergence (EC) have been left as input parameters. Values of 0.001 and 0.0001, respectively, have been sufficient for these calculations.

4. Specifying the number of chordwise integration points: The number of chordwise integration points (NCHORD) is an input variable. DIMENSION statements currently limit the value of NCHORD to 20 in all programs. A value of 8 has yielded sufficient accuracy in the cases checked to date. Execution time increases radically for increases in NCHORD, so its value should be minimized. For cases where the acoustic wavelength is considerably less than the blade or stator chord, it may be necessary to increase NCHORD beyond 8. If NCHORD must be increased beyond 20, the following array sizes must be inspected and/or increased: CASCET, PHASE, DELTAP, F, A, WA, and B.

5. Specifying the number of radial integration points: The number of radial integration points (NRAD in Mean Wake, or NRADNU in the turbulence programs) is an input variable. The value must be large enough to achieve convergence of the result, yet small enough to minimize program execution time. A value of 7 has yielded adequate accuracy (±1 dB) for the acoustic power levels in cases checked to date. The value of 20 must not be exceeded without inspecting and/or increasing the sizes of arrays PSISTO, BJ, RR, RRNU, and CASCET.

B. Turbulence Programs

1. Specifying the number of radial positions for pressure calculations: Both turbulence programs compute and store pressure distributions at a number (NRAD) of radial positions. Acoustic calculations are later performed by using NRADNU (>NRAD) integration points. This arrangement minimizes the execution time needed for pressure calculations while maintaining good accuracy in the radial integral. NRAD values of 5 have been used for the test cases cited here. NRAD may not be increased beyond 20 without violating DIMENSION statements, as discussed in Sec. A.5.

2. Range of Mode Number m: The turbulence programs that compute power spectral densities contain a summation of autocorrelation functions that vary with mode number m. If the annular duct/turbofan arrangement were to allow propagation of modes with mode number |m| greater than 49 at a given frequency, then program execution would be terminated with an error message. This range may be increased by increasing the size of arrays ISUMTO and SUMTOT and by changing the decision value in the associated IF statement.

3. Selection of frequency values for data storage: The turbulence programs compute and store pressure distributions for a range of frequencies for later interpolation. The most useful frequencies to calculate in the subject spectrum of, for example, from 1 to 3 times the blade passage frequency (BPF), are probably the first, second, and third harmonics of BPF. This range is straightforwardly specified for the wake turbulence program. An important qualification to this procedure must be invoked for inflow turbulence, as explained below.

The sound power calculations for the inflow turbulence routine require pressure distribution data for frequencies up to an equivalent frequency

$$(\omega^*/\Omega) = (\omega/\Omega)_{max} + |m_{max}| + m_{sum},$$

where $(\omega/\Omega)_{max}$ is the highest frequency of interest, m_{max} is the largest value of m which can propagate at frequency (ω/Ω), and $m_{sum} = -|m_{max}| + (\ell B)$ for some integer ℓ such that $0 \le m_{sum} \le B-1$, for B=number of rotor blades. Thus, *before* computing the pressure distribution, the user must know what modes can propagate for the turbofan under discussion. The utility program MAPIN/MAPPER (see appendix) will yield the propagating modes for a given geometry, frequency, and turbofan speed. As an example, suppose that the turbulence noise spectrum between BPF and 3BPF is desired for NASA Rotor 55 with the operating conditions: Number of blades = 15, M_t = 0.508, M = 0.323, and σ_r = 0.484. Pressure data will be needed from the lowest frequency of interest $(\omega/\Omega$ = BPF = 15) to the highest equivalent frequency (ω^*/Ω). Program MAPIN/MAPPER reveals that $|m_{max}| = 21$ for (ω/Ω) = 3BPF = 45. Thus,

$$(\omega^*/\Omega) = (\omega/\Omega)_{max} + |m_{max}| + m_{sum}$$

$$(\omega^*/\Omega) = 45 + 21 + 9 = 75,$$

$$\text{where } m_{sum} = -21+(2\cdot 15) = 9.$$

In the example, (ω^*/Ω) conveniently falls at the fifth multiple of blade passage frequency (5BPF) or ω/Ω=75. The frequency range for pressure distribution calculations is then from (ω/Ω)=15 to (ω/Ω) = 75 in steps of (ω/Ω) = 15. The results of calculations for that case are shown in Table 7. If programs INSRCH/INTURB were to be executed without pressure data of sufficiently high equivalent frequency, the program execution would be terminated with an error message.

4. DIMENSION of storage matrix CASCET: Pressure distributions are stored on the disc in matrix CASCET. It is desirable to keep this matrix as small as possible to minimize core and disc file requirements. The DIMENSION of CASCET should therefore be adjusted as follows to suit the data: DIMENSION CASCET (NCHORD, NRAD, NBLADE, PRANGE), where NCHORD = number of chordwise integration points, NRAD = number of radial positions for pressure calculations, NBLADE = number of rotor blades, or, for the Rotor Wake Turbulence Program, use NVANE = number of stator vanes. PRANGE = number of frequencies for pressure calculations.

FIGURES

Calculate the sound power of each propagating mode for the first 3 harmonics of the blade passage frequency.

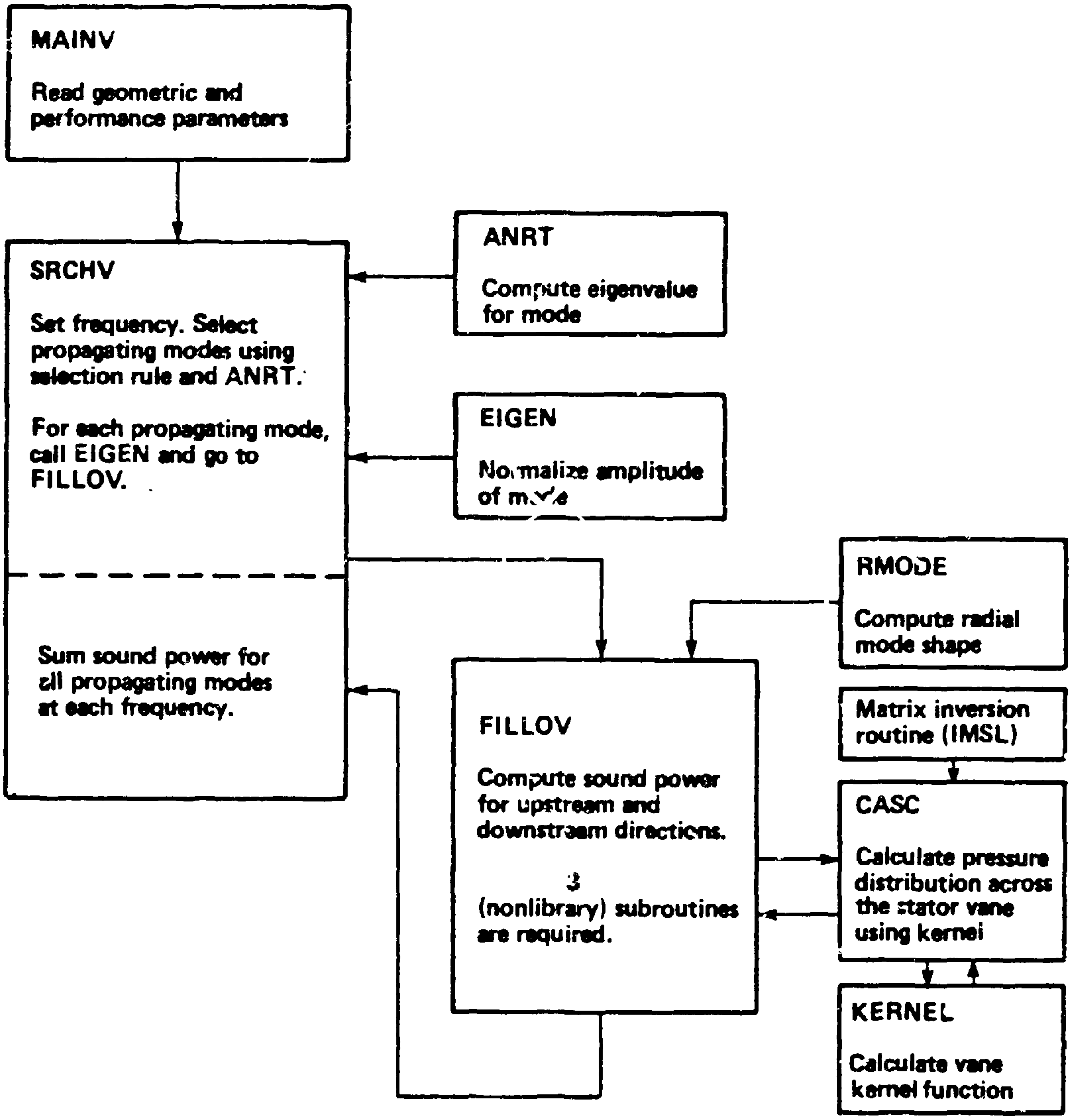

Figure 1. Block Diagram of Mean Rotor Wake Program

A. Calculate pressure distributions across the rotor blades and store data on the disk.

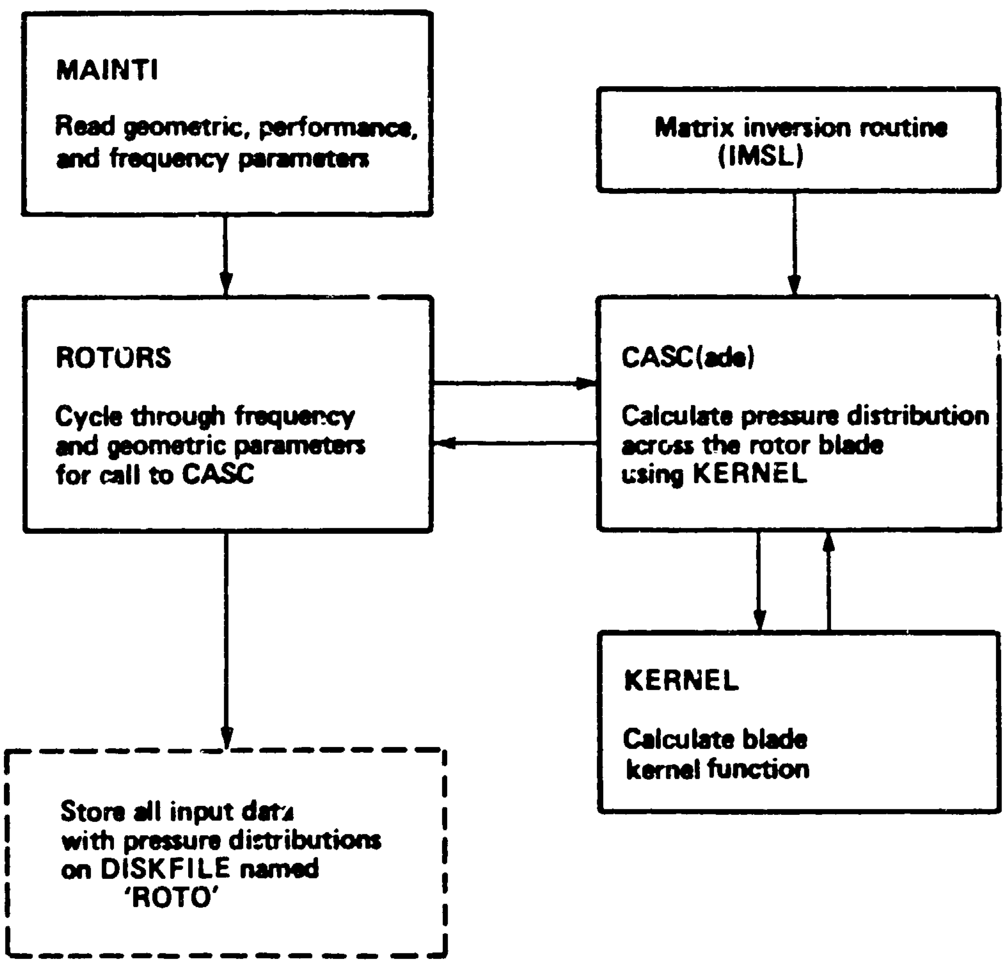

Figure 2. Block Diagram of Inlet Turbulence Program; Part A: Data Storage

B. Calculate the power spectral density of each propagating mode at a given frequency and the total PSD for all modes.

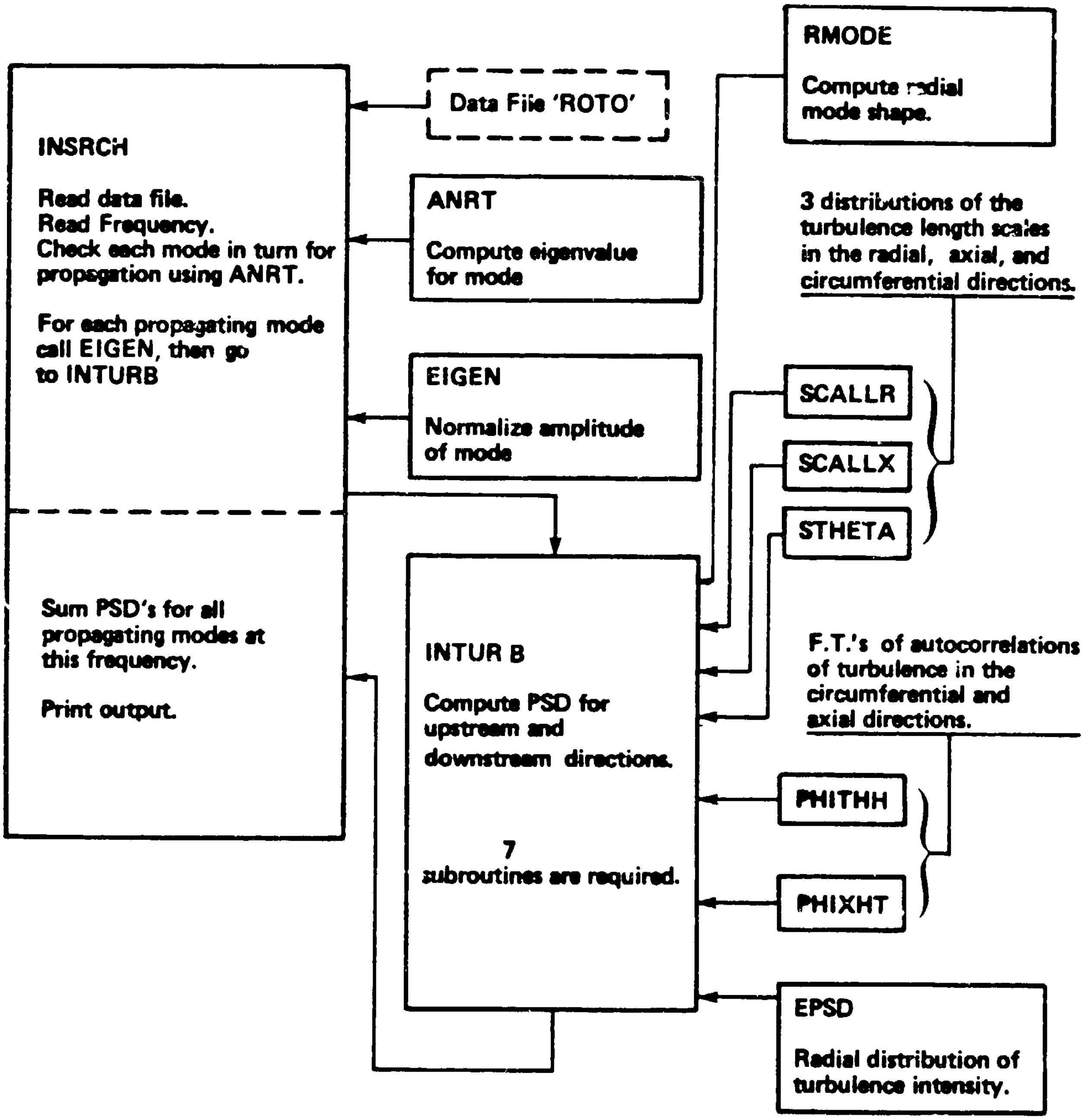

Figure 3. Block Diagram of Inlet Turbulence Program; Part B: Noise Calculation

A. Calculate pressure distributions across the stator vanes and store data on the disk.

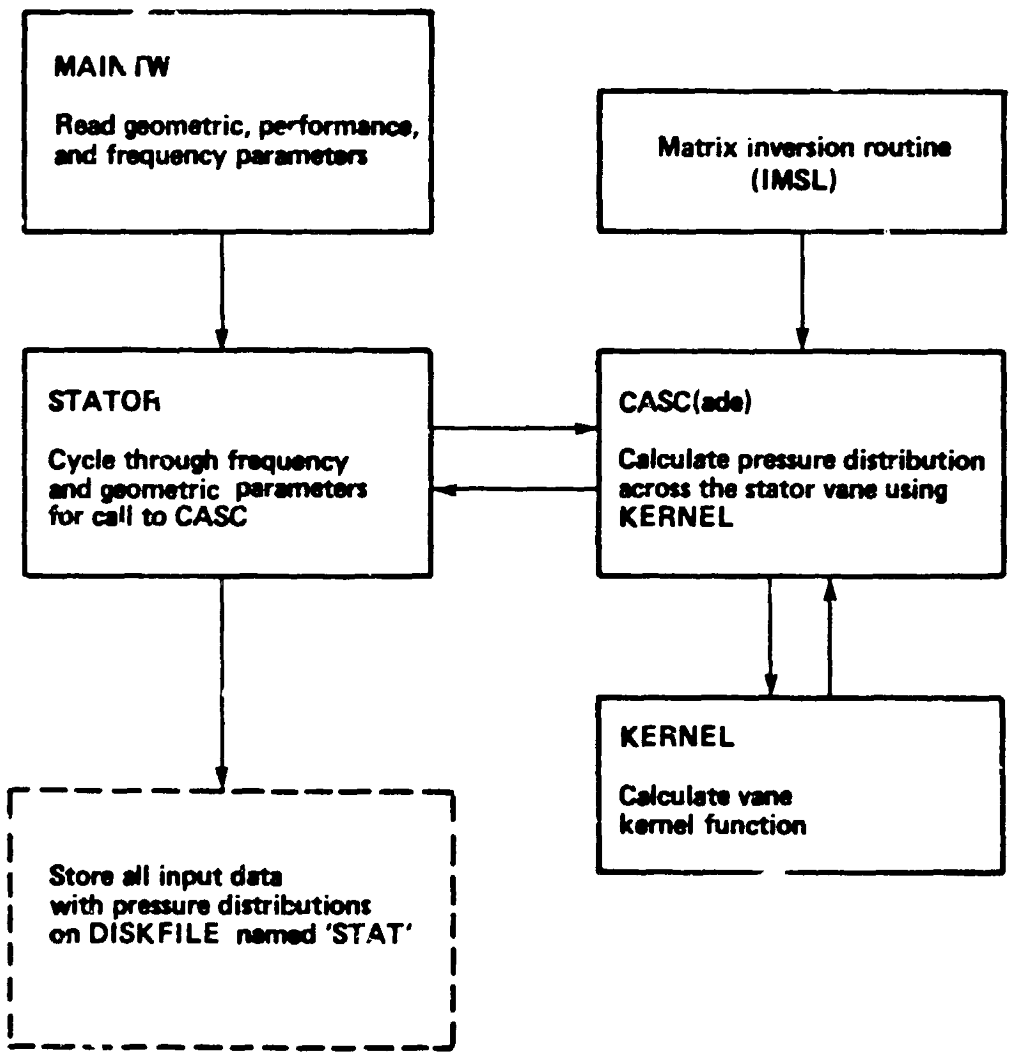

Figure 4. Block Diagram of Rotor Wake Turbulence Program; Part A: Data Storage

B. Calculate the power spectral density of each propagating mode at a given frequency and the total PSD for all modes.

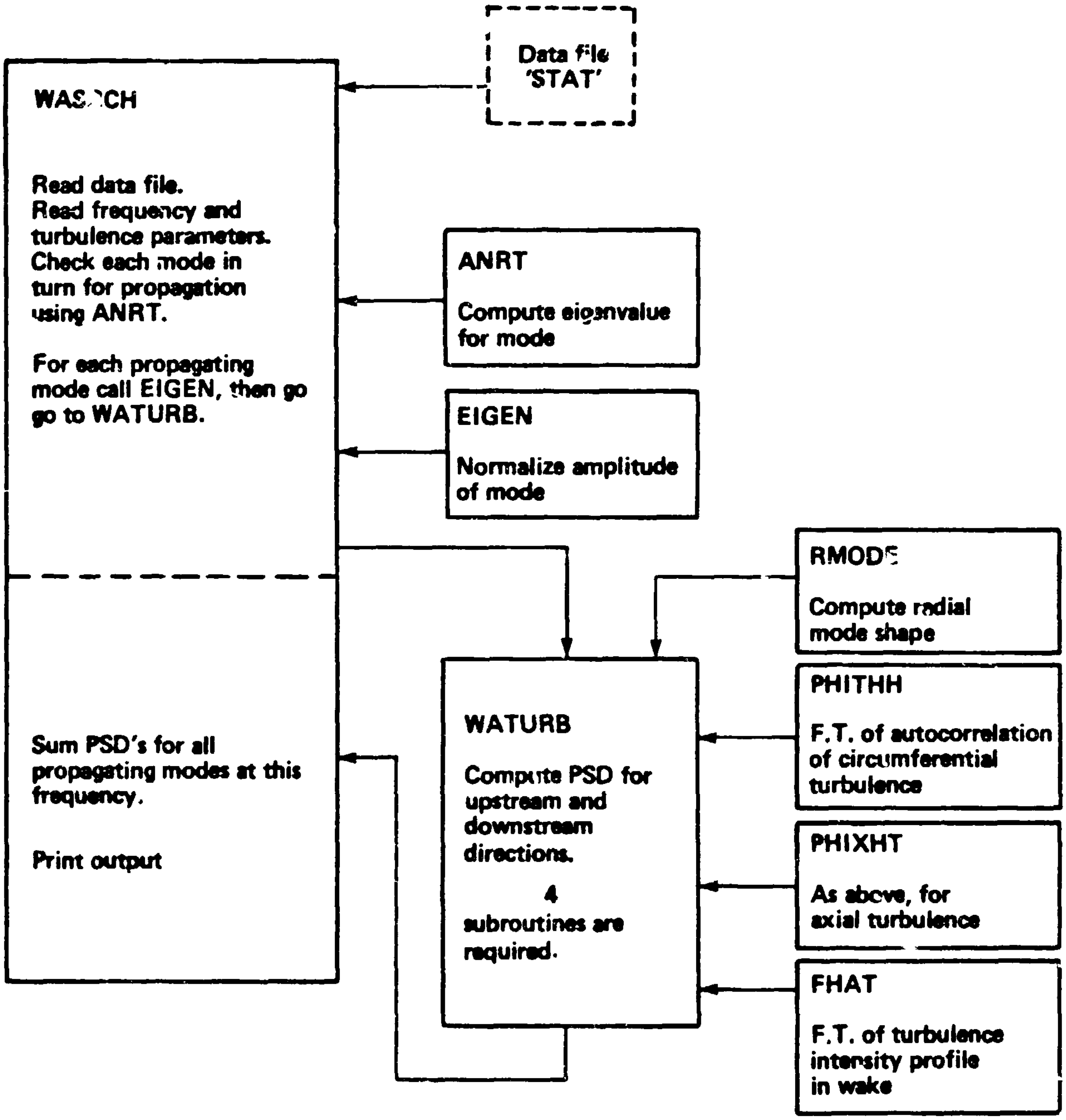

Figure 5. Block Diagram of Rotor Wake Turbulence Program;
Part B: Noise Calculation

ORIGINAL PAGE IS
OF POOR QUALITY

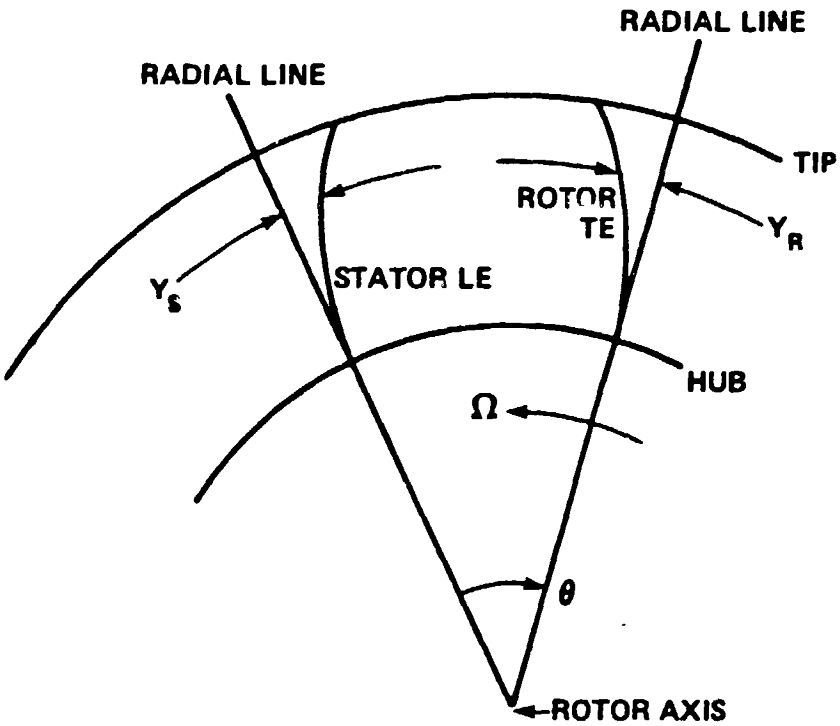

Figure 6. Schematic View of a Rotor Blade TE and Stator Vane LE Looking Down the Turbofan X Axis, in Stator-Fixed Coordinates

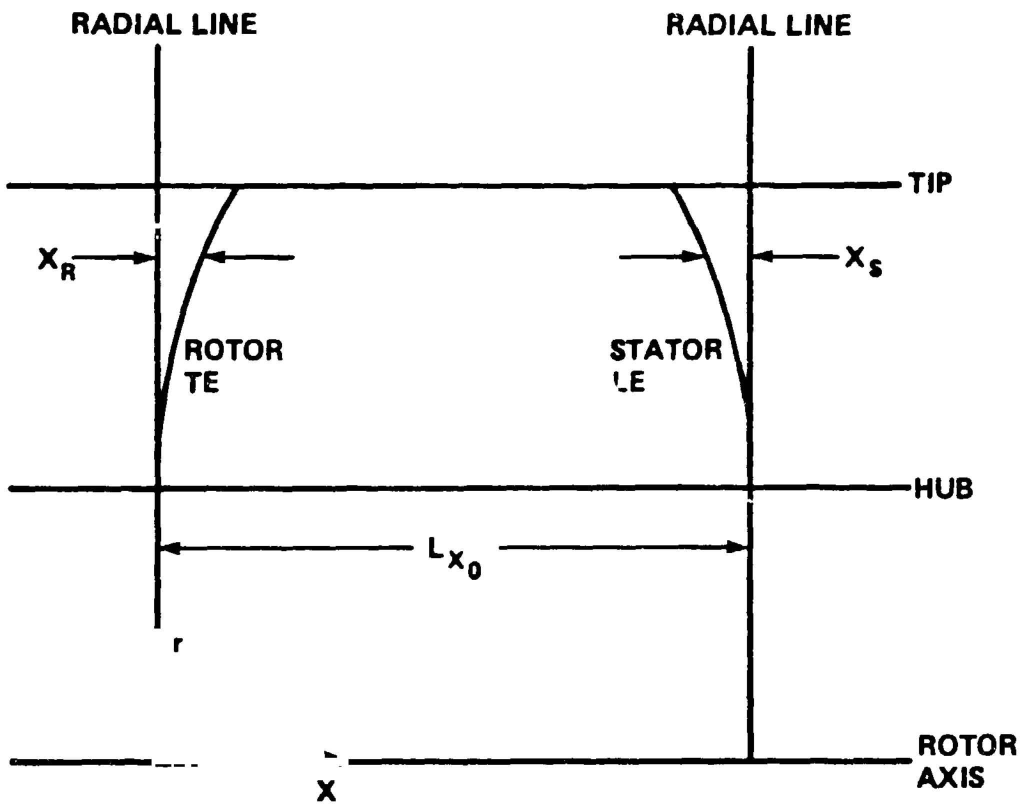

Figure 7. Schematic View of a Rotor Blade TE and Stator Vane LE Looking Perpendicular to the Rotor Axis

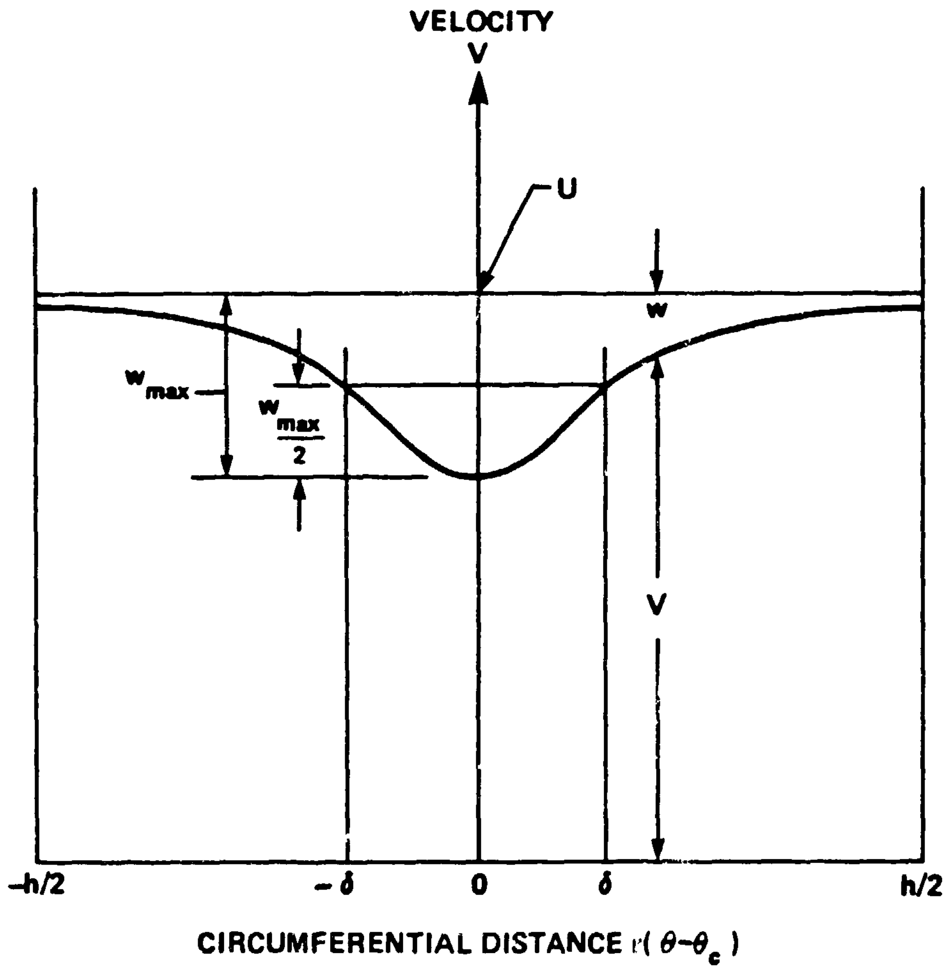

Figure 8. Typical Wake Profile

OR G!NAL PAGE IS
OF POOR QUALITY

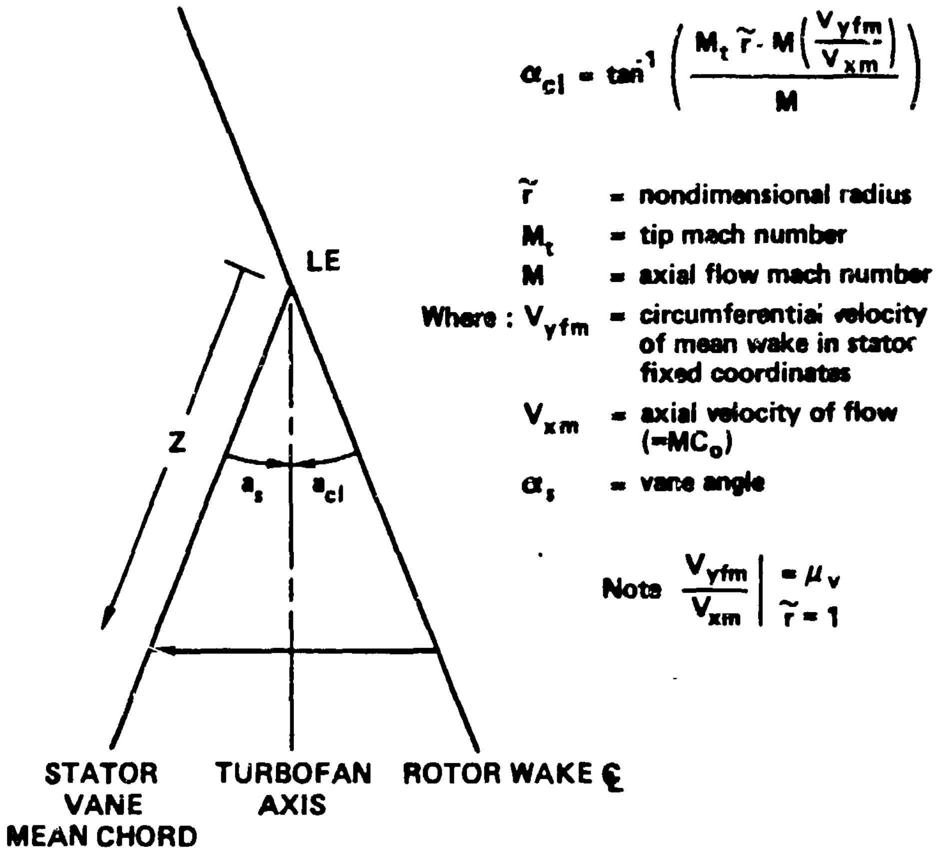

Figure 9. Schematic Diagram of the Stator Vane Mean Chord at the Instant When the Wake Centerline Intersects the Leading Edge

TABLES

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 1. INPUT/OUTPUT DATA FOR THE MEAN ROTOR WAKE PROGRAM

This program calculates the complex modal amplitude and the modal sound power generated by the interaction of the mean rotor wakes with the stator vanes. All input and output data are nondimensional. The input data consist of geometric, performance, and computational parameters.

A. Geometric Data

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
1.	NBLADE	B	No. of rotor blades	Integer
2.	NVANE	V	No. of stator vanes	Integer
3.	SIGMAR	σ_r	Ratio of hub radius to duct radius	Floating Point
4.	SIGMAC	σ_c	Ratio of tip chord of *stator* vane to duct radius	Floating Point
5.	LXO	d/r_{duct}	Ratio of rotor-stator spacing to duct radius	Floating Point
6.			Vector of data for a number, NDAT (an integer variable) of *stator* vane radii	
a.	R	$\tilde{r}$	Nondimensional radius = r/r_{duct}	Floating Point
b.	C	$\tilde{c}$	Nondimensional vane chord = c/c_{tip}	Floating Point
c.	ALPHAS	α_s	Vane angle (degrees)	Floating Point
d.	YS*	$\tilde{Y}_S$	Nonradial variation of stator leading edge = Y_S/c_{tip}	Floating Point
e.	YR*	$\tilde{Y}_R$	Nonradial variation of rotor trailing edge = Y_R/c_{tip}	Floating Point
f.	XS**	$\tilde{X}_S$	Axial variation of stator leading edge = X_S/c_{tip}	Floating Point
g.	XR**	$\tilde{X}_R$	Axial variation of rotor trailing edge = X_R/c_{tip}	Floating Point

*See Figure 6
**See Figure 7

TABLE 1. (Cont.)

B. Performance Data

	FORTRAN NAME	SYMBOL	DEFINITION	FORTRAN VARIABLE TYPE
1.	MT	M_t	Tip Mach number	Floating Point
2.	MA	M	Axial flow Mach number	Floating Point
3.	IVOR	-	Switch for rotor wake specification = 0 for free vortex distribution = 1 for specified input velocity distribution	Integer
4.			(for IVOR = 1) Vector of data for a number, NVELO, (an integer variable) of *vane* radii	
	R	$\tilde{r}$	a. Nondimensional radius	Floating Point
	XVEL	$\frac{V_{yfm}}{V_{xm}}$	b. Ratio of circumferential velocity to axial flow velocity of the mean wake (See Fig. 9)	Floating Point
5.	MUV	μ_v	(for IVOR = 0) Ratio of circumferential velocity at the duct radius to the axial flow velocity	Floating Point
6.	WWIDTH	δ/h	Ratio of wake half-width to wake spacing (See Fig. 8)	Floating Point
7.	VELDEF	Wmax/U	Ratio of maximum wake velocity deficit to the streamwise flow velocity (assumed constant with radius). (See Fig. 8)	Floating Point

TABLE 1. (Cont.)

C. Computational Data

These data set the number of integration stations in the chordwise and spanwise directions (and, indirectly, the accuracy of the results). The required accuracy for the Bessel functions used in the mode shape and eigenvalue routines are also specified.

	FORTRAN NAME	VARIABLE	FORTRAN VARIABLE TYPE
1.	NRAD	No. of integration stations in the radial direction (odd number)	Integer
2.	NCHORD	No. of integration stations in the chordwise direction (even number)	Integer
3.	EB	Absolute accuracy of all Bessel function calculations	Floating Point
4.	EC	Absolute accuracy of convergence to root of eigenvalue equation	Floating Point

Program Output

The output lists the complex nondimensional pressure amplitude for each propagating mode. The amplitude is nondimensionalized by the dynamic pressure of the mean axial flow $\rho_o U^2/2$. Also given is the *relative* modal sound *power* level. To recover the conventional sound power level (dB re 10^{-12} watt) add

$$10 \log_{10} \left[\frac{\rho_o c_o^3 M^4 r_{duct}^2}{8 \cdot 10^{-12}} \right]$$

to the computed relative power. This step is required because all computations performed in the program are nondimensional with absolute values of ρ_o, c_o, and r_{duct} unknown.

TABLE 2. IMPORTANT INTERNAL PROGRAM VARIABLES MEAN ROTOR WAKE PROGRAM

(Used in addition to the INPUT/OUTPUT variables.)

FORTRAN NAME	SYMBOL	VARIABLE
AMN*	$A_{m,n}$	Weighting factors of mode shape $\psi = A_{m,n}J_m + B_{m,n}Y_m$
BMN*	$B_{m,n}$	
ARG*	$X_{m,n}$	Argument of the mode shape function ψ $(=\kappa_{m,n}r)$
B*	b	Nondimensional vane semi-chord = $c/2c_{tip}$
BETASQ*	β^2	$1-M^2$, where M is the axial flow Mach no.
DELTAP	$\Delta\bar{p}$	Nondimensional complex pressure along vane chord
DELTAR*	$\Delta\tilde{r}$	Discrete radial interval for integration
GAMMA	Γ	Nondimensional wave number of the convected gust
GMNS*	$\tilde{\gamma}_{m,n,s}$	Nondimensional acoustic wavenumber for moving medium (See. Eq. [illegible] of Vol. 1)
H1	h_1	Nondimensional stator vane spacing in direction parallel to vane (See Fig. B-1 of Vol. 1)
H2	h_2	Nondimensional stator vane spacing in direction perpendicular to vane (See Fig. B-1 of Vol. 1)
IER*, IERPS1	-	Error codes from Bessel function subroutines
IR*	-	Counter for integration stations in radial direction
IZ*	-	Counter for integration stations in chordwise direction
KMNS*	$K_{m,n,s}$	Nondimensional acoustic wavenumber = $r_{duct} \cdot K_{m,n,s}$ (See Eq. 88 of Vol. 1)

TABLE 2. (Cont.)

ORIGINAL PAGE IS OF POOR QUALITY

KX	k_x	Nondimensional gust wavenumber in the axial direction
MM*	m	Circumferential mode number
MR	M_r	Relative Mach number of mean wake (vector sum of rotational and axial components)
MYM	-	Relative circumferential Mach no. of mean wake at $\tilde{r}$.
NN*	n	Radial mode number
PMNS	-	Sum of terms comprising the radial integral
PSI*, OR PSISTO*	ψ	Nondimensional radial mode shape as a function of radius
QMNS	-	Sum of terms comprising the chordwise integral
R*	$\tilde{r}$	Nondimensional radius = r/r_{duct}
RELPRL	-	Relative modal sound power level at harmonic s
REPLWR, SPWR	-	Relative modal sound power at harmonic s
SB	sB	Product of harmonic number and no. of blades
SS	s	Harmonic number (=1,2,3) multiple of the rotor blade passage frequency
WH	$\hat{w}$	Complex nondimensional amplitude of gust velocity impinging on stator vane
WHTC	-	Weighting factor for Simpson Integration method used in chordwise integral
XMN*	$\kappa_{m,n} r_{duct}$	Nondimensional root of duct eigenvalue equation (See Eq. 6 of Vol. 1)

*These variables are also used in the inlet and wake turbulence programs, but are listed here only.

TABLE 3. UTILITY SUBROUTINES

A. Subroutines common to the Mean Wake, Inlet, and Wake Turbulence programs.

1. ANRT (ANnular RooT), ANFU (ANnular FUnction), ISIGN
   Programs find the roots $\kappa_{m,n}$ of the boundary value equation for the annular duct:

   $$J'_m(\kappa_{m,n}\, r_{duct})\, Y'_m(\kappa_{m,n}\, r_{hub}) - J'_m(\kappa_{m,n}\, r_{hub})\, Y'_m(\kappa_{m,n}\, r_{duct}) = 0$$

2. BESJ, BESY (BESsel functions of the first [J] and the second [Y] kind)

   BESJ computes the mth order Bessel function of the first kind, Jm(x).

   BESY computes the mth order Bessel function of the second kind, Ym(x).

   An error code is returned along with each answer. The error code is:

   ERR = 0 No error (answer is correct to given accuracy)
   = 1 m was negative (must be ≥ 0)
   = 2 Argument was negative (must be ≥ 0)
   = 3 Required accuracy not obtained
   = 4 Range of n compared to the argument was not correct (see program listing).

3. EIGEN

   The program normalizes the weighting factors $A_{m,n}$ and $B_{m,n}$ used in the mode function.

   $$\psi = A_{m,n}\, J_m + B_{m,n}\, Y_m.$$

   The orthogonality condition is

   $$\int_A \psi_{i,j}\, \psi^*_{m,n}\, dA = \begin{cases} \Gamma_{m,n} & \text{if } i = m \text{ and } j = n \\ 0 & \text{otherwise} \end{cases}$$

   The normalizing condition used is that

   $$\bar{\Gamma}_{m,n} \equiv \frac{\Gamma_{m,n}}{\pi(r_{duct}^2 - r_{hub}^2)} = 1 \text{ (non-dimensional).}$$

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 3. (Cont.)

4. RMODE

   Program computes the radial mode shape ψ for the m, nth mode of the annular duct. (See Eq. 4 of Vol. 1)

5. KERNEL

   Program computes the kernel for the Green's function formulation of the cascade routine. (See Eqs. B-25 and B-26 of Vol. 1) Used in conjunction with PRES or CASC.

B. Subroutine used specifically for the Mean Wake Routine

1. PRES (PRESsure distribution)

   The program specifies the complex pressure distribution across the stator vane chord as produced by a velocity gust. The input parameters include the gust wavenumber, the wake velocity, and the stator vane spacings.

C. Subroutines used specifically for the Inlet Turbulence Routine

1. SCALLX (SCALe Length, X direction)

   Program yields the turbulence length scale in the axial direction as a function of radial position. The routine is actually a program input whereby a particular turbulence distribution is loaded.

2. SCALLR (SCALe Length, Radial direction)

   Program yields the turbulence length scale in the radial direction as a function of radial position. The routine is an input parameter.

3. STHETA (Scale length, THETA direction)

   Program yields the turbulence length scale in the circumferential direction as a function of radial position. The routine is an input parameter.

4. EPSD (ϵ_D)

   Program yields the inlet RMS turbulence intensity as a function of radial position. The routire is an input parameter.

5. PHITHH ($\hat{\phi}_\theta$)

   The program computes the Fourier transform of the correlation function in the circumferential direction. This transform is usually a fairly simple analytic function such as an exponential.

6. PHIXHT ($\hat{\phi}_x$)

   The program computes the Fourier transform of the correlation function in the axial direction.

D. Subroutines used for the Inlet and Wake Turbulence Routines

1. CASC (CASCade function)

   The program calculates the complex pressure distribution across a blade chord as produced by a velocity gust. The input parameters include the gust wavenumber, the wake velocity, and the blade spacings. This program is a version of PRES (listed above) specialized for use with the extended range of arguments in the turbulence routines.

E. Subroutine used specifically for the Wake Turbulence Routine

1. FHAT ($\hat{f}$)

   The program computes the Fourier transform of the modulating function for the turbulent wake. The transform is usually a simple analytic function depending on the turbulence model in use.

F. IMSL Library Subroutine called by Subroutine KERNEL

IMSL Library routine LEQTIC is called by KERNEL to solve a set of simultaneous equations. The documentation provided by IMSL follows on the next three pages.

TABLE 3. *(Cont.)* DESCRIPTION OF IMSL LIBRARY ROUTINE LEQT1C.

```
IMSL ROUTINE NAME   - LEQT1C

PURPOSE             - LINEAR EQUATION SOLUTION - COMPLEX MATRIX -
                        SPACE ECONOMIZER SOLUTION

USAGE               - CALL LEQT1C (A,N,IA,B,M,IB,IJOB,WA,IER)

ARGUMENTS    A      - INPUT COMPLEX N BY N MATRIX CONTAINING THE
                        COMPLEX COEFFICIENTS OF THE EQUATION AX = B.
                      ON OUTPUT, A CONTAINS THE L-U DECOMPOSITION OF
                        A ROWWISE PERMUTATION OF THE INPUT MATRIX A.
             N      - ORDER OF MATRIX A. (INPUT)
             IA     - ROW DIMENSION OF MATRIX A EXACTLY AS
                        SPECIFIED IN THE DIMENSION STATEMENT IN THE
                        CALLING PROGRAM. (INPUT)
             B      - INPUT COMPLEX N BY M MATRIX CONTAINING THE
                        COMPLEX VALUED RIGHT HAND SIDES OF THE
                        EQUATION AX = B.
                      ON OUTPUT, THE SOLUTION MATRIX X REPLACES B.
                        IF IJOB=1, B IS NOT USED.
             M      - NUMBER OF RIGHT HAND SIDES (COLUMNS IN B).
                        (INPUT)
             IB     - ROW DIMENSION OF MATRIX B EXACTLY AS SPECIFIED
                        IN THE DIMENSION STATEMENT IN THE CALLING
                        PROGRAM. (INPUT)
             IJOB   - INPUT OPTION PARAMETER.  IJOB=I IMPLIES WHEN
                        I=0, FACTOR THE MATRIX AND SOLVE THE
                          EQUATION AX=B.
                        I=1, FACTOR THE MATRIX A.
                        I=2, SOLVE THE EQUATION AX=B.  THIS
                          OPTION IMPLIES THAT LEQT1C HAS ALREADY
                          BEEN CALLED USING IJOB=0 OR 1 SO THAT
                          THE MATRIX HAS ALREADY BEEN FACTORED.  IN
                          THIS CASE OUTPUT MATRIX A MUST HAVE BEEN
                          SAVED FOR REUSE IN THE CALL TO LEQT1C.
             WA     - WORK AREA OF LENGTH N CONTAINING THE PIVOT
                        INDICES.
             IER    - ERROR PARAMETER. (OUTPUT)
                      TERMINAL ERROR
                        IER=129 INDICATES THAT MATRIX A IS
                          ALGORITHMICALLY SINGULAR.  (SEE THE
                          CHAPTER L PRELUDE.)

PRECISION/HARDWARE  - SINGLE AND DOUBLE/H32
                    - SINGLE/H36,H48,H60

REQD. IMSL ROUTINES - UERTST,UGETIO

NOTATION            - INFORMATION ON SPECIAL NOTATION AND
                        CONVENTIONS IS AVAILABLE IN THE MANUAL
                        INTRODUCTION OR THROUGH IMSL ROUTINE UHELP

REMARKS  1.  WHEN IJOB=1, ARGUMENTS B, M AND IB ARE NOT USED BY
             LEQT1C.
         2.  INPUT MATRIX A IS DESTROYED WHEN IJOB=0 OR 1. WHEN
             IJOB=0 OR 2, B IS REPLACED WITH THE SOLUTION X.
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 3. *(Cont.)* DESCRIPTION OF IMSL LIBRARY ROUTINE LEQT1C.

3. LEQT1C CAN BE USED TO COMPUTE THE INVERSE OF A COMPLEX MATRIX. THIS IS DONE BY CALLING LEQT1C WITH M=N, B=THE N BY N IDENTITY MATRIX AND IJOB=0. WHEN N IS LARGE, IT MAY BE MORE PRACTICAL TO COMPUTE THE INVERSE A COLUMN AT A TIME. TO DO THIS, FIRST CALL LEQT1C WITH IJOB=1 TO FACTOR A. MAKE SUCCEEDING CALLS WITH M=1, B =A COLUMN OF THE IDENTITY MATRIX AND IJOB=2. B WILL BE REPLACED BY THE CORRESPONDING COLUMN OF A INVERSE.
4. THE DETERMINANT OF A CAN BE COMPUTED AFTER LEQT1C HAS BEEN CALLED AS FOLLOWS

```
  DET = (1.0,0.0)
  DO 5 I = 1,N
     IPVT = WA(I)
     IF (IPVT .NE. I) DET = -DET
     DET = DET*A(I,I)
5 CONTINUE
```

Algorithm

For a given N by N complex matrix A and an N by M complex matrix B, LEQT1C computes an LU decomposition of a rowwise permutation of A and solves the system of equations AX=B. Row equilibration and partial pivoting are used.

See reference:

Forsythe, George and Moler, Cleve B., Computer Solution of Linear Algebraic Systems, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1967.

Example

This example shows how to solve a system of 3 equations in 3 unknowns using LEQT1C.

The equations are

```
(1.+i3.)X(1) + (2.5-i3.5)X(2) + (1.+i)X(3)   = (1.-i2.)
(6.+i2.)X(1) + (-4.+i3.)X(2)                 = (3.-i4.)
(5.-i)  X(1) + (1.+i)  X(2) + (3.+i3.)X(3)   = (5.-i6.)
```

and the unknowns are X(1),X(2), and X(3).

In this case, N=3 (A is 3 by 3), and M=1 (B is 3 by 1, one right hand side).

Input:

```
INTEGER   N,IA,M,IB,IJOB,IER
COMPLEX   A(3,3),B(3)
REAL      WA(3)
IA = 3
IB = 3
N  = 3
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 3. *(Cont.)* DESCRIPTION OF IMSL LIBRARY ROUTINE LEQT1C.

```
A  =      (1.,3.)    (2.5,-3.5)    (1.,1.)
          (6.,2.)    (-4.,3.)      (0.,0.)
          (5.,-1.)   (1.,1.)       (3.,3.)

M  = 1

       (1.,-2.)
B  =   (3.,-4.)
       (5.,-6.)

IJOB=0
CALL LEQT1C (A,N,IA,B,M,IB,IJOB,WA,IER)
   :
END

Output:

IER = 0
          (-.32881, -.68795)
B    =    (-.48127, -.55729)
          ( .90092, -1.5177)

Hence the solution is

X(1) = -.32881 -i0.68795
X(2) = -.48127 -i0.55729
X(3) =  .90092 -i1.5177
```

TABLE 4. SAMPLE EXECUTION OF THE MEAN ROTOR WAKE PROGRAM


```
NVANE=
11
NVANE= 11
NBLADE=
15
NBLADE= 15
HUB RADIUS DIVIDED BY DUCT RADIUS =
.484
SIGMAR=  0.48400000E+00
VANE TIP CHORD DIVIDED BY DUCT RADIUS =
.408
SIGMAC=  0.40800000E+00
LX0 =
.6201
LX0=  0.62010000E+00
NDAT =
7
NDAT=  7
VANE GEOMETRY MATRIX INPUT
FEED MATRIX IN ONE ROW AT A TIME, FROM HUB TO TIP
ROW=  1   R/RDUCT =
.484
C/CTIP =
1.
ALPHAS (DEGREES)=
16.3
YS/CTIP =
0.
YR/CTIP =
0.
XS/CTIP =
0.
XR/CTIP =
0.
ROW=  2   R/RDUCT =
.56
C/CTIP =
1.
ALPHAS (DEGREES)=
15.67
YS/CTIP =
.027
YR/CTIP =
-.033
XS/CTIP =
-.001
XR/CTIP =
-.02
```

TABLE 4. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

ROW= 3 R/RDUCT =
.739
C/CTIP =
1.
ALPHAS (DEGREES)=
14.07
YS/CTIP =
.093
YR/CTIP =
-.118
XS/CTIP =
-.005
XR/CTIP =
-.057
ROW= 4 R/RDUCT =
.92
C/CTIP =
1.
ALPHAS (DEGREES)=
12.57
YS/CTIP =
.158
YR/CTIP =
-.213
XS/CTIP =
-.008
XR/CTIP =
-.073
ROW= 5 R/RDUCT =
.946
C/CTIP =
1.
ALPHAS (DEGREES)=
12.36
YS/CTIP =
.163
YR/CTIP =
-.226
XS/CTIP =
-.008
XR/CTIP =
-.07
ROW= 6 R/RDUCT =
.973
C/CTIP =
1.

TABLE 4. (Cont.)


```
ALPHAS (DEGREES)=
12.15
YS/CTIP =
.177
YR/CTIP =
-.242
XS/CTIP =
-.009
XR/CTIP =
-.067
ROW=  7   R/RDUCT =
1.
C/CTIP =
1.
ALPHAS (DEGREES)=
11.92
YS/CTIP =
.187
YR/CTIP =
-.262
XS/CTIP =
-.009
XR/CTIP =
-.064
VANE GEOMETRY MATRIX IS
 0.484E+00 0.100E+01 0.284E+00 0.000E+00 0.000E+00 0.000E+00 0.000E+00
 0.560E+00 0.100E+01 0.273E+00 0.270E-01-0.330E-01-0.100E-02-0.200E-01
 0.739E+00 0.100E+01 0.246E+00 0.930E-01-0.118E+00-0.500E-02-0.570E-01
 0.920E+00 0.100E+01 0.219E+00 0.158E+00-0.213E+00-0.800E-02-0.730E-01
 0.946E+00 0.100E+01 0.216E+00 0.163E+00-0.226E+00-0.800E-02-0.700E-01
 0.973E+00 0.100E+01 0.212E+00 0.177E+00-0.242E+00-0.900E-02-0.670E-01
 0.100E+01 0.100E+01 0.208E+00 0.187E+00-0.262E+00-0.900E-02-0.640E-01
TYPE 1 TO INPUT ROTOR FLOW VELOCITY,0 FOR FREE VORTEX
1
IVOR=  1
NUMBER OF RADII FOR SPECIFYING MEAN ROTOR FLOW=
8
NVELO=  8
INPUT MATRIX FOR MEAN ROTOR FLOW
FEED MATRIX IN ONE ROW AT A TIME,FROM HUB TO TIP
ROW=  1 R/RDUCT=
.484
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.808
ROW=  2 R/RDUCT=
.533
```

TABLE 4. (Cont.)

ORIGINAL PAGE IS OF POOR QUALITY

```
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.700
ROW=  3 R/RDUCT=
.612
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.784
ROW=  4 R/RDUCT=
.834
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.712
ROW=  5 R/RDUCT=
.918
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.688
ROW=  6 R/RDUCT=
.946
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.686
ROW=  7 R/RDUCT=
.974
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.713
ROW=  8 R/RDUCT=
1.
MEAN CIRCUMFERENTIAL VELOCITY RATIO=
.713
MEAN ROTOR FLOW VELOCITY MATRIX IS
  .4840E+00 .8080E+00
  .5330E+00 .7880E+00
  .6120E+00 .7840E+00
  .8340E+00 .7120E+00
  .9180E+00 .6880E+00
  .9460E+00 .6860E+00
  .9740E+00 .7130E+00
  .1000E+01 .7130E+00
MT =
.508
MT=  0.50800000E+00
MA =
.323
MA =  0.32300000E+00
WAKE WIDTH =
.172
WWIDTH=  0.17200000E+00
MAX VEL DEFICIT DIVIDED BY UBAR
.27
VELDEF=  0.27000000E+00
```

TABLE 4. (Cont.)


```
NUMBER OF RADIAL STATIONS =
7
NRAD=   7
NUMBER OF CHORDWISE STATIONS =
8
NCHORD=   8
ACCURACY OF BESSEL FN =
.001
EB=  0.10000000E-02
ACCURACY OF CONVERGENCE TO ROOT XMN =
.0001
EC=  0.10000000E-03

S = 3
XMN(MAX) FOR PROPAGATION =  0.24154718E+02
%FRSAPR Floating underflow      PC= 6115

%FRSAPR Floating underflow      PC= 6163


MODE DATA

S =  3     M =  1     N =  1     XMN =  0.13709056E+01
CUTOFF RATIO FOR MODE= .1762E+02
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.15152305E+01     BMN = -0.42119466E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =    1    N =    1    S =    3
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0

UPSTREAM
GAMMA M,N,SB  =  0.33725477E+02
MODE AMPLITUDE =  0.13613387E-03  0.33358998E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7364E+02

DOWNSTREAM
GAMMA M,N,SB = -0.17237771E+02
MODE AMPLITUDE = -0.47378941E-04 -0.72085580E-05
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7715E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 4. (Cont.)

```
M=  -1   N=   1   S=  -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7364E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7715E+02

MODE DATA

S =  3    M =  1    N =  2    XMN =  0.63900327E+01
CUTOFF RATIO FOR MODE= .3780E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.19356938E+01    BMN =  0.32269858E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =    1    N =    2    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.32857319E+02
MODE AMPLITUDE = -0.11049067E-03 -0.91503578E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7351E+02

DOWNSTREAM
GAMMA M,N,SB = -0.16369614E+02
MODE AMPLITUDE =  0.56242548E-04  0.26364856E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7519E+02

M=  -1   N=   2   S=  -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7351E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7519E+02

MODE DATA

S =  3    M =  1    N =  3    XMN =  0.12324612E+02
CUTOFF RATIO FOR MODE= .1960E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.43514410E+01    BMN =  0.30726765E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

TABLE 4. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

```
M =     1     N =     3     S =     3

UPSTREAM
GAMMA M,N,SB  =  0.30194305E+02
MODE AMPLITUDE = -0.67274867E-04 -0.89817303E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8036E+02

DOWNSTREAM
GAMMA M,N,SB = -0.13706599E+02
MODE AMPLITUDE =  0.39056430E-04  0.46102940E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7634E+02

M=   -1    N=     3    S=    -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8036E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7634E+02

MODE DATA

S =  3     M =  1     N =  4     XMN =  0.18362274E+02
CUTOFF RATIO FOR MODE= .1315E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.61381751E+01     BMN =  0.22116971E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =     1     N =     4     S =     3

UPSTREAM
GAMMA M,N,SB  =  0.24825892E+02
MODE AMPLITUDE =  0.88119635E-06  0.55164443E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.1173E+03

DOWNSTREAM
GAMMA M,N,SB = -0.83381859E+01
MODE AMPLITUDE =  0.10458061E-04  0.45172844E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8064E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 4. (Cont.)

```
M=   -1   N=    4   S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.1173E+03

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8064E+02

LARGEST PROPAGATING N FOR THIS M =  4

MODE DATA

S =  3    M =-10    N =  1    XMN =  0.11769701E+02
CUTOFF RATIO FOR MODE= .2052E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.54757755E+01    BMN = -0.30348594E-02
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =  -10    N =    1    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.30531741E+02
MODE AMPLITUDE = -0.90896823E-04  0.41207877E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7690E+02

DOWNSTREAM
GAMMA M,N,SB = -0.14044035E+02
MODE AMPLITUDE = -0.64741460E-03 -0.25779423E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5498E+02

M=   10   N=    1   S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7690E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5498E+02

MODE DATA

S =  3    M =-10    N =  2    XMN =  0.16348282E+02
CUTOFF RATIO FOR MODE= .1478E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.48044727E+01    BMN = -0.37694076E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

```
M =  -10    N =    2    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.27032509E+02
MODE AMPLITUDE =  0.36213808E-04  0.13253067E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8559E+02

DOWNSTREAM
GAMMA M,N,SB = -0.10544803E+02
MODE AMPLITUDE =  0.54190425E-03  0.25744146E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5754E+02

M=   10    N=    2    S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8559E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5754E+02

MODE DATA

S =  3    M =-10    N =  3    XMN =  0.19710760E+02
CUTOFF RATIO FOR MODE= .1225E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.45357506E+01    BMN = -0.21469782E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =  -10    N =    3    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.22996575E+02
MODE AMPLITUDE =  0.50912282E-05  0.44798741E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8492E+02

DOWNSTREAM
GAMMA M,N,SB = -0.65088688E+01
MODE AMPLITUDE = -0.32015018E-03 -0.56725690E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5844E+02
```

TABLE 4. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

```
M=   10   N=    3   S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.8492E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5844E+02

MODE DATA

S =  3     M =-10     N =  4     XMN =  0.23588640E+02
CUTOFF RATIO FOR MODE= .1024E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.62652037E+01     BMN = -0.98372497E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =  -10     N =     4     S =     3

UPSTREAM
GAMMA M,N,SB  =  0.13736991E+02
MODE AMPLITUDE = -0.50110748E-03 -0.29130448E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6613E+02

DOWNSTREAM
GAMMA M,N,SB =  0.27507146E+01
MODE AMPLITUDE = -0.34158767E-03  0.85520260E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6090E+02

M=   10   N=    4   S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6613E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6090E+02

LARGEST PROPAGATING N FOR THIS M =  4

MODE DATA

S =  3     M = 12     N =  1     XMN =  0.13878611E+02
CUTOFF RATIO FOR MODE= .1740E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.60892918E+01     BMN = -0.63711792E-03
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

TABLE 4. (Cont.)


```
M =   12    N =    1    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.29133072E+02
MODE AMPLITUDE =  0.88696711E-04  0.15987079E-05
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7808E+02

DOWNSTREAM
GAMMA M,N,SB = -0.12645367E+02
MODE AMPLITUDE =  0.95049415E-04  0.45870526E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5892E+02

M=  -12   N=    1   S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7808E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5892E+02

MODE DATA

S =  3    M = 12    N =  2    XMN =  0.18713666E+02
CUTOFF RATIO FOR MODE= .1291E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.53619937E+01     BMN = -0.12842323E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =   12    N =    2    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.24381333E+02
MODE AMPLITUDE =  0.31973551E-03 -0.67991251E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6745E+02

DOWNSTREAM
GAMMA M,N,SB = -0.78936267E+01
MODE AMPLITUDE =  0.14232207E-03 -0.40642054E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6146E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 4. (Cont.)

```
M=  -12    N=     2    S=    -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=      -0.6745E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=      -0.6146E+02

MODE DATA

S =  3     M = 12     N =  3     XMN =  0.22297030E+02
CUTOFF RATIO FOR MODE= .1083E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.50662141E+01     BMN = -0.14574338E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =    12     N =     3     S =     3

UPSTREAM
GAMMA M,N,SB  =  0.18059368E+02
MODE AMPLITUDE = -0.38293325E-03 -0.15805489E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=      -0.6696E+02

DOWNSTREAM
GAMMA M,N,SB = -0.15716621E+01
MODE AMPLITUDE = -0.11038813E-02  0.41711175E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=      -0.5569E+02

M=  -12    N=     3    S=    -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=      -0.6696E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=      -0.5569E+02

LARGEST PROPAGATING N FOR THIS M =  3

MODE DATA

S =  3     M =-21     N =  1     XMN =  0.23254819E+02
CUTOFF RATIO FOR MODE= .1039E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.85042393E+01     BMN = -0.38116337E-05
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 4. (Cont.)

```
M =  -21    N =    1    S =    3

UPSTREAM
GAMMA M,N,SB  =  0.15145542E+02
MODE AMPLITUDE =  0.63165623E-03 -0.10433352E-02
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5882E+02

DOWNSTREAM
GAMMA M,N,SB =  0.13421642E+01
MODE AMPLITUDE = -0.18827030E-02  0.30355092E-02
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4796E+02

M=   21    N=    1    S=   -3

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5882E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4796E+02

LARGEST PROPAGATING N FOR THIS M =  1
NO MORE PROPAGATING MODES FOR THIS VALUE OF S

S = 2
XMN(MAX) FOR PROPAGATION =  0.16103146E+02

MODE DATA

S =  2    M = -3    N =  1    XMN =  0.39866274E+01
CUTOFF RATIO FOR MODE= .4039E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.26023450E+01     BMN = -0.32589376E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =   -3    N =    1    S =    2
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
```

TABLE 4. (Cont.)


```
UPSTREAM
GAMMA M,N,SB  =  0.21981403E+02
MODE AMPLITUDE =  0.13526694E-03 -0.11438489E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7167E+02

DOWNSTREAM
GAMMA M,N,SB = -0.10989599E+02
MODE AMPLITUDE = -0.12497451E-04 -0.25469007E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6288E+02

M=    3   N=    1   S=   -2

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7167E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.6288E+02

MODE DATA

S =  2    M = -3    N =  2    XMN =  0.77467003E+01
CUTOFF RATIO FOR MODE= .2079E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.35705322E+01     BMN = -0.89511162E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =   -3    N =    2    S =    2

UPSTREAM
GAMMA M,N,SB  =  0.20412827E+02
MODE AMPLITUDE = -0.62938154E-0   0.18019367E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7126E+02

DOWNSTREAM
GAMMA M,N,SB = -0.94210234E+01
MODE AMPLITUDE = -0.56493354E-03  0.30607513E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5566E+02
```

TABLE 4. (Cont.)

```
M=    3   N=    2   S=   -2

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7126E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5566E+02

MODE DATA

S =  2    M = -3    N =  3    XMN =  0.13008803E+02
CUTOFF RATIO FOR MODE= .1238E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN = -0.18754296E+00    BMN =  0.53089865E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =   -3    N =    3    S =    2

UPSTREAM
GAMMA M,N,SB  =  0.15524623E+02
MODE AMPLITUDE = -0.87384578E-05 -0.13593062E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7526E+02

DOWNSTREAM
GAMMA M,N,SB = -0.45328195E+01
MODE AMPLITUDE =  0.50005592E-03  0.24067449E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5971E+02

M=    3   N=    3   S=   -2

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.7526E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5971E+02

LARGEST PROPAGATING N FOR THIS M =  3

MODE DATA

S =  2    M =  8    N =  1    XMN =  0.96413107E+01
CUTOFF RATIO FOR MODE= .1670E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.48101923E+01    BMN = -0.14625735E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

TABLE 4. (Cont.)


```
M =    8    N =    1    S =    2

UPSTREAM
GAMMA M,N,SB  =  0.19124322E+02
MODE AMPLITUDE = -0.15710355E-02  0.39986281E-02
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4444E+02

DOWNSTREAM
GAMMA M,N,SB = -0.81325179E+01
MODE AMPLITUDE =  0.66697707E-03 -0.60581800E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5598E+02

M=   -8    N=    1    S=   -2

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4444E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5598E+02

MODE DATA

S =  2    M =  8    N =  2    XMN =  0.13862791E+02
CUTOFF RATIO FOR MODE= .1162E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.41234860E+01     BMN = -0.85984376E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =    8    N =    2    S =    2

UPSTREAM
GAMMA M,N,SB  =  0.14153511E+02
MODE AMPLITUDE = -0.50227789E-02 -0.46086625E-02
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4172E+02

DOWNSTREAM
GAMMA M,N,SB = -0.31617076E+01
MODE AMPLITUDE = -0.18333959E-02 -0.37698765E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.5006E+02
```

TABLE 4. (Cont.)

M= -8 N= 2 S= -2

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)= -0.4172E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)= -0.5006E+02

LARGEST PROPAGATING N FOR THIS M = 2

MODE DATA

S = 2 M =-14 N = 1 XMN = 0.15975402E+02
CUTOFF RATIO FOR MODE= .1008E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.66675895E+01 BMN = -0.10554324E-03
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M = -14 N = 1 S = 2

UPSTREAM
GAMMA M,N,SB = 0.76348570E+01
MODE AMPLITUDE = -0.15299872E-01 0.33717128E-01
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)= -0.3212E+02

DOWNSTREAM
GAMMA M,N,SB = 0.33569469E+01
MODE AMPLITUDE = 0.13580878E-01 0.28533771E-01
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)= -0.3279E+02

M= 14 N= 1 S= -2

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)= -0.3212E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)= -0.3279E+02

LARGEST PROPAGATING N FOR THIS M = 1
NO MORE PROPAGATING MODES FOR THIS VALUE OF S

S = 1
XMN(MAX) FOR PROPAGATION = 0.80515728E+01

TABLE 4. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

```
MODE DATA

S =  1    M =  4    N =  1    XMN =  0.51995376E+01
CUTOFF RATIO FOR MODE= .1549E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.31336383E+01    BMN = -0.20930871E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M =    4    N =    1    S =    1
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0

UPSTREAM
GAMMA M,N,SB  =  0.92437007E+01
MODE AMPLITUDE = -0.16022921E-01 -0.12244703E-01
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.3113E+02

DOWNSTREAM
GAMMA M,N,SB = -0.37477989E+01
MODE AMPLITUDE = -0.10714520E-02  0.23857642E-02
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4450E+02

M=   -4   N=    1   S=   -1

UPSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.3113E+02

DOWNSTREAM
RELATIVE SOUND POWER LEVEL FOR MODE(DB)=     -0.4450E+02

LARGEST PROPAGATING N FOR THIS M =  1
NO MORE PROPAGATING MODES FOR THIS VALUE OF S
PROBLEM COMPLETED
RELATIVE POWER LEVEL UPSTREAM S= 1          -0.2812E+02
RELATIVE POWER LEVEL UPSTREAM S= 2          -0.2843E+02
RELATIVE POWER LEVEL UPSTREAM S= 3          -0.5381E+02
RELATIVE POWER LEVEL DOWNSTREAM S= 1          -0.4149E+02
RELATIVE POWER LEVEL DOWNSTREAM S= 2          -0.2964E+02
RELATIVE POWER LEVEL DOWNSTREAM S= 3          -0.4253E+02
STOP

END OF EXECUTION
CPU TIME: 4:39.68       ELAPSED TIME: 20:51.21
EXIT.
```

TABLE 5. INPUT/OUTPUT DATA FOR THE INLET TURBULENCE PROGRAMS

This program calculates the modal power spectral density for noise generated by a turbofan rotor subjected to inlet turbulence. All input and output data are nondimensional. The input variables are defined in order to parallel as closely as possible the variables used in the Mean Wake Program. The input data consist of geometric, performance, turbulence, and computational parameters.

A. Geometric Data

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
1.	NBLADE	B	No. of rotor blades	Integer
2.	SIGMAR	σ_r	Ratio of hub radius to duct radius	Floating Point
3.	SIGMAC	σ_c	Ratio of tip chord of *rotor* blade to duct radius	Floating Point
4.			Vector of data for a number, NDAT (an integer variable) of *rotor* blade radii	
a.	R	$\tilde{r}$	Nondimensional radius = r/r_{duct}	Floating Point
b.	C	$\tilde{c}$	Nondimensional blade chord= c/c_{tip}	Floating Point
c.	CHI	X	Blade angle (degrees) [See Fig. 3 of Vol. 1]	Floating Point

B. Performance Data

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
1.	MT	M_t	Tip Mach number	Floating Point
2.	MA	M	Axial flow Mach number	Floating Point
3.		Frequency range data: Pressure distributions are calculated and stored for the range of frequencies from PSTART to PEND in steps of PSTEP. These data are interpolated for calculations at the frequency of interest OMEGA.		

TABLE 5. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
a.	PSTART	-	First frequency for pressure calculation. Must be an integer multiple of shaft frequency. (Usually = NBLADE)	Integer
b.	PEND	-	Final frequency for pressure calculation. See text for description of value required.	Integer
c.	PSTEP	-	Frequency step size in pressure calculation. (Usually = NBLADE)	Integer
d.	OMEGA	ω/Ω	Noise frequency of interest (nondimensionalized on the rotor shaft frequency)	Floating Point

C. Turbulence Data

The inlet turbulence data are loaded into separate function subroutines. This arrangement allows rather complicated radial distributions of turbulence to be modeled.

	FORTRAN FUNCTION NAME	SYMBOL	VARIABLE
1.	SCALLX	$\tilde{L}_x(\tilde{r})$	Axial turbulence length scale (nondimensionalized on the duct radius) as a function of nondimensionalized radius.*
2.	SCALLR	$\tilde{L}_r(\tilde{r})$	Radial turbulence length scale (nondimensionalized on the duct radius) as a function of nondimensional radius.
3.	STHETA	$\tilde{L}_\theta(\tilde{r})$	Circumferential turbulence length scale (nondimensionalized on the duct radius) as a function of nondimensional radius.*
4.	EPSD	$\varepsilon_D(\tilde{r})$	RMS turbulence velocity (nondimensionalized on the mean axial velocity) as a function of nondimensional radius. (See Eq. 94 of Vol. 1)

*See page 48 of Vol. 1.

D. Computational Data

These data set the number of integration stations in the chordwise and spanwise directions (and, indirectly, the accuracy of the results). The required accuracy for the Bessel functions used in the mode shape and eigenvalue routines are also specified.

	FORTRAN NAME	VARIABLE	FORTRAN VARIABLE TYPE
1.	NRAD	No. of radial stations at which pressure distributions are calculated (odd number)	Integer
2.	NRADNU	No. of integration stations in the chord-wise direction (odd number)	Integer
3.	NCHORD	No. of integration stations in the chord-wise direction (even number)	Integer
4.	EB	Absolute accuracy of all Bessel function calculations	Floating Point
5.	EC	Absolute accuracy of convergence to annular root for boundary value problem	Floating Point

Program Output

The output consists of the *relative* power spectral density for each propagating mode and for the sum of all propagating modes at the frequency of interest. The output is given in terms of the nondimensional quantity

$$10 \log_{10}\left[\frac{S_{mn}}{\rho_o c_o^2 r_{duct}}\right]$$

where S_{mn} is the spectral density (watt/Hz), ρ_o is the ambient air density, c_o is the small-signal speed of sound and r_{duct} is the (dimensional) duct radius. To recover the conventional spectral density (dB re 10^{-12} watt/Hz) one must add

$$10 \log_{10}\left[\frac{\rho_o c_o^2 r_{duct}}{10^{-12}\,\text{watt}}\right]$$

to the output of the program. This step is required because all computations performed in the program are nondimensional with absolute values of ρ_o, c_o, and r_{duct} unknown.

TABLE 6. IMPORTANT INTERNAL VARIABLES FOR THE INLET TURBULENCE PROGRAM

(Used in addition to INPUT/OUTPUT Variables)

FORTRAN NAME	SYMBOL	VARIABLE
BJ(N)+	$J_N(X)$	Nth order Bessel function of the first kind
CASCET+	$\Delta\tilde{p}$	Stored values of the chordwise pressure distribution
H+	-	Vector length of blade spacing between corresponding points on any two neighboring blades (See Fig. B.1)
KB+	-	Nondimensional gust wave number $[=M_t b\sigma_c(\frac{\omega}{\Omega}-m)/M_r]$
LR+	$\tilde{L}_r(\tilde{r})$	Ratio of radial turbulence length scale to duct radius
LTHETA	$\tilde{L}_\theta(\tilde{r})$	Ratio of circumferential turbulence length scale to duct radius
LX+	$\tilde{L}_x(\tilde{r})$	Ratio of axial turbulence length scale to duct radius
OMEGA+	ω/Ω	Ratio of noise frequency to rotor shaft frequency
OMEGBC+	-	Nondimensional acoustic wavenumber for reduced frequency $[=M_t\sigma_c b(\frac{\omega}{\Omega}-m)]$
PHIARG+	-	Argument of $\hat{\Phi}_x$ (Argument)
PMNS	-	Sum of terms comprising the radial integral
QMNS	-	Sum of terms comprising the chordwise integral
RELPRL+	-	Relative modal power spectral density level at a given frequency
RELPWR+ or SPWR+	-	Relative modal power spectral density at given frequency
RRRS+	R_s	Weighting factor for Bessel-type integration in chordwise direction
SIGMA+	σ	Interblade phase angle (See page 52 of Vol. 1)
THEARG+	-	Argument of $\hat{\Phi}_\theta$ (Argument)
WHTR+	-	Weighting factor for Simpson integration in radial direction

+These variables are also used in the wake turbulence routine, but are listed here only.

TABLE 7. SAMPLE EXECUTION OF THE INLET TURBULENCE DATA STORAGE ROUTINE

ORIGINAL PAGE IS
OF POOR QUALITY

```
@ROTORS
NBLADE=
15
NBLADE= 15
HUB RADIUS DIVIDED BY DUCT RADIUS =
.46
SIGMAR=  0.46000000E+00
BLADE TIP CHORD DIVIDED BY DUCT RADIUS=
.374
SIGMAC=  0.37400000E+00
NDAT =
10
NDAT= 10
BLADE GEOMETRY MATRIX INPUT
FEED MATRIX IN ONE ROW AT A TIME, FROM HUB TO TIP
ROW=  1   R/RDUCT =
.46
C/CTIP =
.628
CHI (DEGREES)=
5.61
ROW=  2   R/RDUCT =
.514
C/CTIP =
.662
CHI (DEGREES)=
9.73
ROW=  3   R/RDUCT =
.541
C/CTIP =
.68
CHI (DEGREES)=
11.74
ROW=  4   R/RDUCT =
.622
C/CTIP =
.735
CHI (DEGREES)=
17.48
ROW=  5   R/RDUCT =
.73
C/CTIP =
.811
CHI (DEGREES)=
24.22
ROW=  6   R/RDUCT =
.838
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 7. (Cont.)

```
C/CTIP =
.887
CHI (DEGREES)=
30.66
ROW= 7   R/RDUCT =
.919
C/CTIP =
.945
CHI (DEGREES)=
35.44
ROW= 8   R/RDUCT =
.946
C/CTIP =
.959
CHI (DEGREES)=
37.05
ROW= 9   R/RDUCT =
.973
C/CTIP =
.976
CHI (DEGREES)=
38.96
ROW= 10  R/RDUCT =
1.
C/CTIP =
1.
CHI (DEGREES)=
41.14
BLADE GEOMETRY MATRIX IS
 0.460E+00 0.628E+00 0.979E-01
 0.514E+00 0.662E+00 0.170E+00
 0.541E+00 0.680E+00 0.205E+00
 0.622E+00 0.735E+00 0.305E+00
 0.730E+00 0.811E+00 0.423E+00
 0.838E+00 0.887E+00 0.535E+00
 0.919E+00 0.945E+00 0.619E+00
 0.946E+00 0.959E+00 0.647E+00
 0.973E+00 0.976E+00 0.680E+00
 0.100E+01 0.100E+01 0.718E+00
MT =
.508
MT=  0.50800000E+00
MA =
.323
MA =  0.32300000E+00
NUMBER OF RADIAL STATIONS =
5
NRAD=   5
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 7. (Cont.)

```
NUMBER OF CHORDWISE STATIONS =
8
NCHORD=    8
START FREQUENCY(HARMONIC NO. OF SHAFT )=
15
PSTART=  15
END FREQUENCY(HARMONIC NO. OF SHAFT)=
75
P END=  75
FREQUENCY STEP SIZE (HARMONIC ORDERS)=
15
PSTEP=  15
ENTER TITLE FOR DATA FILE(MAX 72 CHARACTERS)
NASA ROTOR 55,NCHORD=8,NRAD=580% DESIGN SPEED/9-25-9\980
ERROR CODE FROM INVERSION ROUTINE=  0
```

(Note: Edited for brevity -- Error Code repeated for 375 inversions.)

```
ERROR CODE FROM INVERSION ROUTINE=  0
STOP

END OF EXECUTION
CPU TIME: 52:32.72      ELAPSED TIME: 1:32:28.46
EXIT.
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 8. SAMPLE EXECUTION OF THE INLET TURBULENCE NOISE CALCULATION

WARNING: This sample is for illustration purposes only. The input data for this execution did not correspond with Table 7 above.

```
@INSRCH
USER-ASSIGNED TITLE OF DATA SET FROM FILE "ROTO" IS
NASA ROTOR 55,NCHORD=8,NRAD=580% DESIGN SPEED/9-25-80
NOISE FREQUENCY/SHAFT FREQUENCY=
15.
OMEGA= .1500E+02
NUMBER OF RADIAL POSITIONS FOR ACOUSTIC COMP.=
7
NRADNU=   7
ACCURACY OF BESSEL FNS=
.001
EB= .1000E-02
ACCURACY OF CONVERGENCE TO ROOT XMN=
.0001
EC= .1000E-03
XMN(MAX) FOR PROPAGATION =  0.80515728E+01

MODE DATA

OMEGA= .1500E+02    M=  0    N=  1    XMN= .0000E+00
PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.10000000E+01     BMN =  0.00000000E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=    0      N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.11255539E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2031E-08
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8692E+02

DOWNSTREAM
GAMMA M,N,SB = -0.57596368E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .4316E-08
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8365E+02

MODE DATA

OMEGA= .1500E+02    M=  0    N=  2    XMN= .5944E+01
CUTOFF RATIO FOR MODE= .1354E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN = -0.17828589E+01     BMN =  0.31951480E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

TABLE 8. (Cont.)

```
M=    0      N=    2    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.84861763E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.4482E-08
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8349E+02

DOWNSTREAM
GAMMA M,N,SB = -0.29907743E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .2133E-08
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8671E+02

LARGEST PROPAGATING N FOR THIS M =  2
%FRSAPR Floating underflow      PC= 17366

%FRSAPR Floating underflow      PC= 17433


MODE DATA

OMEGA= .1500E+02    M=  1   N=  1   XMN= .1396E+01
CUTOFF RATIO FOR MODE= .5769E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.15347161E+01     BMN = -0.40872429E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=    1      N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.11126733E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2481E-07
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.7458E+02

DOWNSTREAM
GAMMA M,N,SB = -0.56308314E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .6363E-07
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.7196E+02
```

TABLE 8. (Cont.)


```
M=   -1   N=    1   OMEGA= .1500E+02

M=   -1       N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.11126733E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.1139E-09
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9943E+02

DOWNSTREAM
GAMMA M,N,SB = -0.56308314E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .2030E-09
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9692E+02

MODE DATA

OMEGA= .1500E+02    M=  1   N=  2   XMN= .6150E+01
CUTOFF RATIO FOR MODE= .1309E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.25524893E+01     BMN =  0.26060464E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=    1       N=    2    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.82391651E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.5133E-07
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.7290E+02

DOWNSTREAM
GAMMA M,N,SB = -0.27432631E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .3747E-07
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.7426E+02

M=   -1   N=    2   OMEGA= .1500E+02

M=   -1       N=    2    OMEGA= .1500E+02
```

TABLE 8. (Cont.)


```
UPSTREAM
GAMMA M,N,SB  =  0.82391651E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.3109E-09
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9507E+02

DOWNSTREAM
GAMMA M,N,SB = -0.27432631E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .4165E-10
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1038E+03

LARGEST PROPAGATING N FOR THIS M =  2

MODE DATA

OMEGA= .1500E+02    M=  2   N=  1   XMN= .2748E+01
CUTOFF RATIO FOR MODE= .2930E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.21049825E+01    BMN = -0.38973734E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=    2      N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.10744835E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2065E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.6665E+02

DOWNSTREAM
GAMMA M,N,SB = -0.52489328E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .7372E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.6132E+02

M=   -2   N=    1   OMEGA= .1500E+02

M=   -2      N=    1    OMEGA= .1500E+02
```

TABLE 8. (Cont.)


```
UPSTREAM
GAMMA M,N,SB  =  0.10744835E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.7070E-11
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1115E+03

DOWNSTREAM
GAMMA M,N,SB = -0.52489328E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .6529E-11
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1119E+03

MODE DATA

OMEGA= .1500E+02    M=  2   N=  2   XMN= .6738E+01
CUTOFF RATIO FOR MODE= .1195E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.36099485E+01    BMN =  0.11001426E+09
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=    2      N=    2    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.74045386E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.1927E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.6715E+02

DOWNSTREAM
GAMMA M,N,SB = -0.19086367E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .3230E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.6491E+02

M=   -2   N=    2   OMEGA= .1500E+02

M=   -2      N=    2    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.74045386E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2280E-10
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1064E+03

DOWNSTREAM
GAMMA M,N,SB = -0.19086367E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .2400E-12
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1262E+03

LARGEST PROPAGATING N FOR THIS M =  2
```

TABLE 8. (Cont.)

MODE DATA

```
OMEGA= .1500E+02    M= 3   N= 1   XMN= .4027E+01
CUTOFF RATIO FOR MODE= .2000E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.26815693E+01     BMN = -0.27568805E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   3     N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.10115297E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.8401E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
SUM OF ALL ERRORS IN RS CALCULATIONS=     0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.6076E+02

DOWNSTREAM
GAMMA M,N,SB = -0.46193948E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .7621E-05
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
SUM OF ALL ERRORS IN RS CALCULATIONS=     0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.5118E+02

M=   -3   N=    1   OMEGA= .1500E+02

M=   -3     N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.10115297E+02
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.5393E-12
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
SUM OF ALL ERRORS IN RS CALCULATIONS=     0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1227E+03

DOWNSTREAM
GAMMA M,N,SB = -0.46193948E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .6330E-13
SUM OF ALL ERRORS IN PSI CALCULATIONS =     0
SUM OF ALL ERRORS IN RS CALCULATIONS=     0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1320E+03

MODE DATA

OMEGA= .1500E+02    M= 3   N= 2   XMN= .7642E+01
CUTOFF RATIO FOR MODE= .1054E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.33544744E+01     BMN = -0.11908489E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

TABLE 8. (Cont.)

M= 3 N= 2 OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB = 0.54275270E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2364E-06
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.6626E+02

DOWNSTREAM
GAMMA M,N,SB = 0.68375036E-01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .1706E-05
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.5768E+02

M= -3 N= 2 OMEGA= .1500E+02

M= -3 N= 2 OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB = 0.54275270E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2331E-11
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1163E+03

DOWNSTREAM
GAMMA M,N,SB = 0.68375036E-01
REL MODAL SOUND POWER SPECTRAL DENSITY= .9788E-12
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1201E+03

LARGEST PROPAGATING N FOR THIS M = 2

MODE DATA

OMEGA= .1500E+02 M= 4 N= 1 XMN= .5230E+01
CUTOFF RATIO FOR MODE= .1539E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.32237284E+01 BMN = -0.16030403E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= 4 N= 1 OMEGA= .1500E+02

TABLE 8. (Cont.)

UPSTREAM
GAMMA M,N,SB = 0.92161443E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.2038E-05
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.5691E+02

DOWNSTREAM
GAMMA M,N,SB = -0.37202422E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .6538E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.4185E+02

M= -4 N= 1 OMEGA= .1500E+02

M= -4 N= 1 OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB = 0.92161443E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.5114E-13
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1329E+03

DOWNSTREAM
GAMMA M,N,SB = -0.37202422E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .3263E-14
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1449E+03

LARGEST PROPAGATING N FOR THIS M = 1

MODE DATA

OMEGA= .1500E+02 M= 5 N= 1 XMN= .6376E+01
CUTOFF RATIO FOR MODE= .1263E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.37079083E+01 BMN = -0.81471940E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= 5 N= 1 OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB = 0.79436739E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.1669E-05
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.5778E+02

TABLE 8. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

DOWNSTREAM
GAMMA M,N,SB = -0.24477719E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .4521E-03
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.3345E+02

M= -5 N= 1 OMEGA= .1500E+02

M= -5 N= 1 OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB = 0.79436739E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.4608E-14
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1434E+03

DOWNSTREAM
GAMMA M,N,SB = -0.24477719E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY= .1763E-14
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1475E+03

LARGEST PROPAGATING N FOR THIS M = 1

MODE DATA

OMEGA= .1500E+02 M= 6 N= 1 XMN= .7484E+01
CUTOFF RATIO FOR MODE= .1076E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.41370714E+01 BMN = -0.38730478E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= 6 N= 1 OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB = 0.58865744E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.4244E-04
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.4372E+02

DOWNSTREAM
GAMMA M,N,SB = -0.39067242E+00
REL. MODAL SOUND POWER SPECTRAL DENSITY= .2657E-02
SUM OF ALL ERRORS IN PSI CALCULATIONS = 0
SUM OF ALL ERRORS IN RS CALCULATIONS= 0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.2576E+02

TABLE 8. (Cont.)


```
M=   -6   N=    1   OMEGA= .1500E+02

M=   -6      N=    1    OMEGA= .1500E+02

UPSTREAM
GAMMA M,N,SB  =  0.58865744E+01
REL. MODAL SOUND POWER SPECTRAL DENSITY=-.3327E-15
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1548E+03

DOWNSTREAM
GAMMA M,N,SB = -0.39067242E+00
REL. MODAL SOUND POWER SPECTRAL DENSITY= .2438E-15
SUM OF ALL ERRORS IN PSI CALCULATIONS =    0
SUM OF ALL ERRORS IN RS CALCULATIONS=    0
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1561E+03

LARGEST PROPAGATING N FOR THIS M =  1
NO MORE PROPAGATING MODES FOR THIS OMEGA
PROBLEM COMPLETED
RELATIVE POWER SPECTRAL DENSITY LEVEL UPSTREAM=-.4321E+02
REL. POWER SPECTRAL DENSITY LEVEL DOWNSTREAM=-.2497E+02
STOP

END OF EXECUTION
CPU TIME: 3:0.55        ELAPSED TIME: 11:11.94
EXIT.
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 9. INPUT/OUTPUT DATA FOR THE ROTOR WAKE TURBULENCE PROGRAM

This program calculates the modal power spectral density for noise generated by the interaction of rotor wake turbulence with the stator vanes. All input and output data are nondimensional. The input variables are defined in order to parallel as closely as possible the variables used in the Mean Wake and Inlet Turbulence programs. The input data consist of geometric, performance, turbulence, and computational parameters.

A. Geometric Data

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
1.	NVANE	V	No. of stator vanes	Integer
2.	NBLADE	B	No. of rotor blades	Integer
3.	SIGMAR	σ_r	Ratio of hub radius to duct radius	Floating Point
4.	SIGMAC	σ_c	Ratio of tip chord of *stator* vane to duct radius	Floating Point
5.			Vector of data for a number, NDAT (an integer variable) of *stator* vane radii	
a.	R	$\tilde{r}$	Nondimensional radius = r/r_{duct}	Floating Point
b.	C	$\tilde{c}$	Nondimensional vane chord = c/c_{tip}	Floating Point
c.	THETA	θ	Vane angle (degrees)	Floating Point

B. Performance Data

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
1.	MT	M_t	Tip Mach number	Floating Point
2.	MA	M	Axial flow Mach number	Floating Point
3.			Frequency range data: Pressure distributions are calculated and stored for the range of frequencies from PSTART to PEND in steps of PSTEP. These data are interpolated for calculations at the frequency of interest OMEGA. See text.	

ORIGINAL PAGE
OF POOR QUALITY

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
a.	PSTART	-	First frequency for pressure calculation. Must be an integer multiple of shaft frequency. (Usually = NBLADE)	Integer
b.	PEND	-	Final frequency for pressure calculation. (Usually = 3x NBLADE)	Integer
c.	PSTEP	-	Frequency step size in pressure calculations. (Usually = NBLADE)	Integer
d.	OMEGA	ω/Ω	Noise frequency of interest (non-dimensionalized on the rotor shaft frequency)	Floating Point
4.	IVOR	-	Switch for rotor wake specification = 0 for free vortex distribution = 1 for specified input velocity distribution	Integer
5.		(for IVOR = 1)	Vector of data for a number, NVELO (an integer variable) of *vane* radii	
	R	$\bar{r}$	a. Nondimensional radius	Floating Point
	XVEL	$\frac{Vyfm}{Vxm}$	b. Ratio of circumferential velocity of mean wake to axial flow velocity (See Fig. 9)	Floating Point
6.	MUV	μ_v	(for IVOR = 0) Ratio of circumferential velocity at the duct radius to the axial flow velocity	Floating Point

C. Turbulence Data

The wake turbulence data are assumed to be constant across the duct.

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
1.	LX	$\bar{L}_x$	Axial turbulence length scale (nondimensionalized on the duct radius)	Floating Point

ORIGINAL PAGE IS
OF POOR QUALITY

	FORTRAN NAME	SYMBOL	VARIABLE	FORTRAN VARIABLE TYPE
2.	LR	$\bar{L}_r$	Radial turbulence length scale (nondimensionalized on the duct radius)	Floating Point
3.	LTHETA	$\bar{L}_\theta$	Circumferential turbulence length scale (nondimensionalized on the duct radius)	Floating Point
4.	EPSW	ϵ_w	RMS turbulence velocity at wake centerline (nondimensionalized on the mean axial velocity), [see Eq. 138 of Vol. 1]	Floating Point
5.	WWIDTH	$\tilde{\delta}$	Ratio of turbulent wake width to duct radius (See Eq. 140 of Vol. 1)	Floating Point

D. Computational Data

The computational input data are identical to those used for the inlet turbulence routine.

Program Output

The output consists of the *relative* power spectral density for all propagating modes at the frequency of interest. The nondimensionalization used here is identical to that used for the inlet turbulence routine.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 10. IMPORTANT INTERNAL PROGRAM VARIABLES FOR THE ROTOR WAKE TURBULENCE PROGRAM

(Used in addition to INPUT/OUTPUT variables)

FORTRAN NAME	SYMBOL	VARIABLE
OMEGA	ω/Ω	Ratio of noise frequency to rotor shaft frequency
FARG		Argument of $\hat{f}$ (argument)
-THETA		Stator vane angle (radius)
QSUM	-	Sum of correlation function contributions multiplying pressure distribution. Value must exceed QTESTR (input variable) before pressure distribution is actually calculated.
MTHETA	-	Circumferential wake Mach number at a given radius
MYM	-	Relative circumferential wake Mach number at a given radius

TABLE 11. SAMPLE EXECUTION OF THE ROTOR WAKE TURBULENCE DATA STORAGE ROUTINE


```
@STATOR
NVANE=
11
NVANE= 11
NBLADE=
15
NBLADE= 15
HUB RADIUS DIVIDED BY DUCT RADIUS =
.484
SIGMAR=  0.48400000E+00
VANE TIP CHORD DIVIDED BY DUCT RADIUS=
.408
SIGMAC=  0.40800000E+00
NDAT =
7
NDAT=  7
VANE GEOMETRY MATRIX INPUT
FEED MATRIX IN ONE ROW AT A TIME, FROM HUB TO TIP
ROW=  1   R/RDUCT =
.484
C/CTIP =
1.
THETA (DEGREES)=
16.3
ROW=  2   R/RDUCT =
.56
C/CTIP =
1.
THETA (DEGREES)=
15.67
ROW=  3   R/RDUCT =
.739
C/CTIP =
1.
THETA (DEGREES)=
14.07
ROW=  4   R/RDUCT =
.92
C/CTIP =
1.
THETA (DEGREES)=
12.57
ROW=  5   R/RDUCT =
.946
C/CTIP =
1.
```

TABLE 11. (Cont.)

ORIGINAL PAGE IS
OF POOR QUALITY

```
THETA (DEGREES)=
12.36
ROW=  6   R/RDUCT =
.973
C/CTIP =
1.
THETA (DEGREES)=
12.15
ROW=  7   R/RDUCT =
1.
C/CTIP =
1.
THETA (DEGREES)=
11.92
VANE GEOMETRY MATRIX IS
 0.484E+00 0.100E+01 0.284E+00
 0.560E+00 0.100E+01 0.273E+00
 0.739E+00 0.100E+01 0.246E+00
 0.920E+00 0.100E+01 0.219E+00
 0.946E+00 0.100E+01 0.216E+00
 0.973E+00 0.100E+01 0.212E+00
 0.100E+01 0.100E+01 0.208E+00
TYPE 1 TO INPUT ROTOR WAKE VELOCITY; 0 FOR FREE  VORTEX
1
IVOR=  1
NUMBER OF RADII FOR SPECIFYING MEAN ROTOR FLOW=
8
NVELO=  8
INPUT MATRIX FOR MEAN ROTOR FLOW
FEED MATRIX IN ONE ROW AT A TIME,FROM HUB TO TIP
ROW=  1 R/RDUCT=
.484
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.808
ROW=  2 R/RDUCT=
.533
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.788
ROW=  3 R/RDUCT=
.612
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.784
ROW=  4 R/RDUCT=
.834
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.712
```

TABLE 11. (Cont.)


```
ROW=  5 R/RDUCT=
.918
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.688
ROW=  6 R/RDUCT=
.946
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.686
ROW=  7 R/RDUCT=
.974
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.713
ROW=  8 R/RDUCT=
1.
MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U  AXIALO=
.713
MEAN ROTOR FLOW VELOCITY MATRIX IS
   .4840E+00 .8080E+00
   .5330E+00 .7880E+00
   .6120E+00 .7840E+00
   .8340E+00 .7120E+00
   .9180E+00 .6880E+00
   .9460E+00 .6860E+00
   .9740E+00 .7130E+00
   .1000E+01 .7130E+00
MT =
.508
MT=  0.50800000E+00
MA =
.323
MA =  0.32300000E+00
NUMBER OF RADIAL STATIONS =
5
NRAD=   5
NUMBER OF CHORDWISE STATIONS  (.LE.20) =
8
NCHORD=   8
START FREQUENCY(HARMONIC NO. OF SHAFT )=
15
PSTART=  15
END FREQUENCY(HARMONIC NO. OF SHAFT)=
45
P END=  45
FREQUENCY STEP SIZE (HARMONIC ORDERS)=
15
PSTEP=  15
```

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 11. (Cont.)

```
ENTER TITLE FOR DATA FILE(MAX 72 CHARACTERS)
NASA ROTOR/STATOR 55/15 BLADES,11 VANES/WAKE TURBULENCE STORAGE
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
ERROR CODE FROM INVERSION ROUTINE=  0
```

(Note: Edited for brevity -- Error Code repeated 165 times.)

```
ERROR CODE FROM INVERSION ROUTINE=  0
STOP

END OF EXECUTION
CPU TIME: 21:28.07     ELAPSED TIME: 1:6:29.97
EXIT.
^C
```

TABLE 12. SAMPLE EXECUTION OF THE ROTOR WAKE TURBULENCE NOISE CALCULATION


```
WASRCH
USER-ASSIGNED TITLE OF DATA SET FROM FILE "STAT" IS
NASA ROTOR/STATOR 55/15 BLADES,11 VANES/WAKE TURBULENCE STORAGE
NOISE FREQUENCY/SHAFT FREQUENCY=
15.
OMEGA= .1500E+02
NUMBER OF RADIAL POSITIONS FOR ACOUSTIC COMP.=
7
NRADNU=   7
ACCURACY OF BESSEL FNS=
.001
EB= .1000E-02
ACCURACY OF CONVERGENCE TO ROOT XMN=
.0001
EC= .1000E-03
WAKE WIDTH=
.072
WWIDTH= .7200E-01
TURBULENCE INTENSITY(RMS FLUCTUATING U/U BAR)=
.01
EPSU= .1000E-01
TURBULENCE LENGTH SCALE IN AXIAL DIRECTION=
.5
LX= .5000E+00
TURBULENCE LENGTH SCALE IN RADIAL DIRECTION=
.5
LR= .5000E+00
TURBULENCE LENGTH SCALE IN CIRCUMFERENTIAL DIRECTION=
.5
LTHETA= .5000E+00

M=    0      N=    1    OMEGA= .1500E+02
%FRSAPR Floating underflow      PC= 17374

%FRSAPR Floating underflow      PC= 17374


UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9627E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9577E+02

M=    0      N=    2    OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9546E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1009E+03
```

TABLE 12. (Cont.)

M= 1 N= 1 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8998E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1116E+03

M= -1 N= 1 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1050E+03

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9336E+02

M= 1 N= 2 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8850E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9785E+02

M= -1 N= 2 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1071E+03

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9508E+02

M= 2 N= 1 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8758E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9249E+02

M= -2 N= 1 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1222E+03

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9020E+02

TABLE 12. (Cont.)

M= 2 N= 2 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8410E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8596E+02

M= -2 N= 2 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1090E+03

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9225E+02

M= 3 N= 1 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8632E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8503E+02

M= -3 N= 1 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1081E+03

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8705E+02

M= 3 N= 2 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.7849E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.7829E+02

M= -3 N= 2 OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9663E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9060E+02

TABLE 12. (Cont.)


```
M=    4       N=    1     OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8833E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8132E+02

M=   -4       N=    1     OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.1006E+03

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8528E+02

M=    5       N=    1     OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8864E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8005E+02

M=   -5       N=    1     OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.9098E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8460E+02

M=    6       N=    1     OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8764E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8147E+02

M=   -6       N=    1     OMEGA= .1500E+02

UPSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8544E+02

DOWNSTREAM
REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=-.8534E+02
PROBLEM COMPLETED
RELATIVE POWER SPECTRAL DENSITY LEVEL UPSTREAM=-.7473E+02
REL. POWER SPECTRAL DENSITY LEVEL DOWNSTREAM=-.7218E+02
STOP

END OF EXECUTION
CPU TIME: 1:59.28       ELAPSED TIME: 4:6.50
EXIT.
```

APPENDIX

FORTRAN CODE LISTINGS

TABLE OF CONTENTS

PRECEDING PAGE BLANK NOT FILMED

PAGE II INTENTIONALLY BLANK

TABLE OF CONTENTS *(Cont.)*

PROGRAM MAINV

This is the calling program controlling the computation of mean wake/stator interaction noise.

ORIGINAL PAGE IS
OF POOR QUALITY

```
      PROGRAM MAINV
      DIMENSION IDATA(10),FDATA(72),VGEOM(10,7),VELOCV(10,2)
      READ NT,MA,NUV,LXR
C
C
      WRITE(5,2001)
2001  FORMAT(' NVANE=')
      READ(5,1001) NVANE
1001  FORMAT(G20.8)
      WRITE(5,2101) NVANE
2101  FORMAT( ' NVANE=',I3)
      WRITE(5,2002)
2002  FORMAT(' NBLADE=')
      READ(5,1001) NBLADE
      WRITE(5,2102) NBLADE
2102  FORMAT(' NBLADE=',I3)
      WRITE(5,2003)
2003  FORMAT(' HUB RADIUS DIVIDED BY DUCT RADIUS =')
      READ(5,1001) SIGMAR
      WRITE(5,2103) SIGMAR
2103  FORMAT(' SIGMAR=',E16.8)
      WRITE(5,2004)
2004  FORMAT(' VANE TIP CHORD DIVIDED BY DUCT RADIUS =')
      READ(5,1001) SIGMAC
      WRITE(5,2104) SIGMAC
2104  FORMAT(' SIGMAC=',E16.8)
      WRITE(5,2005)
2005  FORMAT(' LXR =')
      READ(5,1001) LXR
      WRITE(5,2105) LXR
2105  FORMAT(' LXR=',F16.8)
C
C
      WRITE(5,2006)
2006  FORMAT(' NDAT =')
      READ(5,1001) NDAT
      WRITE(5,2106) NDAT
2106  FORMAT(' NDAT=',I3)
      WRITE(5,2007)
2007  FORMAT(' VANE GEOMETRY MATRIX INPUT')
      WRITE(5,2008)
 2008 FORMAT(' FEED MATRIX IN ONE ROW AT A TIME, FROM HUB TO TIP')
C
C
      DO 10 IROW=1,NDAT
      WRITE(5,2009) IROW
2009  FORMAT(' ROW=',I3,'   R/RDUCT =')
      READ(5,1001) VGEOM(IROW,1)
      WRITE(5,2010)
2010  FORMAT(' C/CTIP =')
      READ(5,1001) VGEOM(IROW,2)
      WRITE(5,2011)
```

```
 2011     FORMAT(' ALPHAS (DEGREES)=')
        READ(5,1001) VGEOM(IROW,3)
        VGEOM(IROW,3)=VGEOM(IROW,3)*.0174533
        WRITE(5,2012)
 2012   FORMAT(' YS/CTIP =')
        READ(5,1001) VGEOM(IROW,4)
        WRITE(5,2013)
 2013   FORMAT(' YF/CTIP =')
        READ(5,1001) VGEOM(IROW,5)
        WRITE(5,2014)
 2014   FORMAT(' YS/CTIP =')
        READ(5,1001) VGEOM(IROW,6)
        WRITE(5,2015)
 2015   FORMAT(' XF/CTIP =')
        READ(5,1001) VGEOM(IROW,7)
 10     CONTINUE
 C
 C
        WRITE(5,2020)
 2020   FORMAT(' VANE GEOMETRY MATRIX IS')
        DO 11 IROW=1,NDAT
        WRITE(5,2120) (VGEOM(IROW,ICOLMN),ICOLMN=1,7)
 2120   FORMAT(' ',7E10.3)
 11     CONTINUE
         WRITE(5,5000)
  5000  FORMAT(' TYPE 1 TO INPUT ROTOR FLOW VELOCITY,0 FOR FREE VORTEX')
          READ(5,1001) IVOR
           WRITE(5,5010) IVOR
  5010    FORMAT(' IVOR=',I3)
          IF (IVOR.EQ.0) NVELO=0
          IF (IVOR.EQ.0) GO TO 4005
 C
 C
          WRITE(5,3000)
  3000  FORMAT(' NUMBER OF RADII FOR SPECIFYING MEAN ROTOR FLOW=')
          READ(5,1001) NVELO
          WRITE(5,3002) NVELO
  3002   FORMAT(' NVELO=',I3)
          WRITE(5,3004)
  3004     FORMAT(' INPUT MATRIX FOR MEAN ROTOR FLOW')
           WRITE(5,3006)
  3006  FORMAT(' FEED MATRIX IN ONE ROW AT A TIME,FROM HUB TO TIP')
         DO 4000 IROW=1,NVELO
         WRITE(5,3008) IROW
  3008   FORMAT(' ROW=',I3,' R/RDUCT=')
         READ(5,1001) VELOCV(IROW,1)
         WRITE(5,3012)
  3012   FORMAT(' MEAN CIRCUMFERENTIAL VELOCITY RATIO=')
         READ(5,1001) VELOCV(IROW,2)
  4000   CONTINUE
 C
```

ORIGINAL PAGE IS OF POOR QUALITY

```
C
      WRITE(5,3014)
 3014 FORMAT(' MEAN ROTOR FLOW VELOCITY MATRIX IS')
      DO 4022 IROW=1,NVELP
      WRITE(5,3016) (VELOCV(IROW,ICOLMN),ICOLMN=1,2)
 3016 FORMAT(' ',2E16.4)
 4022 CONTINUE
C
C
 4025 CONTINUE
C
C
      WRITE(5,2030)
 2030 FORMAT(' MT =')
      READ(5,1071) MT
      WRITE(5,2130) MT
 2130 FORMAT(' MT=',E16.8)
      WRITE(5,2031)
 2031 FORMAT(' MA =')
      READ(5,1071) MA
      WRITE(5,2131) MA
 2131 FORMAT(' MA =',F16.8)
      IF(TVOR.EQ.1) GO TO 6002
      WRITE(5,2032)
 2032 FORMAT(' VXP4/VX4 & PRODUCT =')
      READ(5,1071) MUV
      WRITE(5,2132) MUV
 2132 FORMAT(' MUV=',F16.8)
 6002 CONTINUE
      IF(TVOR.EQ.1) MUV=0.
      WRITE(5,2033    )
 2033 FORMAT(' WAKE WIDTH =')
      READ(5,1071) WWIDTH
      WRITE(5,2133) WWIDTH
 2133 FORMAT(' WWIDTH=',E16.8)
      WRITE(5,2034)
 2034 FORMAT(' MAX VEL DEFICIT DIVIDED BY UBAR')
      READ(5,1071) VELDEF
      WRITE(5,2134) VELDEF
 2134 FORMAT(' VELDEF=',E16.8)
C
C
      WRITE(5,2035)
 2035 FORMAT(' NUMBER OF RADIAL STATIONS =')
      READ(5,1071) NRAD
      WRITE(5,2135) NRAD
 2135 FORMAT(' NRAD=',I4)
      WRITE(5,2036)
 2036 FORMAT(' NUMBER OF CHORDWISE STATIONS =')
      READ(5,1071) NCHORD
      WRITE(5,2136) NCHORD
```

```
2136    FORMAT(' NCHORD=',I4)
        WRITE(5,2037)
2037    FORMAT(' ACCURACY OF BESSEL FN =')
        READ(5,1001) EB
        WRITE(5,2137) EB
2137    FORMAT(' EB=',E16.8)
        WRITE(5,2038)
2038    FORMAT(' ACCURACY OF CONVERGENCE TO ROOT XMN =')
        READ(5,1001) EC
        WRITE(5,2138) EC
2138    FORMAT(' EC=',E16.8)
C
C
        IDATA(1)=NVANE
        IDATA(2)=NBLADE
        IDATA(3)=NRAD
        IDATA(4)=NCHORD
        IDATA(5)=NDAT
        IDATA(9)=IVOR
        IDATA(10)=NVELO
C
C
        RDATA(1)=MT
        RDATA(2)=MA
        RDATA(3)=LX0
        RDATA(4)=SIGMAC
        RDATA(5)=SIGMAR
        RDATA(6)=MUV
        RDATA(7)=WWIDTH
        RDATA(8)=VELDEF
        RDATA(9)=EB
        RDATA(10)=EC
C
C
        CALL SRCHV(IDATA,RDATA,VGEOM,VELOCV)
        STOP
        END
```

PROGRAM SRCHV

Program determines which modes propagate for the mean wake/ stator interaction and enables the noise calculation.

```
      SUBROUTINE SPCHV(IDATA,RDATA,VGEOM,VELOCV)
      INTEGER S,P1
      DIMENSION IER(8),IDATA(10),M(100)
      DIMENSION RDATA(20),VGEOM(10,7),POWER(3,2),SPWR(2),RELEF(3,2)
      DIMENSION VELOCV(10,2)
      REAL MT,MA
C
          DATA((POWER(I,J),I=1,3),J=1,2)/6*0.0/
C
      S=4
10    S=S-1
      IF(S.LE.0) GO TO 9000
C
C
      WRITE(5,2007) S
2007  FORMAT('0S =',I2)
C
C
      NVANE=IDATA(1)
      NBLADE=IDATA(2)
      MT=RDATA(1)
      MA=RDATA(2)
      SIGMAF=RDATA(5)
      ZP=RDATA(9)
      ZC=RDATA(10)
C
C
      XMAX=S*NBLADE*MT/(1.-MA**2)**.5
C
C
      WRITE(5,2001) XMAX
2001  FORMAT(' XMN(MAX) FOR PROPAGATION =',E16.8)
C
C
      P1=(S*NBLADE)/NVANE
      MTEST=2*S*NBLADE-(2*P1+1)*NVANE
      IF(MTEST.GT.0) P1=P1+1
C
C
      J=1
      M(1)=S*NBLADE-P1*NVANE
C
C
15    N=0
20    N=N+1
C
C
      MABS=M(J)
      IF(M(J).LT.0) MABS=-MABS
C
C
```

```
C
        CALL ANPT (MABS,N,SIGMAP,EB,EC,XMN,IEB,IEC)
C
C
        IF(XMN.GE.XMAX) GO TO 6003
C
C
        WRITE(5,3000)
3000    FORMAT('0MODE DATA')
        WRITE(5,3001) S,M(J),N,XMN
3001    FORMAT('S =',I3,'    M =',I3,'    N =',I3,'    XMN =',E16.8)
        IF(XMN.EQ.0.) GO TO 1022
        COFFRA=XMAX/XMN
        WRITE(5,4003) COFFRA
 4003   FORMAT(' CUTOFF RATIO FOR MODE=',E12.4)
        GO TO 1025
 1022   CONTINUE
        WRITE(5,4005)
 4005   FORMAT(' PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE')
 1025   CONTINUE
        WRITE(5,3002) IEB
3002    FORMAT(' SUM OF BESSEL FUNCTION ERROR CODES =',I3)
        WRITE(5,3003) IEC
3003    FORMAT(' ERROR CODE FOR CONVERGENCE TO ROOT XMN =',I3)
C
C
        CALL EIGEN (MABS,SIGMAR,XMN,AMN,BMN,EB,IER)
C
C
        WRITE(5,3004) AMN,BMN
3004    FORMAT(' AMN =',E16.8,'    BMN =',E16.8)
        WRITE(5,3005) IER
3005    FORMAT(' ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC =',I3)
C
C
        RDATA(11)=XMN
        RDATA(12)=AMN
        RDATA(13)=BMN
        IDATA(6)=M(J)
        IDATA(7)=N
        IDATA(8)=S
C
C
        CALL FILLOV(IDATA,RDATA,VGEOM,VELOCV,SPWR)
         II=IABS(IDATA(8))
           POWER(II,1)=POWER(II,1)+SPWR(1)
           POWER(II,2)=POWER(II,2)+SPWR(2)
C
C
        IDATA(6)=-IDATA(6)
        IDATA(8)=-IDATA(8)
```

```
C
C
      WRITE(5,8300) IDATA(6),IDATA(7),IDATA(8)
 8300    FORMAT('M=',I5,'   N=',I5,'   S='I5)
        RFLUP=10.*ALOG10(SPWR(1))
        WRITE(5,8400)
 8400   FORMAT('0UPSTREAM ')
        WRITE(5,8600) RFLUP
 8600   FORMAT(' RELATIVE SOUND POWER LEVEL FOR MODE(DB)=',F16.4)
        RFLDN=10.*ALOG10(SPWR(2))
        WRITE(5,8200)
 8200   FORMAT('0DOWNSTREAM ')
        WRITE(5,8600) RFLDN
      II=IABS(IDATA(8))
        POWER(II,1)=POWER(II,1)+SPWR(1)
        POWER(II,2)=POWER(II,2)+SPWR(2)
C
C
      IDATA(6)=-IDATA(6)
      IDATA(8)=-IDATA(8)
C
C
      GO TO 2?
C
C
C
C
6000  IF(M.EQ.1) GO TO 7000
C
C
      NMAX=N-1
      WRITE(5,6001) NMAX
6001  FORMAT('0LARGEST PROPAGATING N FOR THIS M =',I3)
C
C
      IF(J.EQ.1) GO TO 6100
C
C
      IF(M(1).EQ.0) GO TO 6200
C
C
      IF(M(J-1).LT.0) M(J+1)=M(J-1)-NWAVE
C
C
      IF(M(J-1).GT.0) M(J+1)=M(J-1)+NWAVE
      J=J+1
      GO TO 15
C
C
C
```

```
6100    CONTINUE
        IF(M(1).LE.0) M(2)=M(1)+NVANE
        IF(M(1).GT.0) M(2)=M(1)-NVANE
C
C
        J=2
        GO TO 15
C
C
6200    M(J+1)=M(J)+NVANE
        J=J+1
        GO TO 15
C
C
7000    IF(J.EQ.1) GO TO 8000
C
C
        WRITE(5,7001)
7001    FORMAT(' NO MORE PROPAGATING MODES FOR THIS VALUE OF S')
        GO TO 10
8000    CONTINUE
        WRITE(5,8001)
8001    FORMAT(' NO PROPAGATING MODES FOR THIS VALUE OF S')
        GO TO 10
C
C
9000    CONTINUE
        WRITE(5,9001)
9001    FORMAT(' PROBLEM COMPLETED')
          DO 550 II=1,3
            DO 550 JJ=1,2
          IF(POWER(II,JJ).EQ.0.) GO TO 5215
            RELPR(II,JJ)=10.*ALOG10(POWER(II,JJ))
            GO TO 550
 5215       RELPR(II,JJ)=1.E+35
 550        CONTINUE
            DO 600 II=1,3
            IF(RELPR(II,1).EQ.1.E+35) GO TO 5315
            WRITE(5,610) II,RELPR(II,1)
            GO TO 600
 5315       WRITE(5,5610) II
 5610       FORMAT(' S=',I2,' HARMONIC IS UNEXCITED')
 600        CONTINUE
 610      FORMAT(' RELATIVE POWER LEVEL UPSTREAM S=',I2,5X,E16.4)
            DO 620 II=1,3
            IF(RELPR(II,2).EQ.1.E+35) GO TO 5415
            WRITE(5,630) II,RELPR(II,2)
            GO TO 620
 5415       WRITE(5,5610) II
 620        CONTINUE
 630      FORMAT(' RELATIVE POWER LEVEL DOWNSTREAM S=',I2,5X,E16.4)
        RETURN
        END
```

PROGRAM FILLOV

Program computes the noise generated by the mean wake/ stator interaction in a subsonic turbofan.

```
      SUBROUTINE FILLOV(IDATA,RDATA,VGEOM,VELOCV,SPWR)
     DIMENSION VGEOM(10,7),RDATA(20),IDATA(10),XX(6),IERPSI(2),SPWR(2)
      DIMENSION PHASE(40),VELOCV(10,2)
     DIMENSION PSISTO(2*)
     INTEGER SS,SB
      REAL KX,K,KMNS,MR,MT,MA,MUV,NYM,LYZ
      REAL MTHETA,MTHHUB
      COMPLEX I,AP,DELTAP(100),QMNS,FMNS,TMNS,AMNS
      COMPLEX FMN(40)
       COMPLEX QMNSTM(50,50)
      COMPLEX CASCET(20,30)
      DATA ISS/2/
      I=(0.,1.)
      PI=3.1415926
C
C
      NVANE=IDATA(1)
      NBLADE=IDATA(2)
      NRAD=IDATA(3)
      NCHORD=IDATA(4)
      NDAT=IDATA(5)
      MM=IDATA(6)
      NN=IDATA(7)
      SS=IDATA(8)
      IVDP=IDATA(9)
      NVELO=IDATA(10)
C
C
      WRITE(5,8300) MM,NN,SS
8300  FORMAT('M =',I5,'    N =',I5,'    S =',I5)
 7274    FORMAT(' NRAD=',I3,' NCHORD=',I3)
C
C
      MT=RDATA(1)
      MR=RDATA(2)
      LYZ=RDATA(3)
      SIGMAT=RDATA(4)
      SIGMAR=RDATA(5)
      MUV=RDATA(6)
      WWIDTH=RDATA(7)
      VELREF=RDATA(8)
      EB=RDATA(9)
      EC=RDATA(10)
      XMW=RDATA(11)
      AMN=RDATA(12)
      BMN=RDATA(13)
C
C
      SB=SS*NBLADE
C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
      BETASQ=1.-VA**2
      KMNS=(((GRWMT)**2-BETASQ*(KMNS**2)**.5)*ABS(1.*SS)/(1.*SS)
 7016     FORMAT(' KMNS=',E10.4)
C
C
      WBETA=EXP(14.?38879*WWIDTH**2)
      SUM=0.
      ERROR=.1E-5
      INDEX=0
 100  INDEX=INDEX+1
      DSUM=?./(WBETA**(INDEX**2))
      SUM=SUM+DSUM
      IF(DSUM.LT.ERROR) GO TO 110
      GO TO 100
C
C
 110  XSIGN=1.
 111  GMNS=(SP*MA*NT+XSIGN*KMNS)/BETASQ
      PMNS=(0.,0.)
      DELTAR=(1.-SIGMAR)/(NRAD-1)
      IFR=0
      IDAT=?
      IDOT=2
C
C
      SMTP=-1.
      DO 2?? IR=1,NRAD
C
C
      R=SIGMAR+(IR-1.)*(1.-SIGMAR)/(NRAD-1.)
      IF(R.LE.VGEOM(IDAT,1)) GO TO 201
      IDAT=IDAT+1
      IF(IDAT.GT.NDAT) IDAT=NDAT
 201  DELTA=(R-VGEOM(IDAT,1))/(VGEOM(IDAT,1)-VGEOM(IDAT-1,1))
      DO 210 IX=2,7
 210  XX(IX-1)=VGEOM(IDAT,IX)+DELTA*(VGEOM(IDAT,IX)-VGEOM(IDAT-1,IX))
C
C
      B=XX(1)/2.
      ALPHAS=XX(2)
      YS=XX(3)
      YR=XX(4)
      XS=XX(5)
      XR=XX(6)
 7011    FORMAT(' B,ALPHAS,YS,YR,XS,XR OF R IS')
 7010      FORMAT(6E10.4)
C
C
      IF(IVOR.EQ.?) GO TO 4???
C
```

ORIGINAL PAGE IS OF POOR QUALITY

```
C
C  INTERPOLATE TO GET MEAN CIRCUMFERENTIAL MACH NUMBERS AT THIS R
C
         IF(R.LE.VELOCV(IDOT,1)) GO TO 4031
         IDOT=IDOT+1
         IF(IDOT.GT.NVELO) IDOT=NVELO
 4031    DELTA=(R-VELOCV(IDOT,1))/(VELOCV(IDOT,1)-VELOCV(IDOT-1,1))
         XVEL=VELOCV(IDOT,2)+DELTA*(VELOCV(IDOT,2)-VELOCV(IDOT-1,2))
         MTHETA=XVEL*MA
C
C
 4015    FORMAT(' R=',E10.4,' MTHETA=',E10.4)
C
C
         IF(IR.EQ.1) MTHHUB=MTHETA
 4020    CONTINUE
C
C
      MYM=MT*R-MA*MUV/R
      IF(IVOR.EQ.1) MYM=MT*R-MTHETA
      ACL=ATAN2(MYM,MA)
C
C
      COSAS=COS(ALPHAS)
      SINAS=SIN(ALPHAS)
C
C
      PART1=LX2*MUV*(1./SIGMAR**2-1./R**2)
      IF(IVOR.EQ.1) PART1=LX2*(MTHHUB/MA/SIGMAR-MTHETA/MA/R)
      PART2=SIGMAC*(XS+XR)/R
      PART3=SIGMAC*(XS+XR)*(MT/MA-MUV/R**2)
      IF(IVOR.EQ.1) PART3=SIGMAC*(XS+XR)*(MT/MA-MTHETA/MA/R)
C
C
      PHASE1=SS*(PART1-PART2-PART3)
C
C
      PHASE2=SS*SIGMAC*R*(SINAS+COSAS*MYM/MA)/R
C
C
C
C
      W=(VELDEF/STM)*SIN(ACL+ALPHAS)*MBETA**(-SS**2)
      WH=W*CEXP(-I*(PHASE1+PHASE2))
C
C
      KY=-PHASE2
C
C
      MR=MA*(1.+(MUV/R)**2)**.5
      IF(IVOR.EQ.1) MR=(MA**2+MTHETA**2)**.5
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
        K=SP*MT*B*SIGMAC/(1.-MR**2)
C
C
      GAMMA=(2.*SP*PI/NVANE)*(1.+MT*MR*K*SINAS/(1.-MR**2))
C
C
        H1=-2.*PI*K*SINAS/(NVANE*B*SIGMAC)
        H2=2.*PI*P*COSAS/(NVANE*B*SIGMAC)
C
C
C   SKIP CALL AND STORE OF PRES FOR DOWNSTREAM
          IF(XSTGM.EQ.-1.) GO TO 308
C
C     SKIP PRES CALL AFTER NRAD CALLS FOR THIS VALUE OF S(=SS)
          IF(SS.EQ.ISS) GO TO 308

 302        FORMAT('ZINPUT TO SUBROUTINE PRES')

 303        FORMAT(' MM=',2E16.4)

 304     FORMAT(' KX=',E16.4,' K=',E16.4,' GAMMA=',E16.4)
         WRITE(21,306) MR,H1,H2,NCHORD
 306     FORMAT(' MR=',E16.4,' H1=',E16.4,' H2=',E16.4,' NCRD=',I3)
C
         CALL PRES((1.,0.),KX,K,GAMMA,MR,H1,H2,NCHORD,DELTAP,IERR)
C
C
          IF(IR.EQ.NRAD) ISS=SS
         DO 310 ICK=1,NCHORD
         CASCET(IR,ICK)=DELTAP(ICK)
 310     CONTINUE
          WRITE(5,7500) IERR
 7500       FORMAT(' ERROR CODE FROM INVERSION ROUTINE=',I3)

 7025       FORMAT(' OUTPUT FROM PRES:(DELTAP(IQ),IQ=1,NCHORD')
                       (DELTAP(IQ),IQ=1,NCHORD)
C
C
C
C
 308     CONTINUE
         ZMOD=SIGMAC*B*(GAMS*COSAS-MV*SINAS/P)
C
C
         QMAS=CASCET(IR,NCHORD)/2.
         QMKSTM(IR,NCHORD)=QMAS
         SWTC=1.
C
```

```
C
      NFIN=NCHORD-1
      DO 220 IZ=1,NFIN
      WHTC=1.+SWTC/3.
      SWTC=-SWTC
      THETA=(IZ*PI)/NCHORD
C       QMNSTV(IR,IZ)=CASCET(IR,IZ)*CEXP(I*ZMOL*(1.+COS(THETA)))
220   QMNS=QMNS+WHTC*CASCET(IR,IZ)*CEXP(I*ZMOD*(1.+COS(THETA)))
      QMNS=QMNS*W*(1.+(UN/R)**2)*PI/NCHORD
C
 7022   FORMAT(' IR=',I3,' QMNS(R)=',2E12.4,' WH(R)=',2E12.4)
        QMNS=QMNS*V
C
C
C   SKIP PHODE CALL AND STORE FOR DOWNSTREAM
        IF(XSIGN.EQ.-1.) GO TO 7021
C
C
      ARG=XMN*R
C
C
      CALL PHODE (MP,ARG,AMN,BMN,PSI,TB,IERPSI)
      PSISTO(IR)=PSI
 7020   FORMAT(' PSI(R)=',F16.5)
C
C
      IER=IER+IERPSI(1)+IERPSI(2)
C
C
 7021 CONTINUE
      CA=QMNS*SINAS+WM*COSAS/R
      CR=(WM*YS/R-QMNS*XS)*SIGMAC
      FMNS=(QMNS*PSISTO(IR)/KMNS)*CA
      FMN(IR)=FMNS
      PHASE(IR)=CR-PHASE1-PHASE2
200   CONTINUE
      DO 2004 J=1,NRAD
2001  FORMAT(I4,3E14.5)
2004  CONTINUE
      PMNS=0.
      DO 270 N=1,NRAD-1
      GP=.5*(PHASE(N)+PHASE(N+1))
      X=.5*(PHASE(N+1)-PHASE(N))
      IF(ABS(X).LT.0.001) GO TO 268
      W0=SIN(X)/(2.*X)
      W1=(SIN(X)/X-COS(X))/(4.*X)
      GO TO 269
 268  W0=.5*(1.-X**2/6.)
      W1=X*(1.-X**2/10.)/6.
 269  CONTINUE
      PMNS=PMNS+DELTAR*CEXP(I*GP)*((W0-I*W1)*FMN(N)+(W0+I*W1)*FMN(N+1)
     **)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
270     CONTINUE
        AMNS=-PMNS*NVANE*SIGMAC/(PI*(1.-SIGMAR**2))
C
        POWNUM=2.*AMNS*SL*RT*(CABS(PMNS)*BETASQ*NVANE*SIGMAC)**2
        POWDEN=PI*(1.-SIGMAR**2)*((SB*MT)+XSIGN*MA*KMNS)**2
        RELPWR=POWNUM/POWDEN
        RELPRL=10.*ALOG10(RELPWR)
C
CC
C
        IF(XSIGN.LT.0.) GO TO 9000
C
C
        WRITE(5,3000)
3000    FORMAT('0UPSTREAM ')
        WRITE(5,8101) GMNS
8101    FORMAT(' GAMMA M,N,SB  =',E16.8)
        WRITE(5,8102) AMNS
8102    FORMAT(' MODE AMPLITUDE =',2E16.8)
        WRITE(5,8500) IER
8500    FORMAT(' SUM OF ALL ERRORS IN PSI CALCULATIONS =',I5)
       WRITE (5,8600) RELPRL
8600   FORMAT(' RELATIVE SOUND POWER LEVEL FOR MODE(DB)=',E16.4)
       DO 7006 IR=1,NRAD
         R=SIGMAR+(IR-1.)*(1.-SIGMAR)/(NRAD-1.)
 7002    FORMAT(5X,6E12.4)
 7006    CONTINUE
 7200      FORMAT(' END OF UPSTREAM INTEGRATION')
         XSIGN=-1.
        SPWR(1)=RELPWR
          GO TO 111
9000      WRITE(5,8200)
8200    FORMAT('0DOWNSTREAM')
        WRITE(5,8201) GMNS
8201    FORMAT(' GAMMA M,N,SB =',E16.8)
        WRITE(5,8202) AMNS
8202    FORMAT(' MODE AMPLITUDE =',2E16.8)
        WRITE(5,8400) IER
8400    FORMAT(' SUM OF ALL ERRORS IN PSI CALCULATIONS =',I5)
        WRITE(5,8600) RELPRL
        SPWR(2)=RELPWR
 7202        FORMAT(' END OF DOWNSTREAM INTEGRATION')
          RETURN
          END
```

PROGRAM MAINTI

This is a calling program to control the computation of pressure data for the inlet turbulence case.

```
            PROGRAM MAINTI
         DIMENSION IDATA(10),RDATA(20),VGEOM(10,7)
         REAL MT,MA
         INTEGER PSTART,PEND,P,PRANGE,PSTEP
         DATA (IDATA(J),J=1,10)/10*0/
         DATA (RDATA(J),J=1,20)/20*0.0/
C
C
C     ROTOR INFLOW TURBULENCE -- MATRIX STORAGE PROGRAM
C
C     THIS PROGRAM GROUP CALCULATES AND STORES VECTORS OF
C     THE COMPLEX PRESSURE DISTRIBUTION ACROSS THE ROTOR BLADE.
C
C     DATA REQUIRED TO EXECUTE THIS ROUTINE INCLUDE:
C
C        1. BLADE GEOMETRY
C        2. ROTOR OPERATING SPEEDS
C        3. NUMBER OF RADIAL AND CHORDWISE POSITIONS AT WHICH THE
C           PRESSURES ARE CALCULATED
C        4. FREQUENCY RANGE OF INTEREST (EXPRESSED IN TERMS
C           OF HARMONIC ORDER OF SHAFT ROTATION).
C
C     THE RANGE OF CIRCUMFERENTIAL MODE NUMBER M IS 0 TO NBLADE-1.
C
C     NCHORD MUST BE NO GREATER THAN 20.
C
C
1001     FORMAT(G20.8)
         WRITE(5,2002)
2002     FORMAT(' NBLADE=')
         READ(5,1001) NBLADE
         WRITE(5,2102) NBLADE
2102     FORMAT(' NBLADE=',I3)
         WRITE(5,2003)
2003     FORMAT(' HUB RADIUS DIVIDED BY DUCT RADIUS =')
         READ(5,1001) SIGMAR
         WRITE(5,2103) SIGMAR
2103     FORMAT(' SIGMAR=',E16.8)
         WRITE(5,2004)
2004     FORMAT(' BLADE TIP CHORD DIVIDED BY DUCT RADIUS=')
         READ(5,1001) SIGMAC
         WRITE(5,2104) SIGMAC
2104     FORMAT(' SIGMAC=',E16.8)
C
C
         WRITE(5,2006)
2006     FORMAT(' NDAT =')
         READ(5,1001) NDAT
         WRITE(5,2106) NDAT
2106     FORMAT(' NDAT=',I3)
         WRITE(5,2007)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 2007   FORMAT(' BLADE GEOMETRY MATRIX INPUT')
        WRITE(5,2008)
 2008   FORMAT(' FEED MATRIX IN ONE ROW AT A TIME, FROM HUB TO TIP')
C
C
        DO 10 IROW=1,NDAT
        WRITE(5,2009) IROW
2009    FORMAT(' ROW=',I3,'   R/RDUCT =')
        READ(5,1001) VGEOM(IROW,1)
        WRITE(5,2010)
2010    FORMAT(' C/CTIP =')
        READ(5,1001) VGEOM(IROW,2)
        WRITE(5,2011)
 2011     FORMAT(' CHI (DEGREES)=')
        READ(5,1001) VGEOM(IROW,3)
        VGEOM(IROW,3)=VGEOM(IROW,3)*.0174533
10      CONTINUE
C
C
        WRITE(5,2020)
 2020   FORMAT(' BLADE GEOMETRY MATRIX IS')
        DO 11 IROW=1,NDAT
        WRITE(5,2120) (VGEOM(IROW,ICOLMN),ICOLMN=1,3)
2120    FORMAT(' ',7E10.3)
11      CONTINUE
C
C
        WRITE(5,2030)
2030    FORMAT(' MT =')
        READ(5,1001) MT
        WRITE(5,2130) MT
2130    FORMAT(' MT=',E16.8)
        WRITE(5,2031)
2031    FORMAT(' MA =')
        READ(5,1001) MA
        WRITE(5,2131) MA
2131    FORMAT(' MA =',E16.8)
C
C
        WRITE(5,2035)
2035    FORMAT(' NUMBER OF RADIAL STATIONS =')
        READ(5,1001) NRAD
        WRITE(5,2135) NRAD
2135    FORMAT(' NRAD=',I4)
        WRITE(5,2036)
2036    FORMAT(' NUMBER OF CHORDWISE STATIONS =')
        READ(5,1001) NCHORD
        WRITE(5,2136) NCHORD
2136    FORMAT(' NCHORD=',I4)
C
```

```
C
        WRITE(5,100)
 100   FORMAT(' START FREQUENCY(HARMONIC NO. OF SHAFT )=')
       READ(5,1001) PSTART
       WRITE(5,110) PSTART
 110   FORMAT(' PSTART=',I4)
       WRITE(5,120)
 120   FORMAT(' END FREQUENCY(HARMONIC NO. OF SHAFT)=')
       READ(5,1001) PEND
       WRITE(5,130) PEND
 130    FORMAT(' P END=',I4)
C
C
        WRITE(5,140)
 140   FORMAT(' FREQUENCY STEP SIZE (HARMONIC ORDERS)=')
       READ(5,1001) PSTEP
       WRITE(5,150) PSTEP
 150   FORMAT(' PSTEP=',I4)
       PRANGE=(PEND-PSTART)/PSTEP+1
       IDATA(2)=NBLADE
       IDATA(3)=NRAD
       IDATA(4)=NCHORD
       IDATA(5)=NDAT
       IDATA(6)=PSTART
       IDATA(7)=PEND
       IDATA(8)=PSTEP
C
C
       RDATA(1)=MT
       RDATA(2)=MA
       RDATA(4)=SIGMAC
       RDATA(5)=SIGMAR
C
C
      CALL ROTORS(IDATA,RDATA,VGEOM)
       STOP
       END
```

ORIGINAL PAGE IS
OF POOR QUALITY

PROGRAM ROTORS

Program calculates and stores the pressures generated on a turbofan rotor by inlet turbulence.

```
      SUBROUTINE ROTORS(IDATA,RDATA,VGEOM)
C
C
C     ADJUSTABLE DIMENSION STATEMENTS USED TO MINIMIZE STORAGE
C     REQUIREMENTS OF MATRIX CASCET
C
C          DIMENSION OF DELTAP=20
C          DIMENSION OF CASCET=(NCHORD,NRAD,NBLADE,PRANGE)
C
C     NOTE NCHORD MUST BE NO GREATER THAN 20
       DIMENSION DELTAP(20)
      DIMENSION VGEOM(10,7),RDATA(20),IDATA(12),XX(6)
       DIMENSION CASCET(8,5,15,5)
       DIMENSION A(72)
       REAL KB,MP,MA,MT
        COMPLEX I,DELTAP,CASCET
        INTEGER PSTART,PEND,P,PRANGE,PSTEP
       I=(0.,1.)
       PI=3.1415926
C
C
       NBLADE=IDATA(2)
       NRAD=IDATA(3)
       NCHORD=IDATA(4)
       NDAT=IDATA(5)
       PSTART=IDATA(6)
       PEND=IDATA(7)
       PSTEP=IDATA(8)
C
C
C
C
       MT=RDATA(1)
       MA=RDATA(2)
       SIGMAC=RDATA(4)
       SIGMAR=RDATA(5)
C
C
C     INITIALIZE IDAT FOR GEOMETRIC DATA
C
         IDAT=2
C
       WRITE(5,100)
 100  FORMAT(' ENTER TITLE FOR DATA FILE(MAX 72 CHARACTERS)')
      READ(5,1003) A
 1003  FORMAT(72A1)
C
C     START LOOPS TO CALL CASC--BEGIN WITH LOOP ON RADIAL
C     POSITION TO MINIMIZE INTERPOLATION.
C
         DO 310 IR=1,NRAD
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
      R=SIGMAR+(IR-1.)*(1.-SIGMAR)/(NRAD-1.)
      IF(R.LE.VGEOM(IDAT,1)) GO TO 201
      IDAT=IDAT+1
      IF(IDAT.GT.NDAT) IDAT=NDAT
201   DELTA=(R-VGEOM(IDAT,1))/(VGEOM(IDAT,1)-VGEOM(IDAT-1,1))
      DO 210 IX=2,3
210   XX(IX-1)=VGEOM(IDAT,IX)+DELTA*(VGEOM(IDAT,IX)-VGEOM(IDAT-1,IX))
C
C
      B=XX(1)/2.
      CHI=XX(2)
         WRITE(21,7010) B,CHI,R
 7010    FORMAT(' B=',E10.4,' CHI=',E10.4,' R=',E10.4)
C
C
C
C
      COSCHI=COS(CHI)
        SINCHI=SIN(CHI)
C
C     BEGIN LOOPS ON MODE NUMBER M AND FREQUENCY COUNTER P
C
         DO 306 MCOUNT=1,NBLADE
               MM=MCOUNT-1
C
C     IP IS STORAGE COUNTER FOR PARAMETER P IN MATRIX CASCET
         IP=0
C
C
              DO 310 P=PSTART,PEND,PSTEP
              IP=IP+1
C
C
C
        OMEGBC=MT*SIGMAC*B*(P-MM)
C
C
C
C
      MR=(MA**2+(MT*R)**2)**.5
C
C
      KB=MT/MR*B*SIGMAC*(P-MM)
      SIGMA=-2.*PI*MM/NBLADE
C
C
     S=2.*PI*R/NBLADE/SIGMAC/B
C
C
```

```
C
          WRITE(21,300)
 300        FORMAT('1INPUT TO SUBROUTINE CASC')
            WRITE(21,302) OMEGBC
 302        FORMAT(' OMEGBC=',E12.4)
              WRITE(21,304) MR,KB,SIGMA
 304        FORMAT(' MR=',E10.4,' KB=',E10.4,' SIGMA=',E10.4)
            WRITE(21,306) H,CHI,NCHORD
 306        FORMAT(' H=',E10.4,' CHI=',E10.4,' NCHORD=',I3)
C
C
C
C
C
         CALL CASC(OMEGBC,MR,KB,SIGMA,H,CHI,NCHORD,DELTAP,IERP)
C
C       STORE RESULTS OF CASC IN STORAGE MATRIX FOR DISK FILE.
C       MATRIX POSITION DESCRIPTORS ARE(IN ORDER OF APPEARANCE):
C          1. CHORDWISE POSITION(TRAILING TO LEADING EDGES)
C          2. RADIAL POSITION(INNER RADIUS TO OUTER RADIUS)
C          3. CIRCUMFERENTIAL MODE NUMBER + 1 (M+1)
C          4. ORDER OF FREQUENCY COMPUTATIONS(FIRST TO LAST)
C             WITH: IP=1 FOR PSTART
C                   IP=PRANGE FOR PEND
C
C
         DO 310 ICA=1,NCHORD
         CASCET(ICA,IR,NCOUNT,IP)=DELTAP(ICA)
 310     CONTINUE
          WRITE(5,7500) IERP
 7500      FORMAT(' ERROR CODE FROM INVERSION ROUTINE=',I3)
         WRITE(21,7025)
 7025     FORMAT(' OUTPUT FROM CASC:(DELTAP(IQ),IQ=1,NCHORD')
          WRITE(21,7002) (DELTAP(IQ),IQ=1,NCHORD)
 7002     FORMAT(5X,6E12.4)
C
C
C
C
 390     CONTINUE
C
C       STORE MATRIX OF PRESSURE VALUES ON THE DISK FOR LATER
C       PROCESSING BY OTHER PROGRAMS.
C
          CALL OFILE(23,'ROTO')
          WRITE (23) A
          WRITE (23) IDATA
          WRITE (23) RDATA
          WRITE (23) VGEOM
          WRITE (23) CASCET
C
```

```
C
      ENDFILE 23
C
C
      STOP
      END
```

PROGRAM INSRCH

Program determines which modes propagate for the case of inlet turbulence and controls the noise computation.

```
      PROGRAM INSRCH
      DIMENSION CASCET(8,5,15,5)
      DIMENSION IER(8),IDATA(10)
      DIMENSION RDATA(20),VGEOM(10,7),POWER(2),SPWR(2),RELPR(2)
      DIMENSION NPCON(4)
      DIMENSION A(72)
      INTEGER PRANGE,PEND,PSTART,PSTEP
      COMPLEX CASCET
      REAL MT,MA
C
            DATA(POWER(J),J=1,2)/2*0.0/
C
C
C
C   CALLING PROGRAM FOR INTUR3 TO SEARCH OUT AND CALCULATE
C   ALL PROPAGATING MODES AT A GIVEN POSITIVE FREQUENCY.
C
C
C
C
C     OPEN FILE "ROTO" AND READ  INPUT ARRAYS
C     INCLUDING MAIN STORAGE MATRIX "CASCET".
C
C*****NOTE THAT "ROTO" IS A SEQUENTIAL UNFORMATTED BINARY FILE--
C
C     ORDER OF STORAGE=A,IDATA,RDATA,VGEOM,CASCET
C*****
C
      CALL IFILE(23,'ROTO')
           READ (23) A
           READ (23) IDATA
           READ (23) RDATA
           READ (23) VGEOM
           READ (23) CASCET
      ENDFILE 23
C
C
        WRITE(5,5)
 5    FORMAT(' USER-ASSIGNED TITLE OF DATA SET FROM FILE "ROTO" IS')
      WRITE(5,6) A
 6    FORMAT(' ',72A1)
      PSTART=IDATA(5)
      PEND=IDATA(7)
      PSTEP=IDATA(8)
      PRANGE=(PEND-PSTART)/PSTEP+1
      NPCON(1)=PSTART
      NPCON(2)=PEND
      NPCON(3)=PSTEP
      NPCON(4)=PRANGE
C
C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C  <MTHEOBALD>INSRCH.FOR;18    Mon 29-Sep-80 10:11AM    PAGE 1:1

C      ENTER NOISE FREQUENCY OF INTEREST(MUST BE POSITIVE FREQUENCY)
C
 1001    FORMAT(G20.8)
         WRITE(5,110)
 110     FORMAT(' NOISE FREQUENCY/SHAFT FREQUENCY=')
         READ(5,1001) OMEGA
         RDATA(8)=OMEGA
 115     FORMAT(' OMEGA=',E10.4)
         WRITE(5,115) RDATA(8)
C
C      SELECT NO. OF RADIAL POSITIONS TO INTERPOLATE PRESSURES
C      (ASSUMED THAT NRADNU.GE.NRAD)
C
C*****NOTE: NRADNU MUST BE .LE.30 UNLESS DIMENSION STATEMENTS IN
C              "INTURB" ARE TO BE MODIFIED
C*****
         WRITE(5,120)
 120     FORMAT(' NUMBER OF RADIAL POSITIONS FOR ACOUSTIC COMP.=')
         READ(5,1001) NRADNU
         IDATA(1)=NRADNU
         WRITE(5,130) IDATA(1)
 130     FORMAT(' NRADNU=',I4)
          WRITE(5,2037)
 2037    FORMAT(' ACCURACY OF BESSEL FNS=')
         READ(5,1001) EB
         WRITE(5,2137) EB
 2137    FORMAT(' EB=',E10.4)
         WRITE(5,2038)
 2038    FORMAT(' ACCURACY OF CONVERGENCE TO ROOT XMN=')
         READ(5,1001) EC
         WRITE(5,2138) EC
 2138    FORMAT(' EC=',E10.4)
         RDATA(9)=EB
         RDATA(10)=EC
         MT=RDATA(1)
         MA=RDATA(2)
         SIGMAR=RDATA(5)
C
C
         XMAX=OMEGA*MT/(1.-MA**2)**.5
C
C
         WRITE(5,2001) XMAX
2001     FORMAT(' XMN(MAX) FOR PROPAGATION =',E16.8)
C
C
C
C
C
C    START WITH PLANE WAVE MODE (0,1).  COUNT UP IN N, THEN M
C    UNTIL ALL POSSIBLE HIGHER MODES ARE CUT OFF.
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
      J=1
      M=0
C
C
C
C   RESTART N COUNT HERE FOR NEW M
C
15    N=0
C
C   INCREMENT N
C
20    N=N+1
C
C
      MABS=IABS(M)
C
C
C
      CALL AWRT (MABS,N,SIGMAR,EB,EC,XMN,IEB,IEC)
C
C
C
C   IF MODE IS CUT OFF, GO TO 6000
C
      IF(XMN.GE.XMAX) GO TO 6000
C
C
      WRITE(5,3000)
3000  FORMAT('0MODE DATA')
      WRITE(5,3001) OMEGA,M,N,XMN
3001  FORMAT('0OMEGA=',E10.4,'    M=',I3,'   N=',I3,'   XMN=',E10.4)
      IF(XMN.EQ.0.) GO TO 1000
      COFFRA=XMAX/XMN
      WRITE(5,4000) COFFRA
4000  FORMAT(' CUTOFF RATIO FOR MODE=',E10.4)
      GO TO 1005
1000  CONTINUE
      WRITE(5,4005)
4005  FORMAT(' PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE')
1005  CONTINUE
      WRITE(5,3002) IEB
3002  FORMAT(' SUM OF BESSEL FUNCTION ERROR CODES =',I3)
      WRITE(5,3003) IEC
3003  FORMAT(' ERROR CODE FOR CONVERGENCE TO ROOT XMN =',I3)
C
C   NORMALIZE MODE AMPLITUDE
C
      CALL EIGEN (MABS,SIGMAR,XMN,AMN,BMN,EB,IER)
C
C
      WRITE(5,3004) AMN,BMN
```

```
3004    FORMAT(' AMN =',E16.8,'    BMN =',E16.8)
        WRITE(5,3005) IER
3005    FORMAT(' ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC =',8I2)
C
C
        RDATA(11)=XMN
        RDATA(12)=AMN
        RDATA(13)=BMN
        IDATA(6)=M
        IDATA(7)=N
C
C
       CALL INTURB(IDATA,RDATA,VGEOM,SPWR,CASCET,NPCON)
           POWER(1)=POWER(1)+SPWR(1)
           POWER(2)=POWER(2)+SPWR(2)
C
C
C
C    DO NOT SWITCH SIGN ON M IF M=0
C
         IF(M.EQ.0) GO TO 20
C
        IDATA(6)=-IDATA(6)
C
C
        WRITE(5,8300) IDATA(6),IDATA(7),RDATA(8)
 8300      FORMAT('0M=',I5,'   N=',I5,'   OMEGA=',E10.4)
C
C    SWITCH SIGN ON M AND RECALCULATE
C
        CALL INTURB(IDATA,RDATA,VGEOM,SPWR,CASCET,NPCON)
          POWER(1)=POWER(1)+SPWR(1)
          POWER(2)=POWER(2)+SPWR(2)
C
C
        IDATA(6)=-IDATA(6)
C
C
        GO TO 20
C
C
C
C
C
C    IF MODE IS CUTOFF, DECIDE WHICH MODE TO TRY NEXT.
C
6000    IF(N.EQ.1) GO TO 7000
C
C
        NMAX=N-1
        WRITE(5,6001) NMAX
```

```
6001     FORMAT('0LARGEST PROPAGATING N FOR THIS M =',I3)
C
C
         J=J+1
C
C    INCREMENT M
C
         M=M+1
         GO TO 15
C
C
C
C    N=1 TO REACH THIS POINT
 7000    CONTINUE
C
C
         WRITE(5,7001)
 7001    FORMAT(' NO MORE PROPAGATING MODES FOR THIS OMEGA')
 10      CONTINUE
         WRITE(5,9001)
9001     FORMAT(' PROBLEM COMPLETED')
           DO 550 JJ=1,2
           IF(POWER(JJ).EQ.0.) GO TO 5215
             RELPR(JJ)=10.*ALOG10(ABS(POWER(JJ)))
             GO TO 550
 5215        RELPR(JJ)=1.E+35
 550         CONTINUE
             IF(RELPR(1).EQ.1.E+35) GO TO 5315
             WRITE(5,610) RELPR(1)
             GO TO 600
 5315        WRITE(5,5610)
 5610    FORMAT(' FREQUENCY IS CUT OFF')
 600         CONTINUE
 610     FORMAT(' RELATIVE POWER SPECTRAL DENSITY LEVEL UPSTREAM=',E10.4)
             IF(RELPR(2).EQ.1.E+35) GO TO 5415
             WRITE(5,630) RELPR(2)
             GO TO 620
 5415        WRITE(5,5610)
 620         CONTINUE
 630    FORMAT(' REL. POWER SPECTRAL DENSITY LEVEL DOWNSTREAM=',E10.4)
         STOP
         END
```

PROGRAM INTURB

Program computes the noise generated by a turbofan rotor subjected to inlet turbulence.

```
      SUBROUTINE INTURB(IDATA,RDATA,VGEOM,SPWR,CASCET,NPCON)
C
C
C
C     PROGRAM COMPUTES THE SOUND POWER GENERATED BY A TURBOFAN
C     ROTOR SUBJECTED TO INLET TURBULENCE.  THIS IS A SIMPLIFIED
C     VERSION SUITABLE FOR VERY SMALL TURBULENCE LENGTH SCALES
C     IN THE RADIAL DIRECTION
C
C
C     PROGRAM INTERPOLATES IN RADIAL POSITION AND FREQUENCY FROM A
C     STORED MATRIX OF PRESSURE DISTRIBUTIONS (NAME=CASCET)
C
C
       DIMENSION VGEOM(10,7),RDATA(20),IDATA(10),XX(6),IERPSI(2),SPWR(2)
       DIMENSION BJ(20)
       DIMENSION CASCET(8,5,15,5)
       DIMENSION PSISTO(30),NPCON(4),RR(30),RRNU(30)
       INTEGER P,PSTEP,PRANGE,PSTART,PEND
        REAL KMNS,MT,MA,LR,LTHETA,LX
        COMPLEX I,DELTAP,QMNS,CASCET
        COMPLEX Z,RRS,RRRS,QMNST
        COMPLEX Y
C
C*********************************************************************
C     DIMENSION OF PSISTO AND RRNU MUST BE .GE. NRADNU
C     DIMENSION OF RR MUST BE .GE.NRAD
C*********************************************************************
        I=(0.,1.)
        PI=3.1415926
C
C
        NRADNU=IDATA(1)
        NBLADE=IDATA(2)
        NRAD=IDATA(3)
        NCHORD=IDATA(4)
        NDAT=IDATA(5)
        MM=IDATA(6)
        NN=IDATA(7)
          OMEGA=RDATA(8)
        PRANGE=NPCON(4)
        PSTART=NPCON(1)
        PEND=NPCON(2)
        PSTEP=NPCON(3)
        SIGMAR=RDATA(5)
C
C
        WRITE(5,8300) MM,NN,OMEGA
 8300    FORMAT('0M=',I5,'      N=',I5,'    OMEGA=',E10.4)
        WRITE(21,8300) MM,NN,OMEGA
          WRITE(21,700)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 700        FORMAT(' INTURB EXECUTION')
          WRITE(21,7204) NRAD,NCHORD
 7204       FORMAT(' NRAD=',I3,' NCHORD=',I3)
C
C
C     LIST NONDIMENSIONAL RADIAL POSITIONS AVAILABLE IN STORED
C     DATA (NEEDED FOR INTERPOLATION)
C
        DO 5 IR=1,NRAD
        R=SIGMAR+(IR-1.)*(1.-SIGMAR)/(NRAD-1.)
        RR(IR)=R
  5     CONTINUE
C
C
C
C
         WRITE(21,6)
 6          FORMAT(' R VALUES FROM STORAGE ARE')
            WRITE(21,8) RR
 8          FORMAT(7E10.4)
C
C
C
        MT=RDATA(1)
        MA=RDATA(2)
        SIGMAC=RDATA(4)
        EB=RDATA(9)
        EC=RDATA(10)
        XMN=RDATA(11)
        AMN=RDATA(12)
        BMN=RDATA(13)
C
C     INITIALIZE ERROR ACCUMULATOR IN PSI CALCULATIONS
        IER=0
C
C
C
C     LIST NONDIMENSIONAL RADIAL POSITIONS TO BE USED IN INTERPOLATION
C
C
        DO 7 IRNU=1,NRADNU
        R=SIGMAR+(IRNU-1.)*(1.-SIGMAR)/(NRADNU-1.)
        RRNU(IRNU)=R
C
C           COMPUTE AND STORE MODE SHAPE WEIGHTING AT EACH RADIUS
C
             ARG=XMN*R
             CALL RMODE(MN,ARG,AMN,BMN,PSI,EB,IERPSI)
             PSISTO(IRNU)=PSI
             IER=IER+IERPSI(1)+IERPSI(2)
 7      CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
          WRITE(21,9)
 9          FORMAT(' R VALUES FOR INTERPOLATION ARE')
            WRITE(21,8) RRNU
            WRITE(21,11)
 11         FORMAT(' PSISTO VALUES ARE')
            WRITE(21,3) PSIST3
C
C
        BETASQ=1.-MA**2
        KMNS=(((OMEGA*MT)**2-BETASQ*XMN**2)**.5)*ABS(OMEGA)/OMEGA
          WRITE(21,7018) KMNS
 7018        FORMAT(' KMNS=',E10.4)
C
C
C     CHECK TO BE SURE MODE NUMBER M IS IN RANGE OF STORED DATA
C
          IF(MM.LT.0) GO TO 300
C
C     FOR POSITIVE M, WANT 0.LE.M.LE.(NBLADE-1)
C
          IF(0.LE.MM.AND.MM.LE.(NBLADE-1)) GO TO 310
C
C     IF M POSITIVE BUT OUT OF RANGE, SUBTRACT INTEGER*NBLADE
C
          MTRIAL =MM
          NTRIAL=0
 320      CONTINUE
          NTRIAL=NTRIAL+1
          MSUM=MTRIAL-NTRIAL*NBLADE
          IF(0.LE.MSUM.AND.MSUM.LE.(NBLADE-1)) GO TO 330
          GO TO 320
 330      MUSE=MSUM+1
          GO TO 340
C
C     FOR NEGATIVE M, ADD INTEGER*NBLADE TO M
C
 300      MTRIAL=MM
          NTRIAL=0
 350      CONTINUE
          NTRIAL=NTRIAL+1
          MSUM=MTRIAL+NTRIAL*NBLADE
          IF(0.LE.MSUM.AND.MSUM.LE.(NBLADE-1)) GO TO 360
          GO TO 350
 360      MUSE=MSUM+1
          GO TO 340
C
 310      MUSE=MM+1
C
 340      CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
         WRITE(21,341) MM,MUSE
 341     FORMAT(' MM=',I5,' MUSE=',I5)
C
C
C     SEARCH HARMONIC ORDERS FOR FREQUENCY INTERPOLATION
C
C     WANT PMIN.LE.OMEGA.LE.PMAX
C
        IF(MM.LT.0) GO TO 400
        NUPMIN=PSTART

        NUPMAX=PEND
        DO 10 P=PSTART,PEND-PSTEP,PSTEP
        IF (OMEGA.GE.(1.*P)) NUPMIN=P
 10     CONTINUE
        DO 20 P=PEND,PSTART,-PSTEP
        IF (OMEGA.LT.(1.*P)) NUPMAX=P
 20     CONTINUE
        GO TO 490
C
C
 400       CONTINUE
C
C     FOR MODE NUMBER M.LT.0 MUST SHIFT FREQUENCY PARAMETER TO
C     RETRIEVE CASCET DATA.
C          EQUIV.(OMEGA)=OMEGA+ABS(MM)+MSUM
C
         EQOMEG=OMEGA+IABS(MM)+MSUM
         NUPMIN=PSTART
         NUPMAX=PEND
         DO 410 P=PSTART,PEND-PSTEP,PSTEP
         IF(EQOMEG.GE.(1.*P)) NUPMIN=P
 410     CONTINUE
         DO 420 P=PEND,PSTART,-PSTEP
         IF(EQOMEG.LT.(1.*P)) NUPMAX=P
 420     CONTINUE
C
C
 490     CONTINUE
C
C
C
C     CONVERT TO FIND IP PARAMETER OF DATA STORAGE IN CASCET
C
         IP=0
         DO 25 P=PSTART,PEND,PSTEP
         IP=IP+1
         IPP=P
         IF (NUPMIN.EQ.IPP) IPMIN=IP
         IF (NUPMAX.EQ.IPP) IPMAX=IP
 25      CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
           WRITE(21,21) NUPMIN,NUPMAX
 21        FORMAT(' NUPMIN=',I5,' NUPMAX=',I5)
           WRITE(21,22) IPMIN,IPMAX
 22        FORMAT(' IPMIN=',I5,' IPMAX=',I5)
C
C
C     KILL PROGRAM IF DESIRED FREQUENCY IS OUT OF RANGE OF DATA
        PSTARF=FLOAT(PSTART)
        PENDF=FLOAT(PEND)
        OMEGAT=OMEGA
        IF (MM.LT.0) OMEGAT=EROMEG
        IF(PSTARF.LE.OMEGAT.AND.OMEGAT.LE.PENDF) GO TO 343
        WRITE(5,342)
 342    FORMAT(' FREQUENCY OUT OF RANGE OF DATA FILE')
        STOP
 343    CONTINUE
C
C
110     XSIGN=1.
C
C********** RETURN TO LINE (111) TO REPEAT FOR DOWNSTREAM
C           PROPAGATION ******************************************
C
 111    GMNS=(MT*MA*OMEGA+XSIGN*KMNS)/BETASQ
C
C     INITIALIZE VARIABLES
C          ISUMR IS ERROR CODE ACCUMULATOR FOR BESSEL WEIGHTING FUNCTION
C          PMNS IS RESULT OF DOUBLE INTEGRAL
C          SWTR IS WEIGHTING SIGN FUNCTION FOR RADIAL INTEGRATION
C
        PMNS=0.
        DELTAR=(1.-SIGMAR)/(NRADNU-1)
        ISUMR=0
        IDAT=2
C
C
        SWTR=-1.
C
C**********START RADIAL INTEGRATION LOOP HERE **********
C
         DO 901 IRNU=1,NRADNU
C
C
C
C     RADIAL POSITION SEARCH FOR INTERPOLATION COORDINATES
C
C     WANT RR(IRMIN).LE.RRNU(IRNU).LE.RR(IRMAX)
C          ***NOTE: RR(1)=RRNU(1)
C                   RR(NRAD)=RRNU(NRADNU)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
      IF (IRNU.EQ.1) GO TO 50
      IF (IRNU.EQ.NRADNU) GO TO 60
      IRMIN=1
      IRMAX=NRAD
      DO 30 IR=1,NRAD-1,1
         IF (RRNU(IRNU).GE.RR(IR)) IRMIN=IR
   30 CONTINUE
      DO 40 IR=NRAD,1,-1
         IF (RRNU(IRNU).LT.RR(IR)) IRMAX=IR
  40  CONTINUE
      GO TO 70
 50   CONTINUE
           IRMIN=1
           IRMAX=2
      GO TO 70
 60   CONTINUE
           IRMIN=NRAD-1
           IRMAX=NRAD
      GO TO 70
 70   CONTINUE
C
C
          WRITE(21,71) IRMIN,IRMAX
 71       FORMAT(' IRMIN=',I5,' IRMAX=',I5)
            WRITE(21,72) RR(IRMIN),RRNU(IRNU),RR(IRMAX)
 72   FORMAT(' RR(IRMIN=',E10.4,' RRNU(NU=',E10.4,' RR(IRMAX=',E10.4)
C
C
C     BLADE GEOMETRY AT RADIAL STATION OF INTEGRATION
        R=RRNU(IRNU)
      IF(R.LE.VGEOM(IDAT,1)) GO TO 201
      IDAT=IDAT+1
      IF(IDAT.GT.NDAT) IDAT=NDAT
201   DELTA=(R-VGEOM(IDAT,1))/(VGEOM(IDAT,1)-VGEOM(IDAT-1,1))
      DO 210 IX=2,3
210   XX(IX-1)=VGEOM(IDAT,IX)+DELTA*(VGEOM(IDAT,IX)-VGEOM(IDAT-1,IX))
C
C
      B=XX(1)/2.
      CHI=XX(2)
         WRITE(21,7010) B,CHI,R
 7010    FORMAT(' B=',E10.4,' CHI=',E10.4,' R=',E10.4)
C
C
C
C
      COSCHI=COS(CHI)
        SINCHI=SIN(CHI)
C
C
      ZMOD=SIGMAC*B*(GMNS*COSCHI+MM*SINCHI/R)
```

```
C
C
C
C      CHORDWISE INTEGRATION USES BESSEL INTEGRATION METHOD
C
C      ---CALCULATE BESSEL WEIGHTING FUNCTIONS
C   BESJ REQUIRES POSITIVE ARGUMENT
        X=ABS(ZMOD)
        CALL BESJ(X,0,BJ0,EB,IRROR1)
         CALL BESJ(X,NCHORD,BJ(NCHORD),EB,IRROR2)
         CALL BESJ(X,NCHORD-1,BJ(NCHORD-1),EB,IRROR3)
        ISERR=IRROR1+IRROR2+IRROR3
        WRITE(21,5400) ZMOD,ISERR
 5400   FORMAT(' ZMOD=',E10.4,' ISERR=',I5)
        IF(ZMOD.GT.0.) GO TO 5040
C   BJ0 IS EVEN FUNCTION, ADJUST SIGN ON OTHERS
        BJ(NCHORD)=BJ(NCHORD)*(-1.)**NCHORD
         BJ(NCHORD-1)=BJ(NCHORD-1)*(-1.)**(NCHORD-1)
 5040   CONTINUE
C   USE RECURSION RELATION TO COMPUTE AND STORE BESSEL FUNCTIONS
        DO 2080 N=NCHORD-1,2,-1
        BJ(N-1)= -BJ(N+1)+2.*N*BJ(N)/ZMOD
 2080   CONTINUE
C
C
C
C
C      INITIALIZE INTERPOLATION TO GET PRESSURE VALUE FOR THIS
C      FREQUENCY AND RADIAL POSITION FROM STORED DATA
C
C        USE 4 POINT BIVARIATE INTERPOLATION.  SEE ABRAMOWITZ AND
C        STEGUN 25.2.65
C           NOTE QQ AND PP ARE LESS THAN 1
C
C        DETERMINE FRACTIONAL PARTS OF FREQ. AND RADIAL SPACING
C
          PP=(RRNU(IRNU)-RR(IRMIN))/(RR(IRMAX)-RR(IRMIN))
          QQ=(OMEGA-NUPMIN)/(NUPMAX-NUPMIN)
          IF(MM.LT.0) QQ=(EQOMEG-NUPMIN)/(NUPMAX-NUPMIN)
C
C
         WRITE(21,912) PP,QQ
 912    FORMAT(' PP=',E10.4,' QQ=',E10.4)
C
        QMNS=(0.,0.)
C
C ********** BEGIN CHORDWISE INTEGRATION LOOP HERE **********
C
        DO 220 IZ=1,NCHORD
C
C  COMPUTE R SUB S, THE WEIGHTING FUNCTION AT THE CHORDAL
```

```
C  STATION
C     NOTE THAT Z=I**IKOUNT BELOW
C
          RRRS=(0.,0.)
          Z=(1.,0.)
          DO 5000 IKOUNT=0,NCHORD
          BM=1.
          IF(IKOUNT.EQ.0.OR.IKOUNT.EQ.NCHORD) BM=0.5
          ARGU=IKOUNT*IZ*PI/NCHORD
          IF(IKOUNT.GT.0) GO TO 5010
             RRS=BM*Z*BJ0
             GO TO 5020
 5010     CONTINUE
          RRS=BM*Z*BJ(IKOUNT)*COS(ARGU)
 5020     CONTINUE
          Z=Z*I
          RRRS=RRRS+RRS
 5000     CONTINUE
C
          RRRS=RRRS*2.*PI/NCHORD
          BN=1.
          IF(IZ.EQ.NCHORD) BN=0.5
C
C
C     INTERPOLATE PRESSURE VALUES IN FREQUENCY AND RADIUS
C
           MCOUNT=MUSE
           Y=(1.-PP)*(1.-QQ)*CASCET(IZ,IRMIN,MCOUNT,IPMIN)
           Y=Y+PP*(1.-QQ)*CASCET(IZ,IRMAX,MCOUNT,IPMIN)
           Y=Y+QQ*(1.-PP)*CASCET(IZ,IRMIN,MCOUNT,IPMAX)
           Y=Y+PP*QQ*CASCET(IZ,IRMAX,MCOUNT,IPMAX)
C
         WRITE(21,2) CASCET(IZ,IRMIN,MCOUNT,IPMIN)
           WRITE(21,2) CASCET(IZ,IRMAX,MCOUNT,IPMIN)
           WRITE(21,2) CASCET(IZ,IRMIN,MCOUNT,IPMAX)
            WRITE(21,2) CASCET(IZ,IRMAX,MCOUNT,IPMAX)
 2     FORMAT(' CASCET=',2E10.4)
C
C   EVALUATE FINITE CHORDWISE SUM TO APPROX. INTEGRAL
C
           QMNST=Y*RRRS*BN
           QMNS=QMNS+Y*RRRS*BN
           WRITE(21,5030) IZ,QMNST
 5030      FORMAT(' IZ=',I5,' QMNST=',2E10.4)
 220       CONTINUE
C
C**********END OF CHORDWISE INTEGRATION LOOP (220) **********
C
           WRITE(21,7022) IRNU,QMNS
 7022      FORMAT(' IRNU=',I3,' QMNS=',2E10.4)
C
```

```
 8010      CONTINUE
            GO TO 8013
 8012       CONTINUE
              SUMNU=0.
 8013         CONTINUE
            WRITE(21,8015) SUMNU,P,TURM
 8015       FORMAT(' SUMNU=',E10.4,' P=',I5,' TURM=',E10.4)
C
C
C
           FMNS=CB*CC*CD*SUMNU/R
C
C
         WHTR=1.+SWTR/3.
         SWTR=-SWTR
         IF(IR.EQ.1.OR.IR.EQ.NRAD) WHTR=WHTR/2.
C
C
         PMNS=PMNS+WHTR*FMNS*DELTAR*SIGMAC*SIGMAC*2.
          WRITE(21,7016) PMNS
 7016         FORMAT(' PMNS SUM(R)=',E10.4)
 901       CONTINUE
C
C********** END OF RADIAL INTEGRATION LOOP (901) **********
C
C
C
           SMN=((BETAS2*NBLADE)**2)*MA*MT*OMEGA*PMNS/PI/KMNS
           SMN=SMN/(1.-SIGMAR**2)/(OMEGA*MT+XSIGN*MA*KMNS)**2
           SMN=-XSIGN*SMN
C
C
C   MULTIPLY OUTPUT BY 2 (ADDS +3DB) TO ACCOUNT FOR ENERGY
C   IN NEGATIVE FREQUENCY
           SMN=SMN*2.
C
         RELPWR=SMN
           IPWR=1
             IF(RELPWR.EQ.0.) GO TO 915
         RELPRL=10.*ALOG10(ABS(RELPWR))
           GO TO 920
 915         IPWR=0
 920         CONTINUE
C
CC
C
         IF(XSIGN.LT.0.) GO TO 9000
C
C
         WRITE(5,3000)
3000     FORMAT('0UPSTREAM ')
         WRITE(5,8101) GMNS
```

```
8101      FORMAT(' GAMMA M,N,SB  =',E16.8)
          WRITE(5,8102) SMN
 8102     FORMAT(' REL. MODAL SOUND POWER SPECTRAL DENSITY=',E10.4)
          WRITE(5,8500) IER
          WRITE(5,5500) ISUMR
8500      FORMAT(' SUM OF ALL ERRORS IN PSI CALCULATIONS =',I5)
             IF(IPWR.EQ.0) GO TO 970
               WRITE(5,3600) RELPRL
               GO TO 980
 970            CONTINUE
               WRITE(5,952)
 980           CONTINUE
 8600     FURMAT(' REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=',E10.4)
            WRITE(21,7200)
 7200          FORMAT(' END OF UPSTREAM INTEGRATION')
            XSIGN=-1.
          SPWR(1)=RELPWR
             GO TO 111
C
C********** RETURN TO LINE 111 TO COMPUTE DOWNSTREAM PROPAGATION
C*******************************************************************
C
9000         WRITE(5,8200)
8200      FORMAT('0DOWNSTREAM')
          WRITE(5,8201) GMNS
8201      FORMAT(' GAMMA M,N,SB =',E16.8)
          WRITE(5,8202) SMN
 8202     FORMAT(' REL. MODAL SOUND POWER SPECTRAL DENSITY=',E10.4)
          WRITE(5,8400) IER
          WRITE(5,5500) ISUMR
 5500     FORMAT(' SUM OF ALL ERRORS IN RS CALCULATIONS=',I5)
8400      FORMAT(' SUM OF ALL ERRORS IN PSI CALCULATIONS =',I5)
            IF(IPWR.EQ.0) GO TO 950
              WRITE(5,8500) RELPRL
              GO TO 960
 950            CONTINUE
                WRITE(5,952)
 952      FORMAT(' SOUND POWER SPECTRAL DENSITY FOR MODE=0/UNEXCITED')
 960            CONTINUE
          SPWR(2)=RELPWR
            WRITE(21,7202)
 7202          FORMAT(' END OF DOWNSTREAM INTEGRATION')
            RETURN
            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

PROGRAM MAINTW

This is a calling program to control the computation of pressure data for the wake turbulence case.

```
              PROGRAM MAINTW
       DIMENSION IDATA(10),RDATA(20),VGEOM(10,7)
       DIMENSION VELOCV(10,2)
       REAL MT,MA,MUV,LX,LR,LTHETA
       INTEGER PSTART,PEND,P,PRANGE,PSTEP
       DATA (IDATA(J),J=1,10)/10*0/
       DATA (RDATA(J),J=1,20)/20*0.0/
C
C
C     ROTOR WAKE TURBULENCE-STATOR INTERACTION---MATRIX STORAGE
C
C     THIS PROGRAM GROUP CALCULATES AND STORES VECTORS OF
C     THE COMPLEX PRESSURE DISTRIBUTION ACROSS THE STATOR VANE.
C
C     DATA REQUIRED TO EXECUTE THIS ROUTINE INCLUDE:
C
C        1. VANE GEOMETRY
C        2. ROTOR OPERATING SPEEDS
C        3. NUMBER OF RADIAL AND CHORDWISE POSITIONS AT WHICH THE
C           PRESSURES ARE CALCULATED
C        4. FREQUENCY RANGE OF INTEREST (EXPRESSED IN TERMS
C           OF HARMONIC ORDER OF SHAFT ROTATION).
C
C     THE RANGE OF CIRCUMFERENTIAL MODE NUMBER M IS 0 TO NVANE-1.
C
C     NCHORD MUST BE NO GREATER THAN 20.
C
C
C
1001   FORMAT(G20.8)
       WRITE(5,2001)
 2001  FORMAT(' NVANE=')
       READ(5,1001) NVANE
       WRITE(5,2101) NVANE
 2101  FORMAT(' NVANE=',I3)
       WRITE(5,2002)
2002   FORMAT(' NBLADE=')
       READ(5,1001) NBLADE
       WRITE(5,2102) NBLADE
2102   FORMAT(' NBLADE=',I3)
       WRITE(5,2003)
2003   FORMAT(' HUB RADIUS DIVIDED BY DUCT RADIUS =')
       READ(5,1001) SIGMAR
       WRITE(5,2103) SIGMAR
2103   FORMAT(' SIGMAR=',E16.8)
       WRITE(5,2004)
 2004  FORMAT(' VANE TIP CHORD DIVIDED BY DUCT RADIUS=')
       READ(5,1001) SIGMAC
       WRITE(5,2104) SIGMAC
2104   FORMAT(' SIGMAC=',E16.8)
C
C
       WRITE(5,2006)
```

```
2006    FORMAT(' NDAT =')
        READ(5,1001) NDAT
        WRITE(5,2106) NDAT
2106    FORMAT(' NDAT=',I3)
        WRITE(5,2007)
 2007   FORMAT(' VANE GEOMETRY MATRIX INPUT')
        WRITE(5,2008)
 2008   FORMAT(' FEED MATRIX IN ONE ROW AT A TIME, FROM HUB TO TIP')
C
C
        DO 10 IROW=1,NDAT
        WRITE(5,2009) IROW
2009    FORMAT(' ROW=',I3,'    R/RDUCT =')
        READ(5,1001) VGEOM(IROW,1)
        WRITE(5,2010)
2010    FORMAT(' C/CTIP =')
        READ(5,1001) VGEOM(IROW,2)
        WRITE(5,2011)
 2011     FORMAT(' THETA (DEGREES)=')
        READ(5,1001) VGEOM(IROW,3)
        VGEOM(IROW,3)=VGEOM(IROW,3)*.0174533
10      CONTINUE
C
C
        WRITE(5,2020)
 2020   FORMAT(' VANE GEOMETRY MATRIX IS')
        DO 11 IROW=1,NDAT
        WRITE(5,2120) (VGEOM(IROW,ICOLMN),ICOLMN=1,3)
2120    FORMAT(' ',7E10.3)
11      CONTINUE
C
C
C
        WRITE(5,5000)
 5000   FORMAT(' TYPE 1 TO INPUT ROTOR WAKE VELOCITY; 0 FOR FREE
     1 VORTEX')
        READ(5,1001) IVOR
        WRITE(5,5010) IVOR
 5010   FORMAT(' IVOR=',I3)
        IF (IVOR.EQ.0) NVELO=0
        IF (IVOR.EQ.0) GO TO 4005
C
C
        WRITE(5,3000)
 3000   FORMAT(' NUMBER OF RADII FOR SPECIFYING MEAN ROTOR FLOW=')
        READ(5,1001) NVELO
        WRITE(5,3002) NVELO
 3002   FORMAT(' NVELO=',I3)
        WRITE(5,3004)
 3004   FORMAT(' INPUT MATRIX FOR MEAN ROTOR FLOW')
        WRITE(5,3006)
```

```
 3006   FORMAT(' FEED MATRIX IN ONE ROW AT A TIME,FROM HUB TO TIP')
        DO 4000 IROW=1,NVELO
        WRITE(5,3008) IROW
 3008   FORMAT(' ROW=',I3,' R/RDUCT=')
        READ(5,1001) VELOCV(IROW,1)
        WRITE(5,3012)
 3012   FORMAT(' MEAN CIRCUMFERENTIAL VELOCITY RATIO(U CIRCUM/U
     1 AXIAL0=')
        READ(5,1001) VELOCV(IROW,2)
 4000   CONTINUE
C
C
        WRITE(5,3014)
 3014   FORMAT(' MEAN ROTOR FLOW VELOCITY MATRIX IS')
        DO 4002 IROW=1,NVELO
        WRITE(5,3016) (VELOCV(IROW,ICOLMN),ICOLMN=1,2)
 3016   FORMAT('    ',2E10.4)
 4002   CONTINUE
C
C
 4005   CONTINUE
C
C
C
        WRITE(5,2030)
 2030   FORMAT(' MT =')
        READ(5,1001) MT
        WRITE(5,2130) MT
 2130   FORMAT(' MT=',E16.8)
        WRITE(5,2031)
 2031   FORMAT(' MA =')
        READ(5,1001) MA
        WRITE(5,2131) MA
 2131   FORMAT(' MA =',E16.8)
C
C
C
C
        IF(IVOR.EQ.1) GO TO 6000
        WRITE(5,2032)
 2032   FORMAT(' VYFM/VXM @ R=RDUCT=')
        READ(5,1001) MUV
        WRITE(5,2132) MUV
 2132   FORMAT(' MUV=',E10.4)
 6000   CONTINUE
        IF (IVOR.EQ.1) MUV=0.
C
C
C
        WRITE(5,2035)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
2035     FORMAT(' NUMBER OF RADIAL STATIONS =')
         READ(5,1001) NRAD
         WRITE(5,2135) NRAD
2135     FORMAT(' NRAD=',I4)
         WRITE(5,2036)
2036     FORMAT(' NUMBER OF CHORDWISE STATIONS  (.LE.20) =')
         READ(5,1001) NCHORD
         WRITE(5,2136) NCHORD
2136     FORMAT(' NCHORD=',I4)
C
C
           WRITE(5,100)
 100     FORMAT(' START FREQUENCY(HARMONIC NO. OF SHAFT )=')
         READ(5,1001) PSTART
         WRITE(5,110) PSTART
 110     FORMAT(' PSTART=',I4)
         WRITE(5,120)
 120     FORMAT(' END FREQUENCY(HARMONIC NO. OF SHAFT)=')
         READ(5,1001) PEND
         WRITE(5,130) PEND
 130      FORMAT(' P END=',I4)
C
C
           WRITE(5,140)
 140     FORMAT(' FREQUENCY STEP SIZE (HARMONIC ORDERS)=')
         READ(5,1001) PSTEP
         WRITE(5,150) PSTEP
 150     FORMAT(' PSTEP=',I4)
         PRANGE=(PEND-PSTART)/PSTEP+1
         IDATA(1)=NVANE
         IDATA(2)=NBLADE
         IDATA(3)=NRAD
         IDATA(4)=NCHORD
         IDATA(5)=NDAT
         IDATA(6)=PSTART
         IDATA(7)=PEND
         IDATA(8)=PSTEP
         IDATA(9)=IVOR
           IDATA(10)=NVELO
C
C
         RDATA(1)=MT
         RDATA(2)=MA
         RDATA(4)=SIGMAC
         RDATA(5)=SIGMAR
         RDATA(6)=MUV
C
C
        CALL STATOR(IDATA,RDATA,VGEOM,VELOCV)
         STOP
         END
```

ORIGINAL PAGE IS
OF POOR QUALITY

PROGRAM STATOR

Program calculates and stores the pressures generated on a turbofan stator by wake turbulence.

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE STATOR(IDATA,RDATA,VGEOM,VELOCV)
C
C
C
C        DIMENSION OF DELTAP=20
C        DIMENSION OF CASCET=(NCHORD,NRAD,NVANE,PRANGE)
C
C     NOTE NCHORD MUST BE NO GREATER THAN 20
        DIMENSION DELTAP(20)
       DIMENSION VGEOM(10,7),RDATA(20),IDATA(10),XX(6)
        DIMENSION CASCET(8,5,11,3)
        DIMENSION A(72),VELOCV(10,2)
        REAL KB,MR,MA,MT,MYM,MTHETA
         COMPLEX I,DELTAP,CASCET
         INTEGER PSTART,PEND,P,PRANGE,PSTEP
        I=(0.,1.)
        PI=3.1415926
C
C
        NVANE=IDATA(1)
        NBLADE=IDATA(2)
        NRAD=IDATA(3)
        NCHORD=IDATA(4)
        NDAT=IDATA(5)
        PSTART=IDATA(6)
        PEND=IDATA(7)
        PSTEP=IDATA(8)
        IVOR=IDATA(9)
        NVELO=IDATA(10)
C
C
        WRITE(5,160)
 160      FORMAT(' ENTER TITLE FOR DATA FILE(MAX 72 CHARACTERS)')
          READ(5,1003) A
 1003     FORMAT(72A1)
C
C
        MT=RDATA(1)
        MA=RDATA(2)
        SIGMAC=RDATA(4)
        SIGMAR=RDATA(5)
C
        IDAT=2
        IDOT=2
C
C
C     START LOOPS TO CALL CASC--BEGIN WITH LOOP ON RADIAL
C     POSITION TO MINIMIZE INTERPOLATION.
C
          DO 300 IR=1,NRAD
```

```
C
C
            DO 300 P=PSTART,PEND,PSTEP
            IP=IP+1
C
C
C
         OMEGBC=MT*SIGMAC*B*P
C
C
C
C
        MR=(MA**2+MTHETA**2)**.5
C
C
        KB=MT/MR*B*SIGMAC*P
        SIGMA=-2.*PI*MM/NVANE
C
C
       H=2.*PI*R/NVANE/SIGMAC/B
C
       ANGLE= -THETA
C
C
         WRITE(21,300)
 300       FORMAT('0INPUT TO SUBROUTINE CASC')
           WRITE(21,302) OMEGBC
 302       FORMAT(' OMEGBC=',E10.4)
             WRITE(21,304) MR,KB,SIGMA
 304       FORMAT(' MR=',E10.4,' KB=',E10.4,' SIGMA=',E10.4)
           WRITE(21,306) H,ANGLE,NCHORD
 306       FORMAT(' H=',E10.4,' ANGLE=',E10.4,' NCHORD=',I3)
C
C
C
C
C
        CALL CASC(OMEGBC,MR,KB,SIGMA,H,ANGLE,NCHORD,DELTAP,IERP)
C
C      STORE RESULTS OF CASC IN STORAGE MATRIX FOR DISK FILE.
C      MATRIX POSITION DESCRIPTORS ARE(IN ORDER OF APPEARANCE):
C          1. CHORDWISE POSITION(TRAILING TO LEADING EDGES)
C          2. RADIAL POSITION(INNER RADIUS TO OUTER RADIUS)
C          3. CIRCUMFERENTIAL MODE NUMBER + 1 (M+1)
C          4. ORDER OF FREQUENCY COMPUTATIONS(FIRST TO LAST)
C             WITH: IP=1 FOR PSTART
C                   IP=PRANGE FOR PEND
C
C
        DO 310 ICK=1,NCHORD
        CASCET(ICK,IR,MCOUNT,IP)=DELTAP(ICK)
```

```
 310    CONTINUE
         WRITE(5,7500) IERP
 7500      FORMAT(' ERROR CODE FROM INVERSION ROUTINE=',I3)
        WRITE(21,7025)
 7025     FORMAT(' OUTPUT FROM CASC:(DELTAP(IQ),IQ=1,NCHORD')
          WRITE(21,7002) (DELTAP(IQ),IQ=1,NCHORD)
 7002     FORMAT(5X,6E10.4)
C
C
C
C
 300    CONTINUE
C
C      STORE MATRIX OF PRESSURE VALUES ON THE DISK FOR LATER
C      PROCESSING BY OTHER PROGRAMS.
C
          CALL OFILE(23,'STAT')
          WRITE (23) A
          WRITE (23) IDATA
          WRITE (23) RDATA
          WRITE (23) VGEOM
          WRITE (23) VELOCV
          WRITE (23) CASCET
C
C
          ENDFILE 23
C
C
          STOP
          END
```

ORIGINAL PAGE IS
OF POOR QUALITY

PROGRAM WASRCH

Program determines which modes propagate for the case of rotor wake turbulence and controls the noise computation.

PRECEDING PAGE BLANK NOT FILMED

```
C  <MTHEOBALD>WASRCH.FOR;7    Fri 12-Dec-80 1:46PM       PAGE 1

         PROGRAM WASRCH
         DIMENSION CASCET(8,5,11,3)
         DIMENSION IEK(8),IDATA(12)
        DIMENSION RDATA(22),VGEOM(10,7),POWER(2),SPWR(2),RELPR(2)
        DIMENSION NPCON(4),VELOCV(10,2)
        DIMENSION A(72)
        DIMENSION ISUMTO(120)
        COMPLEX CASCET
           REAL LX,LP,LTHETA
        INTEGER PRANGE,PSTART,PSTEP,PEND
        REAL MT,MA
C
              DATA(ISUMTO(J),J=1,120)/120*0/
              DATA(POWER(J),J=1,2)/2*0.0/
C
C
C
C    CALLING PROGRAM FOR DUTORS TO SEARCH OUT AND CALCULATE
C    ALL PROPAGATING MODES AT A GIVEN POSITIVE FREQUENCY.
C
C
C
C
C      OPEN FILE "STAT" AND READ  INPUT ARRAYS
C      INCLUDING MAIN STORAGE MATRIX "CASCET"
C
C*****NOTE THAT "STAT" IS A SEQUENTIAL UNFORMATTED BINARY FILE--
C
C      ORDER OF STORAGE=A,IDATA,RDATA,VGEOM,VELOCV,CASCET
C*****
C
        CALL IFILE(23,'STAT')
              READ (23) A
              READ (23) IDATA
              READ (23) RDATA
              READ (23) VGEOM
              READ (23) VELOCV
              READ (23) CASCET
        ENDFILE 23
C
C
         WRITE(5,5)
 5      FORMAT(' USER-ASSIGNED TITLE OF DATA SET FROM FILE "STAT" IS')
        WRITE(5,6) A
 6      FORMAT(' ',72A1)
C
        PSTART=IDATA(5)
        PEND=IDATA(7)
        PSTEP=IDATA(6)
        PRANGE=(PEND-PSTART)/PSTEP+1
        NPCON(1)=PSTART
        NPCON(2)=PEND
        NPCON(3)=PSTEP
        NPCON(4)=PRANGE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
C     ENTER NOISE FREQUENCY OF INTEREST(MUST BE POSITIVE FREQUENCY)
C
 1001    FORMAT(G24.8)
         WRITE(5,110)
 110     FORMAT(' NOISE FREQUENCY/SHAFT FREQUENCY=')
         READ(5,1001) OMEGA
         RDATA(9)=OMEGA
 115     FORMAT(' OMEGA=',E12.4)
         WRITE(5,115) RDATA(9)
C
C     SELECT NO. OF RADIAL POSITIONS TO INTERPOLATE PRESSURES
C     (ASSUMED THAT NRADNU.GE.NRAD)
C
C*****NOTE: NRADNU MUST BE .LE.30 UNLESS DIMENSION STATEMENTS IN
C              "OUTPRS" ARE TO BE MODIFIED
C*****
         WRITE(5,120)
 120     FORMAT(' NUMBER OF RADIAL POSITIONS FOR ACOUSTIC COMP.=')
         READ(5,1001) NRADNU
         RDATA(18)=FLOAT(NRADNU)
         WRITE(5,130) NRADNU
 130     FORMAT(' NRADNU=',I4)
          WRITE(5,2237)
 2237    FORMAT(' ACCURACY OF BESSEL FNS=')
         READ(5,1001) EB
         WRITE(5,2137) EB
 2137    FORMAT(' EB=',E14.4)
         WRITE(5,2238)
 2238    FORMAT(' ACCURACY OF CONVERGENCE TO ROOT XMN=')
         READ(5,1001) EC
         WRITE(5,2138) EC
 2138    FORMAT(' EC=',E13.4)
C
C     ENTER TURBULENCE CHARACTERISTICS
C
         WRITE(5,2033)
 2033    FORMAT(' WAKE WIDTH=')
         READ(5,1001) WWIDTH
         WRITE(5,2133) WWIDTH
 2133    FORMAT(' WWIDTH=',E10.4)
         WRITE(5,2034)
 2034    FORMAT(' TURBULENCE INTENSITY(RMS FLUCTUATING U/U BAR)=')
         READ(5,1001) EPSW
 2134    FORMAT(' EPSW=',E12.4)
         WRITE(5,2134) EPSW
C
        WRITE(5,6010)
 6010    FORMAT(' TURBULENCE LENGTH SCALE IN AXIAL DIRECTION=')
         READ(5,1001) LX
         WRITE(5,6002) LX
```

```
 6022   FORMAT(' LX=',E14.4)
        WRITE(5,6024)
 6004   FORMAT(' TURBULENCE LENGTH SCALE IN RADIAL DIRECTION=')
        READ(5,1001) LR
        WRITE(5,6025) LR
 6026   FORMAT(' LR=',E14.4)
        WRITE(5,6028)
 6008   FORMAT(' TURBULENCE LENGTH SCALE IN CIRCUMFERENTIAL DIRECTION=')
        READ(5,1001) LTHETA
        WRITE(5,6010) LTHETA
 6010   FORMAT(' LTHETA=',E14.4)
        RDATA(3)=EPSR
        RDATA(7)=WIDTH
        RDATA(15)=LX
        RDATA(16)=LR
        RDATA(17)=LTHETA
        RDATA(9)=EB
        RDATA(10)=EC
        MT=RDATA(1)
        MA=RDATA(2)
        SIGMAR=RDATA(5)
C
C
        XMAX=OMEGAT*MT/(1.-MA**2)**.5
C
C
        WRITE(21,2001) XMAX
2001    FORMAT(' XMN(MAX) FOR PROPAGATION =',E15.8)
C
C
C
C
C
C   START WITH PLANE WAVE MODE (0,1).  COUNT UP IN N, THEN M
C   UNTIL ALL POSSIBLE HIGHER MODES ARE CUT OFF.
C
        J=1
        M=0
C
C
C
C   RESTART N COUNT HERE FOR NEW M
C
15      N=0
C
C   INCREMENT N
C
20      N=N+1
C
C
        MABS=IABS(M)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
C
      CALL ANRT (MASS,N,SIGMAR,EB,EC,XMN,IEB,IEC)
C
C
C
C   IF MODE IS CUT OFF, GO TO 6000
C
      IF(XMN.GE.XMAX) GO TO 6000
C
C
      WRITE(21,3000)
3000  FORMAT('0MODE DATA')
      WRITE(21,3001) OMEGA,M,N,XMN
3001  FORMAT(' OMEGA=',E13.4,'    M=',I3,'   N=',I3,'   XMN=',E10.4)
      IF(XMN.EQ.0.) GO TO 1200
      COFFRA=XMAX/XMN
      WRITE(21,4000) COFFRA
4000  FORMAT(' CUTOFF RATIO FOR MODE=',E10.4)
      GO TO 1205
1200  CONTINUE
      WRITE(21,4005)
4005  FORMAT(' PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE')
1205  CONTINUE
      WRITE(21,3002) IEB
3002  FORMAT(' SUM OF BESSEL FUNCTION ERROR CODES =',I3)
      WRITE(21,3003) IEC
3003  FORMAT(' ERROR CODE FOR CONVERGENCE TO ROOT XMN =',I3)
C
C   NORMALIZE MODE AMPLITUDE
C
      CALL EIGEN (MARS,SIGMAR,XMN,AMN,BMN,EP,IER)
C
C
      WRITE(21,3004) AMN,BMN
3024  FORMAT(' AMN =',E16.8,'    BMN =',E16.8)
      WRITE(21,3005) IER
3005  FORMAT(' ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC =',I12)
C
C
      RDATA(11)=XMN
      RDATA(12)=AMN
      RDATA(13)=BMN
      IDATA(6)=M
      IDATA(7)=N
C
C
     CALL NATURE(IDATA,RDATA,VGEOM,VELOCY,SPWR,CASCET,NPCON,
    1 ISUMTU)
        POWER(1)=POWER(1)+SPWR(1)
        POWER(2)=POWER(2)+SPWR(2)
```

```
C
C
C
C    DO NOT SWITCH SIGN ON M IF M=2
C
           IF(M.EQ.2) GO TO 22
C
          IDATA(5)=-IDATA(5)
C
C
          WRITE(21,8382) IDATA(5),IDATA(7),RDATA(3)
 8382        FORMAT('?M=',I5,'   N=',I5,'   OMEGA=',E16.4)
C
C    SWITCH SIGN ON M AND RECALCULATE
C
          CALL WATOPB(IDATA,RDATA,WGEOM,VELOCV,SPWR,CASCSI,NPCON,
     1     ISOATO)
            POWER(1)=POWER(1)+SPWR(1)
            POWER(2)=POWER(2)+SPWR(2)
C
C
          IDATA(5)=-IDATA(5)
C
C
          GO TO 22
C
C
C
C
C
C    IF MODE IS CUTOFF, DECIDE WHICH MODE TO TRY NEXT.
C
6000      IF(N.EQ.1) GO TO 7000
C
C
          NMAX=N-1
          WRITE(21,6001) NMAX
6001      FORMAT('LARGEST PROPAGATING N FOR THIS M =',I3)
C
C
          J=J+1
C
C    INCREMENT M
C
          M=M+1
          GO TO 15
C
C
C
C    N=1 TO REACH THIS POINT
 7000     CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
          WRITE(21,7001)
 7001     FORMAT(' NO MORE PROPAGATING MODES FOR THIS OMEGA')
 10       CONTINUE
          WRITE(5,9001)
9001      FORMAT(' PROBLEM COMPLETED')
            DO 550 JJ=1,2
            IF(POWER(JJ).EQ.0.) GO TO 5215
              RELPR(JJ)=10.*ALOG10(ABS(POWER(JJ)))
              GO TO 550
 5215         RELPR(JJ)=1.E+35
 550          CONTINUE
              IF(RELPR(1).EQ.1.E+35) GO TO 5315
              WRITE(5,610) RELPR(1)
              GO TO 620
 5315         WRITE(5,5510)
 5510     FORMAT(' FREQUENCY IS CUT OFF')
 620          CONTINUE
 610      FORMAT(' RELATIVE POWER SPECTRAL DENSITY LEVEL UPSTREAM=',E10.4)
              IF(RELPR(2).EQ.1.E+35) GO TO 5415
              WRITE(5,630) RELPR(2)
              GO TO 622
 5415         WRITE(5,5510)
 622          CONTINUE
 630     FORMAT(' REL. POWER SPECTRAL DENSITY LEVEL DOWNSTREAM=',E10.4)
          STOP
          END
```

PROGRAM WATURB

Program computes the sound power generated by a turbofan *stator* subjected to wake turbulence.

PRECEDING PAGE BLANK NOT FILMED

```
C   <MTHEOBALD>NATURR.FOR;10    Fri 12-Dec-80 2:09PM     PAGE 1

      SUBROUTINE NATURR(IDATA,RDATA,VGEOM,VELOCV,SPWR,CASCET,NPCON,
     1   ISUMTO)
C
C
C
C     PROGRAM COMPUTES THE SOUND POWER GENERATED BY A TURBOFAN
C     STATOR SUBJECTED TO INLET TURBULENCE.  THIS IS A SIMPLIFIED
C     VERSION SUITABLE FOR VERY SMALL TURBULENCE LENGTH SCALES
C     IN THE RADIAL DIRECTION
C
C
C     PROGRAM INTERPOLATES IN RADIAL POSITION AND FREQUENCY FROM A
C     STORED MATRIX OF PRESSURE DISTRIBUTIONS (NAME=CASCET)
C
C
       DIMENSION VGEOM(10,7),RDATA(20),IDATA(10),XX(6),IERPSI(2),SPWR(2)
       DIMENSION BJ(20)
       DIMENSION PSISTO(30),NPCON(4),RR(30),RRNU(30)
         DIMENSION VELOCV(10,2)
       DIMENSION CASCET(8,5,11,3)
       DIMENSION ISUMTO(100),SUMTOT(100,30)
       INTEGER P,PSTEP,PRANGE,PSTART,PEND,S1
        REAL KMNS,MR,MT,MA,LR,LTHETA,LX,MYM,MTHETA
        COMPLEX I,QMNS,A1,CASCET
        COMPLEX Z,RRS,RRRS,QMNST
        COMPLEX X
C
C*************************************************************************
C     DIMENSION OF PSISTO AND RRNU MUST BE .GE. NRADNU
C     DIMENSION OF RR MUST BE .GE.NRAD
C*************************************************************************
        I=(0.,1.)
        PI=3.1415926
C
C
        NRADNU=IFIX(RDATA(13))
        NVANE=IDATA(1)
        NBLADE=IDATA(2)
        NRAD=IDATA(3)
        NCHORD=IDATA(4)
        NDAT=IDATA(5)
        MM=IDATA(6)
        NN=IDATA(7)
        IVOR=IDATA(9)
        NVELO=IDATA(10)
          OMEGA=RDATA(8)
        PRANGE=NPCON(4)
        PSTART=NPCON(1)
        PEND=NPCON(2)
        PSTEP=NPCON(3)
        SIGMAR=RDATA(5)
```

```
C
C
      WRITE(5,8300) MM,NN,OMEGA
 8300  FORMAT('/M=',I5,'       N=',I5,'     OMEGA=',E10.4)
 7204    FORMAT(' NRAD=',I3,' NCHORD=',I3)
C
C
C     LIST NONDIMENSIONAL RADIAL POSITIONS AVAILABLE IN STORED
C     DATA (NEEDED FOR INTERPOLATION)
C
       DO 5 IR=1,NRAD
       R=SIGMAR+(IR-1.)*(1.-SIGMAR)/(NRAD-1.)
       RR(IR)=R
 5     CONTINUE
C
C
C
C
 6       FORMAT(' R VALUES FROM STORAGE ARE')
 8       FORMAT(7E10.4)
C
C
C
       MT=RDATA(1)
       MA=RDATA(2)
       EPSW=RDATA(3)
       SIGMAC=RDATA(4)
       MUV=RDATA(6)
       WWIDTH=RDATA(7)
       ENG=RDATA(9)
       EC=RDATA(10)
       XMN=RDATA(11)
       AMN=RDATA(12)
       BMN=RDATA(13)
       LX=RDATA(15)
       LR=RDATA(16)
       LTHETA=RDATA(17)
C
C     INITIALIZE ERROR CODE ACCUMULATOR IN PSI COMPUTATIONS
       IER=0
C
C
C     LIST NONDIMENSIONAL RADIAL POSITIONS TO BE USED IN INTERPOLATION
C
C
       DO 7 IRNU=1,NRADNU
        R=SIGMAR+(IRNU-1.)*(1.-SIGMAR)/(NRADNU-1.)
        RRNU(IRNU)=R
C
C
C        COMPUTE AND STORE MODE SHAPE WEIGHTING AT EACH RADIUS
```

```
C
              ARG=XMN*R
              CALL RMODE(MM,ARG,AMN,BMN,PSI,EB,IERPSI)
              PSISTO(IRNU)=PSI
              IER=IER+IERPSI(1)+IERPSI(2)
 7         CONTINUE
C
C
 9            FORMAT(' R VALUES FOR INTERPOLATION ARE')
 11           FORMAT(' PSISTO VALUES ARE')
C
C     PERFORM SUMMATION OVER TRANSFORMS OF AUTOCORRELATION
C     FUNCTIONS FOR LATER USE.  THIS SUMMATION IS A
C     FUNCTION OF MODE NUMBER M AND RADIUS R FOR A GIVEN SET
C     OF INPUT PARAMETERS.
C
C     CHECK TO SEE WHETHER SUM IS AVAILABLE FROM STORAGE
C
        MINDEX=MM+50
        IF(ISUMTO(MINDEX).EQ.1) GO TO 6040
C
C     MAKE SURE STORAGE LIMITS ARE NOT EXCEEDED
C
        IF(MM.GT.49) WRITE(5,6010)
 6010   FORMAT(' MODE NUMBER M EXCEEDS STORAGE OF MATRIX SUMTOT')
        IF(MM.GT.49) STOP
C
C     COMPUTE SUMMATION AS A FUNCTION OF R AND STORE IN SUMTOT
C
        DO 6020 IRNU=1,NRADNU
        R=RRNU(IRNU)
        SUMTOT(MINDEX,IRNU)=0.
C
C     LIMITS ON DOUBLE SUM ADAPT TO PHYSICAL PARAMETERS
C
        MM1MAX=INT(23.*R/NBLADE/WIDTH)
        MU1MAX=INT(1./NVANE*(16.*R/LTHETA+MM1MAX*NBLADE+20.))
        DO 6002 MM1=-MM1MAX,MM1MAX
        FARG=MM1*NBLADE*WIDTH/R
        PHIARG=MT*LX/MA*(OMEGA+MM1*NBLADE)
        TERM1=PHIXHT(PHIARG)*(ABS(FHAT(FARG)))**2
        DO 6003 MU1=-MU1MAX,MU1MAX
        THEARG=((MU1*NVANE)-(MM1*NBLADE)-MM)*LTHETA/R
        TURM1=TERM1*PHITHH(THEARG)
        SUMTOT(MINDEX,IRNU)=SUMTOT(MINDEX,IRNU)+TURM1
 6000   CONTINUE
        WRITE(21,8015) MM,OMEGA,R,SUMTOT(MINDEX,IRNU)
 8015 FORMAT(' MM=',I5,' OMEGA=',E10.4,' R=',E10.4,' SUMTOT='E10.4)
C
C
 6020   CONTINUE
        ISUMTO(MINDEX)=1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
 6040    CONTINUE
C
C
         BETASQ=1.-MA**2
         KMNS=(((OMEGA*MT)**2-BETASQ*XMN**2)**.5)*ABS(OMEGA)/OMEGA
           WRITE(21,7018) KMNS
 7018         FORMAT(' KMNS=',E10.4)
C
C
C      CHECK TO BE SURE MODE NUMBER M IS IN RANGE OF STORED DATA
C
           IF(MM.LT.0) GO TO 300
C
C      FOR POSITIVE M, WANT 0.LE.M.LE.(NVANE-1)
C
           IF(0.LE.MM.AND.MM.LE.(NVANE-1)) GO TO 310
C
C      IF M POSITIVE BUT OUT OF RANGE, SUBTRACT INTEGER*NVANE
C
           MTRIAL =MM
           NTRIAL=0
 320       CONTINUE
           NTRIAL=NTRIAL+1
           MSUM=MTRIAL-NTRIAL*NVANE
           IF(0.LE.MSUM.AND.MSUM.LE.(NVANE-1)) GO TO 330
           GO TO 320
 330       MUSE=MSUM+1
           GO TO 340
C
C      FOR NEGATIVE M, ADD INTEGER*NVANE TO M
C
 300       MTRIAL=MM
           NTRIAL=0
 350       CONTINUE
           NTRIAL=NTRIAL+1
           MSUM=MTRIAL+NTRIAL*NVANE
           IF(0.LE.MSUM.AND.MSUM.LE.(NVANE-1)) GO TO 360
           GO TO 350
 360       MUSE=MSUM+1
           GO TO 340
C
 310       MUSE=MM+1
C
 340       CONTINUE
C
 341       FORMAT(' MM=',I5,' MUSE=',I5)
C
C      KILL PROGRAM IF DESIRED FREQUENCY IS OUT OF RANGE
C      OF STORED DATA
```

```
C
          PSTARF=FLOAT(PSTART)
          PENDF=FLOAT(PEND)
          IF(PSTARF.LE.OMEGA.AND.OMEGA.LE.PENDF) GO TO 343
          WRITE(5,342)
 342      FORMAT(' FREQUENCY OUT OF RANGE OF DATA')
          STOP
 343      CONTINUE
C
C
C      SEARCH HARMONIC ORDERS FOR FREQUENCY INTERPOLATION
C
C      WANT PMIN.LE.OMEGA.LE.PMAX
C
         NUPMIN=PSTART

         NUPMAX=PEND
         DO 10 P=PSTART,PEND-PSTEP,PSTEP
         IF (OMEGA.GE.(1.*P)) NUPMIN=P
 10      CONTINUE
         DO 20 P=PEND,PSTART,-PSTEP
         IF (OMEGA.LT.(1.*P)) NUPMAX=P
 20      CONTINUE
C
C
C
C
C
C      CONVERT TO FIND IP PARAMETER OF DATA STORAGE IN CASCET
C
          IP=0
          DO 25 P=PSTART,PEND,PSTEP
          IP=IP+1
          IPP=P
          IF (NUPMIN.EQ.IPP) IPMIN=IP
          IF (NUPMAX.EQ.IPP) IPMAX=IP
 25       CONTINUE
C
 21       FORMAT(' NUPMIN=',I5,' NUPMAX=',I5)
 22       FORMAT(' IPMIN=',I5,' IPMAX=',I5)
C
C
C
110      XSIGN=1.
C
C********** RETURN TO LINE (111) TO REPEAT FOR DOWNSTREAM
C           PROPAGATION *******************************************
C
 111     GMNS=(NT*MA*OMEGA+XSIGN*KMNS)/BETASQ
C
C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     INITIALIZE VARIABLES
C          ISUMR IS ERROR CODE ACCUMULATOR FOR BESSEL WEIGHTING FUNCTION
                                              **S
C
C          PMNS IS RESULT OF DOUBLE INTEGRAL
C
C          S*TR IS WEIGHTING SIGN FUNCTION IN RADIAL INTEGRATION
C
        PMNS=0.
        DELTAR=(1.-SIGMAR)/(NRADNU-1)
        ISUMR=0
        IDOT=2
        IDAT=2
C
C
        S*TR=-1.
C
C**********START RADIAL INTEGRATION LOOP HERE **********
C
         DO 901 IRNU=1,NRADNU
C
C
C
C     RADIAL POSITION SEARCH FOR INTERPOLATION COORDINATES
C
C     WANT RR(IRMIN).LE.RRNU(IRNU).LE.RR(IRMAX)
C           ***NOTE: RR(1)=RRNU(1)
C                    RR(NRAD)=RRNU(NRADNU)
C
        IF (IRNU.EQ.1) GO TO 50
        IF (IRNU.EQ.NRADNU) GO TO 60
        IRMIN=1
        IRMAX=NRAD
        DO 30 IR=1,NRAD-1,1
           IF (RRNU(IRNU).GE.RR(IR)) IRMIN=IR
   30   CONTINUE
        DO 40 IR=NRAD,1,-1
           IF (RRNU(IRNU).LT.RR(IR)) IRMAX=IR
   40   CONTINUE
        GO TO 70
   50   CONTINUE
             IRMIN=1
             IRMAX=2
        GO TO 70
   60   CONTINUE
             IRMIN=NRAD-1
             IRMAX=NRAD
        GO TO 70
   70   CONTINUE
C
C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 71          FORMAT(' IRMIN=',I5,' IRMAX=',I5)
 72     FORMAT(' RR(IRMIN=',E10.4,' RRNU(NU=',E10.4,' RR(IRMAX=',E10.4)
C
C
C      BLADE GEOMETRY AT RADIAL STATION OF INTEGRATION
          R=RRNU(IRNU)
         IF(R.LE.VGEOM(IDAT,1)) GO TO 201
         IDAT=IDAT+1
         IF(IDAT.GT.NDAT) IDAT=NDAT
201      DELTA=(R-VGEOM(IDAT,1))/(VGEOM(IDAT,1)-VGEOM(IDAT-1,1))
         DO 210 IX=2,3
210      XX(IX-1)=VGEOM(IDAT,IX)+DELTA*(VGEOM(IDAT,IX)-VGEOM(IDAT-1,IX))
C
C
         B=XX(1)/2.
         THETA=XX(2)
            WRITE(21,7010) B,THETA,R
 7010       FORMAT(' B=',E10.4,' THETA=',E10.4,' R=',E10.4)
C
C
C
C
         COSTHE=COS(THETA)
         SINTHE=SIN(THETA)
C
C
         IF (IVOR.EQ.0) GO TO 4020
C
C
C      INTERPOLATE TO GET MEAN CIRCUMFERENTIAL MACH NUMBERS AT THIS R
C
C
         IF (R.LE.VELOCV(IDOT,1)) GO TO 4001
         IDOT=IDOT+1
         IF (IDOT.GT.NVELO) IDOT=NVELO
 4001    DELTA=(R-VELOCV(IDOT,1))/(VELOCV(IDOT,1)-VELOCV(IDOT-1,1))
         XVEL=VELOCV(IDOT,2)+DELTA*(VELOCV(IDOT,2)-VELOCV(IDOT-1,2))
         MTHETA=XVEL*MA
C
 4015    FORMAT(' R=',E10.4,' MTHETA=',E10.4)
C
 4020    CONTINUE
C
C
C      MYM IS CIRCUMFERENTIAL MACH NUMBER OF WAKE IMPINGING ON STATOR
C
         MYM=MT*R-MA*MUV/R
         IF (IVOR.EQ.1) MYM=MT*R-MTHETA
         IF(IVOR.EQ.0) MTHETA=MA*MUV/R
C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     CHI IS ARCTAN(CIRCUMFERENTIAL VELOCITY OF WAKE/AXIAL VELOCITY)
        CHI=ATAN2(MYM,MA)
        COSCHI=COS(CHI)
C
C
        ZMOD=SIGMAC*B*(GMNS*COSTHE-MM*SINTHE/R)
C
C
C
C     CHORDWISE INTEGRATION USES BESSEL INTEGRATION METHOD
C
C     ---CALCULATE BESSEL WEIGHTING FUNCTIONS
C   BESJ REQUIRES POSITIVE ARGUMENT
        X=ABS(ZMOD)
        CALL BESJ(X,0,BJ0,EB,IRROR1)
         CALL BESJ(X,NCHORD,BJ(NCHORD),EB,IRROR2)
         CALL BESJ(X,NCHORD-1,BJ(NCHORD-1),EB,IRROR3)
        ISERR=IRROR1+IRROR2+IRROR3
 5400   FORMAT(' ZMOD=',E10.4,' ISERR=',I5)
        IF(ZMOD.GT.0.) GO TO 5040
C   BJ0 IS EVEN FUNCTION, ADJUST SIGN ON OTHERS
        BJ(NCHORD)=BJ(NCHORD)*(-1.)**NCHORD
         BJ(NCHORD-1)=BJ(NCHORD-1)*(-1.)**(NCHORD-1)
 5040   CONTINUE
C   USE RECURSION RELATION TO COMPUTE AND STORE BESSEL FUNCTIONS
        DO 2080 N=NCHORD-1,2,-1
        BJ(N-1)= -BJ(N+1)+2.*N*BJ(N)/ZMOD
 2080   CONTINUE
C
C
C
C
C     INITIALIZE INTERPOLATION TO GET PRESSURE VALUE FOR THIS
C     FREQUENCY AND RADIAL POSITION FROM STORED DATA
C
C       USE 4 POINT BIVARIATE INTERPOLATION.  SEE ABRAMOWITZ AND
C       STEGUN 25.2.65
C          NOTE QQ AND PP ARE LESS THAN 1
C
C       DETERMINE FRACTIONAL PARTS OF FREQ. AND RADIAL SPACING
C
          PP=(RRNU(IRNU)-RR(IRMIN))/(RR(IRMAX)-RR(IRMIN))
          QQ=(OMEGA-NUPMIN)/(NUPMAX-NUPMIN)
C
C
 912    FORMAT(' PP=',E10.4,' QQ=',E10.4)
C
        WMNS=(0.,0.)
C
C ********** BEGIN CHORDWISE INTEGRATION LOOP HERE **********
C
        DO 220 IZ=1,NCHORD
```

```
C
C   COMPUTE R SUB S, THE WEIGHTING FUNCTION AT THE CHORDAL
C   STATION
C      NOTE THAT Z=I**IKOUNT BELOW.
C
          RRRS=(0.,0.)
          Z=(1.,0.)
          DO 5000 IKOUNT=0,NCHORD
          BM=1.
          IF(IKOUNT.EQ.0.OR.IKOUNT.EQ.NCHORD) BM=0.5
          ARGU=IKOUNT*IZ*PI/NCHORD
          IF(IKOUNT.GT.0) GO TO 5010
             RRS=BM*Z*BJ0
             GO TO 5020
 5010     CONTINUE
          RRS=BM*Z*BJ(IKOUNT)*COS(ARGU)
 5020     CONTINUE
          Z=Z*I
          RRRS=RRRS+RRS
 5000     CONTINUE
C
          RRRS=RRRS*2.*PI/NCHORD
          BN=1.
          IF(IZ.EQ.NCHORD) BN=0.5
C
C
C      INTERPOLATE PRESSURE VALUES IN FREQUENCY AND RADIUS
C
           MCOUNT=MUSE
           Y=(1.-PP)*(1.-QQ)*CASCET(IZ,IRMIN,MCOUNT,IPMIN)
           Y=Y+PP*(1.-QQ)*CASCET(IZ,IRMAX,MCOUNT,IPMIN)
           Y=Y+QQ*(1.-PP)*CASCET(IZ,IRMIN,MCOUNT,IPMAX)
           Y=Y+PP*QQ*CASCET(IZ,IRMAX,MCOUNT,IPMAX)
C
 2     FORMAT(' CASCET=',2E10.4)
C
C    EVALUATE FINITE CHORDWISE SUM TO APPROX. INTEGRAL
C
           QMNST=Y*RRRS*BN
           QMNS=QMNS+Y*RRRS*BN
 5030      FORMAT(' IZ=',I5,' QMNST=',2E10.4)
 220       CONTINUE
C
C**********END OF CHORDWISE INTEGRATION LOOP (220) **********
C
 7022      FORMAT(' IRNU=',I3,' QMNS=',2E10.4)
C
C
C
        ISUMR=ISUMR+ISERR
 7021   CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C
      CA=MM*COSTHE/R+GMNS*SINTHE
      CB=(CABS(GMNS)*PSISTO(IRNU)*CA*B)**2
      CD=((MT*R)**2+1.)**.5
C
C
C
C
C
C
C
        FMNS=CB*CD*SUMTOT(MINDEX,IRNU)/(R**2)/COSCHI
C
C
      WHTR=1.+SWTR/3.
      SWTR=-SWTR
      IF(IR.EQ.1.OR.IR.EQ.NRAD) WHTR=WHTR/2.
C
C
      PMNS=PMNS+WHTR*FMNS*DELTAR/R
 7016     FORMAT(' PMNS SUM(R)=',E10.4)
 901    CONTINUE
C
C********** END OF RADIAL INTEGRATION LOOP (901) **********
C
        CC=LR*LTHETA*LX*(EPSW*WWIDTH)**2
C
C
        SMN=((BETASQ*NBLADE*NVANE)**2)*MT*OMEGA*PMNS/KMNS*MA**3
        SMN=SMN/(1.-SIGMAR**2)/(OMEGA*MT+XSIGN*MA*KMNS)**2
        SMN=SMN*CC*SIGMAC*SIGMAC*(-XSIGN)/32./(PI**4)
C
C
C   MULTIPLY OUTPUT BY 2 (ADDS +3DB) TO ACCOUNT FOR ENERGY
C   IN NEGATIVE FREQUENCY
        SMN=SMN*2.
C
      RELPWR=SMN
        IPWR=1
          IF(RELPWR.EQ.0.) GO TO 915
      RELPRL=10.*ALOG10(ABS(RELPWR))
        GO TO 920
 915      IPWR=0
 920      CONTINUE
C
CC
C
      IF(XSIGN.LT.0.) GO TO 9000
C
C
      WRITE(5,3000)
```

```
3000     FORMAT('0UPSTREAM ')
8101     FORMAT(' GAMMA M,N,SB  =',E16.8)
         WRITE(21,8102) SMN
 8102    FORMAT(' REL. MODAL SOUND POWER SPECTRAL DENSITY=',E10.4)
         WRITE(21,8500) IER
         WRITE(21,5500) ISUMR
8500     FORMAT(' SUM OF ALL ERRORS IN PSI CALCULATIONS =',I5)
            IF(IPWR.EQ.0) GO TO 970
              WRITE(5,8600) RELPRL
              GO TO 980
 970           CONTINUE
              WRITE(5,952)
 980          CONTINUE
 8600   FORMAT(' REL. SOUND POWER SPECTRAL DENSITY LEVEL (DB)=',E10.4)
          WRITE(21,7200)
 7200       FORMAT(' END OF UPSTREAM INTEGRATION')
          XSIGN=-1.
         SPWR(1)=RELPWR
          GO TO 111
C
C********** RETURN TO LINE 111 TO COMPUTE DOWNSTREAM PROPAGATION
C*********************************************************************
C
9000       WRITE(5,8200)
8200     FORMAT('0DOWNSTREAM')
8201     FORMAT(' GAMMA M,N,SB =',E16.8)
         WRITE(21,8202) SMN
 8202    FORMAT(' REL. MODAL SOUND POWER SPECTRAL DENSITY=',E10.4)
         WRITE(21,8400) IER
         WRITE(21,5500) ISUMR
 5500    FORMAT(' SUM OF ALL ERRORS IN RS CALCULATIONS=',I5)
8400     FORMAT(' SUM OF ALL ERRORS IN PSI CALCULATIONS =',I5)
           IF(IPWR.EQ.0) GO TO 950
             WRITE(5,8600) RELPRL
             GO TO 960
 950          CONTINUE
              WRITE(5,952)
 952    FORMAT(' SOUND POWER SPECTRAL DENSITY FOR MODE=0/UNEXCITED')
 960          CONTINUE
         SPWR(2)=RELPWR
           WRITE(21,7202)
 7202        FORMAT(' END OF DOWNSTREAM INTEGRATION')
           RETURN
           END
```

ORIGINAL PAGE IS
OF POOR QUALITY

SUBROUTINE ANRT

Program finds the roots of the boundary value equation for an annular duct.

PRECEDING PAGE BLANK NOT FILMED

```
C  <MTHEOBALD>ANRT.FOR;3   Mon 14-Jan-80 6:31PM        PAGE 1

C
C     .................................................................
C
      SUBROUTINE ANRT(M,N,S,EB,EC,XMN,IEB,IEC)
C
C       (AN)NULAR FUNCTION (R)OO(T)
C
C      PURPOSE
C       START WITH A GUESS SOLUTION FOR XM1
C       AND GET A BETTER VALUE FOR XM1.
C       GUESS EACH HIGHER ORDER ROOT FROM
C             XM(N+1) = XMN+3.14159
C       AND THEN REFINE IT TO GET THE (M,N)
C       ROOT TO  :    F=P(JS*Y-YS*J) = 0
C
C     .................................................................
C
      IF(M.EQ.0.AND.N.EQ.1) GO TO 6
        XMN = M-3.14159
        DO 5 NN=1,N
           DX = .31415
           IS = (-1)**(NN+1)
           X  = XMN+3.14159
           J  = 1
C
           CALL ANFU(M,X,S,EB,F,IEB)
           DX = -ISIG(F*IS)*DX
           TE = F
C
  1        IF (ABS(F) .LE.  EC) GO TO 3
           IF (J      .GT. 100) GO TO 2
C
           X  = X+DX
           CALL ANFJ(M,X,S,EB,F,IEB)
C
           IF ((F*TE) .LT. 0.) DX = -DX/2.
           J  = J+1
           TE = F
           GO TO 1
C
  2        IEC = 1
           GO TO 4
  3        IEC = 0
  4        XMN = X
  5      CONTINUE
      RETURN
  6   CONTINUE
      IEB=0
      IEC=0
      XMN=0.
      RETURN
      END
```

ORIGINAL PAGE IS OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C     .............................................................
C
        SUBROUTINE ANFU(M,X,S,EB,F,IEB)
C
C         (AN)NULAR (FU)NCTION
C
C        PURPOSE
C         EVALUATE THE DETERMINANT F = P(JS*Y-YS*J)
C
C     .............................................................
C
          IF (M .NE. 0) GO TO 1
          CALL BESJ(  X,1,ZJ  ,EB ,IE3)
          CALL BESJ(S*X,1,ZJS ,EB ,IE4)
          CALL BESY(  X,1,ZY  ,    IE7)
          CALL BESY(S*X,1,ZYS ,    IE8)
          IEB = IE3+IE4+IE7+IE8
          GO TO 2
C
  1       CALL BESJ(  X,M-1,BJM ,EB ,IE1)
          CALL BESJ(S*X,M-1,BJMS,EB ,IE2)
          CALL BESJ(  X,M+1,BJP ,EB ,IE3)
          CALL BESJ(S*X,M+1,BJPS,EB ,IE4)
          ZJ  = .5*(BJM -BJP )
          ZJS = .5*(BJMS-BJPS)
          CALL BESY(  X,M-1,BYM ,    IE5)
          CALL BESY(S*X,M-1,BYMS,    IE6)
          CALL BESY(  X,M+1,BYP ,    IE7)
          CALL BESY(S*X,M+1,BYPS,    IE8)
          ZY  = .5*(BYM -BYP )
          ZYS = .5*(BYMS-BYPS)
          IEB = IE1+IE2+IE3+IE4+IE5+IE6+IE7+IE8
C
  2       F   = ZJS*ZY - ZJ*ZYS
        RETURN
        END
^L
```

```
C
C     ..................................................................
C
C        SUBROUTINE BESJ
C
C        PURPOSE
C           COMPUTE THE J BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDER
C
C        USAGE
C           CALL BESJ(X,N,BJ,D,IER)
C
C        DESCRIPTION OF PARAMETERS
C           X  -THE ARGUMENT OF THE J BESSEL FUNCTION DESIRED
C           N  -THE ORDER OF THE J BESSEL FUNCTION DESIRED
C           BJ -THE RESULTANT J BESSEL FUNCTION
C           D  -REQUIRED ACCURACY
C           IER-RESULTANT ERROR CODE WHERE
C              IER=0  NO ERROR
C              IER=1  N IS NEGATIVE
C              IER=2  X IS NEGATIVE OR ZERO
C              IER=3  REQUIRED ACCURACY NOT OBTAINED
C              IER=4  RANGE OF N COMPARED TO X NOT CORRECT (SEE REMARKS)
C
C        REMARKS
C           N MUST BE GREATER THAN OR EQUAL TO ZERO, BUT IT MUST BE
C           LESS THAN
C              20+10*X-X** 2/3   FOR X LESS THAN OR EQUAL TO 15
C              90+X/2            FOR X GREATER THAN 15
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           RECURRENCE RELATION TECHNIQUE DESCRIBED BY H. GOLDSTEIN AND
C           R.M. THALER,'RECURRENCE TECHNIQUES FOR THE CALCULATION OF
C           BESSEL FUNCTIONS',M.T.A.C.,V.13,PP.102-108 AND I.A. STEGUN
C           AND M. ABRAMOWITZ,'GENERATION OF BESSEL FUNCTIONS ON HIGH
C           SPEED COMPUTERS',M.T.A.C.,V.11,1957,PP.255-257
C
C     ..................................................................
C
      SUBROUTINE BESJ(X,N,BJ,D,IER)
C
        BJ=.0
        IF (X .NE. 0.) GO TO 9
        IER = 0
        IF (N .EQ. 0) BJ = 1.
       RETURN
```

```
C
    9      IF(N)10,20,20
   10      IER=1
           RETURN
   20      IF(X)30,30,31
   30      IER=2
           RETURN
   31      IF(X-15.)32,32,34
   32      NTEST=20.+10.*X-X** 2/3
           GO TO 36
   34      NTEST=90.+X/2.
   36      IF(N-NTEST)40,38,38
   38      IER=4
           RETURN
   40      IER=0
           N1=N+1
           BPREV=.0
C
C          COMPUTE STARTING VALUE OF M
C
           IF(X-5.)50,50,60
   50      MA=X+6.
           GO TO 70
   60      MA=1.4*X+60./X
   70      MB=N+IFIX(X)/4+2
           MZERO=MAX0(MA,MB)
C
C          SET UPPER LIMIT OF M
C
           MMAX=NTEST
  100      DO 190 M=MZERO,MMAX,3
C
C          SET F(M),F(M-1)
C
           FM1=1.0E-28
           FM=.0
           ALPHA=.0
           IF(M-(M/2)*2)120,110,120
  110      JT=-1
           GO TO 130
  120      JT=1
  130      M2=M-2
           DO 160 K=1,M2
           MK=M-K
           BMK=2.*FLOAT(MK)*FM1/X-FM
           FM=FM1
           FM1=BMK
           IF(MK-N-1)150,140,150
  140      BJ=BMK
  150      JT=-JT
           S=1+JT
```

```
160     ALPHA=ALPHA+BMK*S
        BMK=2.*FM1/X-FM
        IF(N)180,170,180
170     BJ=BMK
180     ALPHA=ALPHA+BMK
        BJ=BJ/ALPHA
        IF(ABS(BJ-BPREV)-ABS(D*BJ))200,200,190
190     BPREV=BJ
        IER=3
200     RETURN
        END
```

```
C
C     ...............................................................
C
C        SUBROUTINE BESY
C
C        PURPOSE
C           COMPUTE THE Y BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDER
C
C        USAGE
C           CALL BESY(X,N,BY,IER)
C
C        DESCRIPTION OF PARAMETERS
C           X  -THE ARGUMENT OF THE Y BESSEL FUNCTION DESIRED
C           N  -THE ORDER OF THE Y BESSEL FUNCTION DESIRED
C           BY -THE RESULTANT Y BESSEL FUNCTION
C           IER-RESULTANT ERROR CODE WHERE
C              IER=0  NO ERROR
C              IER=1  N IS NEGATIVE
C              IER=2  X IS NEGATIVE OR ZERO
C              IER=3  BY HAS EXCEEDED MAGNITUDE OF 10**70
C
C        REMARKS
C           VERY SMALL VALUES OF X MAY CAUSE THE RANGE OF THE LIBRARY
C           FUNCTION ALOG TO BE EXCEEDED
C           X MUST BE GREATER THAN ZERO
C           N MUST BE GREATER THAN OR EQUAL TO ZERO
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE
C           AS DESCRIBED BY A.J.M.HITCHCOCK,'POLYNOMIAL APPROXIMATIONS
C           TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED
C           FUNCTIONS', M.T.A.C., V.11,1957,PP.86-88, AND G.N. WATSON,
C           'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE
C           UNIVERSITY PRESS, 1958, P. 62
C
C     ...............................................................
C
      SUBROUTINE BESY(X,N,BY,IER)
C
C     CHECK FOR ERRORS IN N AND X
C
      IF(N)180,10,10
   10 IER=0
      IF(X)190,190,20
C
C     BRANCH IF X LESS THAN OR EQUAL 4
C
   20 IF(X-4.0)40,40,30
```

```
C
C      COMPUTE Y0 AND Y1 FOR X GREATER THAN 4
C
   30 T1=4.0/X
      T2=T1*T1
      P0=((((-.0000037043*T2+.0000173565)*T2-.0000487613)*T2
     1 +.00017343)*T2-.001753062)*T2+.3989423
      Q0=((((.0000032312*T2-.0000142078)*T2+.0000342468)*T2
     1 -.0000869791)*T2+.0004564324)*T2-.01246694
      P1=((((.0000042414*T2-.0000200920)*T2+.0000580759)*T2
     1 -.000223203)*T2+.002921825)*T2+.3989423
      Q1=((((-.0000035594*T2+.0000151622)*T2-.0000398708)*T2
     1 +.0001064741)*T2-.0005390400)*T2+.03740084
      A=2.0/SQRT(X)
      B=A*T1
      C=X-.7853982
      Y0=A*P0*SIN(C)+B*Q0*COS(C)
      Y1=-A*P1*COS(C)+B*Q1*SIN(C)
      GO TO 90
C
C      COMPUTE Y0 AND Y1 FOR X LESS THAN OR EQUAL TO 4
C
   40 XX=X/2.
      X2=XX*XX
      T=ALOG(XX)+.5772157
      SUM=0.
      TERM=T
      Y0=T
      DO 70 L=1,15
      IF(L-1)50,60,50
   50 SUM=SUM+1./FLOAT(L-1)
   60 FL=L
      TS=T-SUM
      TERM=(TERM*(-X2)/FL**2)*(1.-1./(FL*TS))
   70 Y0=Y0+TERM
      TERM = XX*(T-.5)
      SUM=0.
      Y1=TERM
      DO 80 L=2,16
      SUM=SUM+1./FLOAT(L-1)
      FL=L
      FL1=FL-1.
      TS=T-SUM
      TERM=(TERM*(-X2)/(FL1*FL))*((TS-.5/FL)/(TS+.5/FL1))
   80 Y1=Y1+TERM
      PI2=.6366198
      Y0=PI2*Y0
      Y1=-PI2/X+PI2*Y1
C
C     CHECK IF ONLY Y0 OR Y1 IS DESIRED
C
   90 IF(N-1)100,100,130
```

```
C
C     RETURN EITHER Y0 OR Y1 AS REQUIRED
C
  100 IF(N)110,120,110
  110 BY=Y1
      GO TO 170
  120 BY=Y0
      GO TO 170
C
C    PERFORM RECURRENCE OPERATIONS TO FIND YN(X)
C
  130 YA=Y0
      YB=Y1
      K=1
  140 T=FLOAT(2*K)/X
      YC=T*YB-YA
      IF(ABS(YC)-1.7E33)145,145,141
  141 IER=3
      RETURN
  145 K=K+1
      IF(K-N)150,160,150
  150 YA=YB
      YB=YC
      GO TO 140
  160 BY=YC
  170 RETURN
  180 IER=1
      RETURN
  190 IER=2
      RETURN
      END
^L
```

SUBROUTINE CASC

Program computes the complex pressure distribution across a blade chord for incident turbulence.

PRECEDING PAGE BLANK NOT FILMED

SUBROUTINE EIGEN

Program computes the weighting factors for the eigen functions (mode shapes).

```
C  <MATHEORALD>EIGEN.FOR;2   Mon 14-Jan-80 12:19PM     PAGE 1

        SUBROUTINE EIGEN(MM,SIGMAR,XMN,AMN,BMN,D,IER)
        DIMENSION IER(8)
        IF(XMN.EQ.0.) GO TO 22
        N=MM
        IF(N.LT.0) N=-N
        MMINUS=IABS(N-1)
        MPLUS=N+1
        XMNHUB=XMN*SIGMAR
        S=SIGMAR**2
        CALL BESJ (XMN,N,BJM,D,IER(1))
        CALL BESY (XMN,N,BYM,IER(2))
        CALL BESJ (XMNHUB,N,BJMH,D,IER(3))
        CALL BESY (XMNHUB,N,BYMH,IER(4))
        CALL BESJ (XMN,MPLUS,BJMP,D,IER(5))
        CALL BESY (XMN,MPLUS,BYMP,IER(6))
        CALL BESJ (XMN,MMINUS,BJMM,D,IER(7))
        CALL BESY (XMN,MMINUS,BYMM,IER(8))
        IF(N.GT.0) GO TO 100
        BJMM=-BJMM
        BYMM=-BYMM
100     BJPRIM=(BJMM-BJMP)/2.
        BYPRIM=(BYMM-BYMP)/2.
        IF(ABS(BJPRIM).LT.ABS(BYPRIM)) GO TO 10
        AMN=-BYPRIM/BJPRIM
        BMN=1.
        GO TO 20
10      AMN=1.
        BMN=-BJPRIM/BYPRIM
20      PART1=(1.-(MM/XMN)**2)*(AMN*BJM+BMN*BYM)**2/(1.-S)
        PART2=(S-(MM/XMN)**2)*(AMN*BJMH+BMN*BYMH)**2/(1.-S)
        FACTOR=(PART1-PART2)**.5
        AMN=AMN/FACTOR
        BMN=BMN/FACTOR
        RETURN
22      CONTINUE
        AMN=1.
        BMN=0.
        DO 24 J=1,8
 24     IER(J)=0
        RETURN
        END
```

SUBROUTINE EPSD

Program yields the inlet RMS turbulence intensity.

SUBROUTINE FHAT

Program computes the Fourier transform of the spatial distribution of wake turbulence.

```
C  <MTHEOBALD>FHAT.FOR;1    Wed 13-Feb-80 9:13AM        PAGE 1

            FUNCTION FHAT(FARG)
C
C    FUNCTION YIELDS THE FOURIER TRANSFORM OF THE SPATIAL
C    DISTRIBUTION OF WAKE TURBULENCE VELOCITY FOR
C    SUBROUTINE OUTURB
C
         DATA PI/3.14159/
C
C    NEED SWITCH TO AVOID UNDERFLOW ON EXPONENTIATION :
C    RANGE IS E+-38.
C
         A=(FARG**2)/4./PI
         IF(A.GT.87.) GO TO 10
         B=EXP(-A)
         GO TO 20
 10      B=0.
 20      CONTINUE
         FHAT=B
         RETURN
         END
```

ORIGINAL PAGE IS
OF POOR QUALITY

SUBROUTINE KERNEL

Program computes the kernel
of the Green's function.

```
      SUBROUTINE KERNEL(X,K,GAMMA,MR,H1,H2,ERROR,CKF)
      REAL K,MR
      COMPLEX I,CKF,AN,ANP,ANM,RN,ROOT
      RN(AN)=BETAR**2*H2/(2.*D**2)*(AN**2-K**2)*CEXP(I*(AN+K*MR)
     1 *X)/((AN-GN*H1/D**2)*(AN+K/MR))
      DATA I/(0.,1.)/,PI/3.14159/,XMIN/.001/
      BETAR=SQRT(1.-MR**2)
      IF(ABS(X).GT.XMIN) GO TO 10
      CKF=BETAR/(2.*PI)*CEXP(-I*BETAR**2*K*X/MR)*(1./X-I*K/MR
     1 *ALOG(ABS(X)))
      RETURN
10    CONTINUE
      D=SQRT(H1**2+(BETAR*H2)**2)
      R=D*SQRT(K**2+(D*ALOG(ERROR)/(BETAR*H2*X))**2)
      N1=(GAMMA-R)/(2.*PI)
      N2=(GAMMA+R)/(2.*PI)
      CKF=0.
      IF(X.GT.0.) GO TO 50
      DO 40 N=N1,N2
      GN=GAMMA-2.*PI*N
      S=(K*D)**2-GN**2
      IF(S.GE.0.) ROOT=-SQRT(S)*ABS(K)/K
      IF(S.LT.0.) ROOT=I*SQRT(-S)
      ANM=(GN*H1-BETAR*H2*ROOT)/D**2
      CKF=CKF+RN(ANM)
40    CONTINUE
      RETURN
50    CONTINUE
      DO 70 N=N1,N2
      GN=GAMMA-2.*PI*N
      S=(K*D)**2-GN**2
      IF(S.GE.0.) ROOT=-SQRT(S)*ABS(K)/K
      IF(S.LT.0.) ROOT=I*SQRT(-S)
      ANP=(GN*H1+BETAR*H2*ROOT)/D**2
      CKF=CKF-RN(ANP)
70    CONTINUE
      CKF=CKF-BETAR**2*K/(2.*MR)*SINH(BETAR**2*K
     1 *H2/MR)*CEXP(-I*BETAR**2*K*X/MR)/(COSH(BETAR**2*K*H2/MR)-
     2 COS(GAMMA+K*H1/MR))
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

SUBROUTINE PHITHH

Program computes the Fourier transform of the correlation function in the circumferential direction.

```
C  <MTHFORALD>PHITHM.FOR;3   Tue 22-Jan-80 3:27PM     PAGE 1

         FUNCTION PHITHM(ARGUM2)
C
C
C  PROGRAM CALCULATES THE FOURIER TRANSFORM OF THE CORRELATION
C  FUNCTION IN THE AZIMUTHAL DIRECTION AS A FUNCTION OF
C
C
C  12/14/79 VERSION USES GAUSSIAN AUTOCORRELATION FUNCTION.
C  FORM FOLLOWS EQUATION 27 IN M.D. MARK'S "ROTOR INLET
C  TURBULENCE" PAGE 10.
C
         DATA PI/3.14159/
C
C  NEED SWITCH TO AVOID UNDERFLOW ON EXPONENTIATION.  RANGE
C  IS E+-38.
C
         A=ARGUM2**2
         B=A/PI
         IF(B.GT.87.) GO TO 10
         C=2.*EXP(-B)
         GO TO 20
 10       C=0.
 20      CONTINUE
         PHITHM=C
            RETURN
            END
```

ORIGINAL PAGE IS
OF POOR QUALITY

SUBROUTINE PHIXHT

Program calculates the Fourier transform of the correlation function in the axial direction.

```
C   <MTHRCDALD>PHIXHT.FOR;3   Tue 22-Jan-80 3:19PM      PAGE 1

              FUNCTION PHIXHT(ARGUM1)
C
C   PROGRAM CALCULATES THE FOURIER TRANSFORM OF THE CORRELATION
C   FUNCTION IN THE AXIAL DIRECTION AS A FUNCTION OF
C   NONDIMENSIONAL FREQUENCY
C
C
C   12/14/79 VERSION USES GAUSSIAN AUTOCORRELATION FUNCTION.
C   FORM FOLLOWS EQUATION 27 IN W.D. MARK'S "ROTOR INLET
C   TURBULENCE" PAGE 10.
C
C
C
C   NEED SWITCH TO AVOID UNDERFLOW ON EXPONENTIATION.  RANGE
C   IS E+-38
C
            DATA PI/3.14159/
           A=ARGUM1**2
            B=A/PI
            IF(B.GT.87.) GO TO 10
            C=2.*EXP(-B)
            GO TO 20
 10         C=0.
 20         CONTINUE
            PHIXHT=C
          RETURN
          END
```

ORIGINAL PAGE IS
OF POOR QUALITY

SUBROUTINE PRES

Program generates the complex pressure distribution across the stator vane chord.

```
      SUBROUTINE PRES(WH,XX,K,GAMMA,MR,H1,H2,MSEG,F,IER)
      REAL K,MR,KX
      COMPLEX I,WH,F,A,CKF
      COMPLEX G,G0,G1,G2,G3
      DIMENSION F(100),A(100,100),WA(100),B(100)
      DATA I/(0.,1.)/,ERROR/.0001/,PI/3.14159/
      DATA (B(J),J=1,100)/100*1.0/
      G(X)=(G0+G1*X+G2*X**2+G3*X**3)*CEXP(I*K*MR*X)
      B(MSEG)=.5
      BETAR=SQRT(1.-MR**2)
      G0=I*BETAR*K/(2.*PI*MR)
      G1=BETAR*K**2/(2.*PI)*(1./MR**2-0.5)
      G2=I*BETAR*K**3/(4.*PI)*(1./(2.*MR)-1./MR**3)
      G3=-BETAR*K**4/(12.*PI)*(1./MR**4-1./(2.*MR**2)-0.125)
      DO 10 M=1,MSEG
      X=COS((M-.5)*PI/MSEG)
      F(M)=WH*CEXP(I*KX*X)
10    CONTINUE
      N=MSEG
      DO 50 M=1,N
      DO 50 L=1,N
      S=-ALOG(2.)
      DO 40 IR=1,N
      S=S-2.*B(IR)/IR*COS((M-.5)*IR*PI/N)*COS(PI*IR*L/N)
40    CONTINUE
      ARG=COS((M-.5)*PI/N)-COS(L*PI/N)
      CALL KERNEL(ARG,K,GAMMA,MR,H1,H2,ERROR,CKF)
      A(M,L)=B(L)*(CKF+G(ARG)*(S-ALOG(ABS(ARG))))*PI/N
50    CONTINUE
      CALL LEQT1C(A,MSEG,100,F,1,100,0,WA,IER)
      RETURN
      END
```

SUBROUTINE RMODE

Program computes the radial
mode shape for an annular
duct.

```
      SUBROUTINE RMODE (MM,X,AMN,BMN,PSI,D,IERROR)
      DIMENSION IERROR(2)
      IF(X.EQ.0.) GO TO 300
      FACTOR=1.
      M=MM
      IF(M.LT.0) GO TO 100
200   CONTINUE
      CALL BESJ (X,M,BJ,D,IERROR(1))
      CALL BESY (X,M,BY,IERROR(2))
      PSI=FACTOR*(AMN*BJ+BMN*BY)
      RETURN
100   M=-M
      K=M/2
      IF((1.*M)/(2.).GT.(1.*K)) FACTOR=-1.
      GO TO 200
300   CONTINUE
      PSI=AMN
      IERROR(1)=0
      IERROR(2)=0
      RETURN
      END
```

SUBROUTINE SCALLR

Program yields the turbulence length scale in the radial direction.

```
C  <MTHEODALD>SCALLR.FOR;6    Mon 29-Sep-80 11:11PM     PAGE 1


         FUNCTION SCALLR(R)
C
C  PROGRAM CALCULATES THE INLET TURBULENCE LENGTH
C  SCALE IN THE RADIAL DIRECTION AS A FUNCTION OF THE MEAN RADIUS.
C  THE LENGTH SCALE IS NONDIMENSIONALIZED ON THE DUCT RADIUS.
C  NOTE SIGMAR.LE.R.LE.1.
C
C  12/21/79 VERSION USES A CONSTANT
C
       B=0.50
       SCALLR=B
       RETURN
       END
```

SUBROUTINE SCALLX

Program yields the turbulence length scale in the axial direction.

```
C   <MTHEOBALD>SCALLX.FOR;4    Mon 29-Sep-80 11:11PM     PAGE 1

          FUNCTION SCALLX(R)
C
C   PROGRAM CALCULATES THE INLET TURBULENCE LENGTH SCALE
C   IN THE AXIAL DIRECTION AS A FUNCTION OF THE MEAN RADIUS.
C   LENGTH SCALE IS NONDIMENSIONALIZED ON THE DUCT RADIUS.
C
C    12/16/79 VERSTON USES A CONSTANT
C   NOTE SIGMAR.LE.R.LE.1.
      B=40.
      SCALLX=B
      RETURN
      END
```

SUBROUTINE STHETA

Program yields the turbulence length scale in the circumferential direction.

```
C   <MTHEOBALD>STHETA.FOR;4    Mon 29-Sep-80 11:12PM    PAGE 1

          FUNCTION STHETA(R)
C
C   PROGRAM CALCULATES THE INLET TURBULENCE LENGTH SCALE
C   IN THE AZIMUTHAL DIRECTION AS A FUNCTION OF MEAN RADIUS.
C   THE LENGTH SCALE IS NONDIMENSIONALIZED ON THE DUCT RADIUS.
C
        DATA PI/3.14159/
C
C    12/18/79 VERSION USES A CONSTANT
C   NOTE SIGMAR.LE.R.LE.1.
        B=0.5
        STHETA=B
        RETURN
        END
```

PROGRAM MAPIN/
SUBROUTINE MAPPER
(with sample execution)

Program lists which duct modes will propagate for inlet turbulence noise.

```
          DIMENSION RDATA(20)
          REAL MT,MA
C
C
C
C     INPUT VARIABLES TO SUBROUTINE MAPPER FOR LISTING OF ALL
C     PROPAGATING MODES FOR TURBULENCE PROGRAMS AT A
C     GIVEN FREQUENCY
C
1001      FORMAT(G20.8)
          WRITE(5,2003)
2003      FORMAT(' HUB RADIUS DIVIDED BY DUCT RADIUS =')
          READ(5,1001) SIGMAR
          WRITE(5,2103) SIGMAR
2103      FORMAT(' SIGMAR=',E16.8)
C
C
          WRITE(5,2030)
2030      FORMAT(' MT =')
          READ(5,1001) MT
          WRITE(5,2130) MT
2130      FORMAT(' MT=',E16.8)
          WRITE(5,2031)
2031      FORMAT(' MA =')
          READ(5,1001) MA
          WRITE(5,2131) MA
2131      FORMAT(' MA =',E16.8)
          WRITE(5,2037)
2037      FORMAT(' ACCURACY OF BESSEL FN =')
          READ(5,1001) EB
          WRITE(5,2137) EB
2137      FORMAT(' EB=',E16.8)
          WRITE(5,2038)
2038      FORMAT(' ACCURACY OF CONVERGENCE TO ROOT XMN =')
          READ(5,1001) EC
          WRITE(5,2138) EC
2138      FORMAT(' EC=',E16.8)
C
C
C
C
          RDATA(1)=MT
          RDATA(2)=MA
          RDATA(9)=EB
          RDATA(10)=EC
C
C
                RDATA(5)=SIGMAR
C
C
          CALL MAPPER(RDATA)
          STOP
          END
```

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE MAPPER(RDATA)
      DIMENSION IER(3),RDATA(20)
     REAL MT,MA
C
C
C
C
C   PROGRAM LISTS ALL PROPAGATING MODES FOR INLET AND WAKE
C   TURBULENCE PROGRAMS WITHOUT CALCULATING MODAL
C   AMPLITUDES.  USE THIS PROGRAM AS A PRECURSOR TO
C   INTURB OR WATURB SEARCH ROUTINES TO ESTIMATE RUN TIME.
C
C
C
C
 1001  FORMAT(G20.8)
       WRITE(5,110)
 110   FORMAT(' NOISE FREQUENCY/SHAFT FREQUENCY=')
       READ(5,1001) OMEGA
       RDATA(8)=OMEGA
 115   FORMAT(' OMEGA=',E10.4)
       WRITE(5,115) RDATA(8)
       MT=RDATA(1)
       MA=RDATA(2)
       SIGMAR=RDATA(5)
       EB=RDATA(9)
       EC=RDATA(10)
C
C
       XMAX=OMEGA*MT/(1.-MA**2)**.5
C
C
       WRITE(5,2001) XMAX
2001   FORMAT(' XMN(MAX) FOR PROPAGATION =',E16.8)
C
C
C
C
C
C   START WITH PLANE WAVE MODE (0,1).  COUNT UP IN N, THEN M
C   UNTIL ALL POSSIBLE HIGHER MODES ARE CUT OFF.
C
       J=1
       M=0
C
C
C
C   RESTART N COUNT HERE FOR NEW M
C
15     N=0
C
```

```
C   INCREMENT N
C
20        N=N+1
C
C
          MABS=IABS(M)
C
C
C
          CALL ANRT (MABS,N,SIGMAR,EB,EC,XMN,IEB,IEC)
C
C
C
C   IF MODE IS CUT OFF, GO TO 6000
C
          IF(XMN.GE.XMAX) GO TO 6000
C
C
          WRITE(5,3000)
3000      FORMAT('0MODE DATA')
          WRITE(5,3001) OMEGA,M,N,XMN
 3001     FORMAT('0OMEGA=',E10.4,'    M=',I3,'    N=',I3,'    XMN=',E10.4)
          IF(XMN.EQ.0.) GO TO 1000
          COFFRA=XMAX/XMN
          WRITE(5,4000) COFFRA
 4000     FORMAT(' CUTOFF RATIO FOR MODE=',E10.4)
          GO TO 1005
 1000     CONTINUE
          WRITE(5,4005)
 4005     FORMAT(' PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE')
 1005     CONTINUE
          WRITE(5,3002) IEB
3002      FORMAT(' SUM OF BESSEL FUNCTION ERROR CODES =',I3)
          WRITE(5,3003) IEC
3003      FORMAT(' ERROR CODE FOR CONVERGENCE TO ROOT XMN =',I3)
C
C   NORMALIZE MODE AMPLITUDE
C
          CALL EIGEN (MABS,SIGMAR,XMN,AMN,BMN,EB,IER)
C
C
          WRITE(5,3004) AMN,BMN
3004      FORMAT(' AMN =',E16.8,'    BMN =',E16.8)
          WRITE(5,3005) IER
3005      FORMAT(' ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC =',8I2)
C
C
C
C
C
C
```

```
C
C
            MMM=M
C
C   DO NOT SWITCH SIGN ON M IF M=0
C
            IF(M.EQ.0) GO TO 20
C
          MMM=-MMM
C
C
          WRITE(5,8300) MMM,N,RDATA(8)
 8300       FORMAT('0M=',I5,'   N=',I5,'   OMEGA=',E10.4)
C
C   SWITCH SIGN ON M
C
C
C
C
C
          GO TO 20
C
C
C
C
C
C   IF MODE IS CUTOFF, DECIDE WHICH MODE TO TRY NEXT.
C
6000      IF(N.EQ.1) GO TO 7000
C
C
          NMAX=N-1
          WRITE(5,6001) NMAX
6001      FORMAT('0LARGEST PROPAGATING N FOR THIS M =',I3)
C
C
          J=J+1
C
C   INCREMENT M
C
          M=M+1
          GO TO 15
C
C
C
C   N=1 TO REACH THIS POINT
 7000     CONTINUE
C
C
          WRITE(5,7001)
 7001     FORMAT(' NO MORE PROPAGATING MODES FOR THIS OMEGA')
 10       CONTINUE
          WRITE(5,9001)
9001      FORMAT(' PROBLEM COMPLETED')
          RETURN
          END
```

Sample Execution of MAPIN/MAPPER

```
@MAPPER
HUB RADIUS DIVIDED BY DUCT RADIUS =
.484
SIGMAR=  0.48400000E+00
MT =
.508
MT=  0.50800000E+00
MA =
.323
MA =  0.32300000E+00
ACCURACY OF BESSEL FN =
.0001
EB=  0.10000000E-03
ACCURACY OF CONVERGENCE TO ROOT XMN =
.0001
EC=  0.10000000E-03
NOISE FREQUENCY/SHAFT FREQUENCY=
15.
OMEGA= .1500E+02
XMN(MAX) FOR PROPAGATION =  0.80515728E+01

MODE DATA

OMEGA= .1500E+02    M=  0    N=  1    XMN= .0000E+00
PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.10000000E+01     BMN =  0.00000000E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

MODE DATA

OMEGA= .1500E+02    M=  0    N=  2    XMN= .6205E+01
CUTOFF RATIO FOR MODE= .1298E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN = -0.26166472E+01     BMN =  0.27171760E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

LARGEST PROPAGATING N FOR THIS M =  2
%FRSAPR Floating underflow      PC= 3100

%FRSAPR Floating underflow      PC= 3146
```

ORIGINAL PAGE IS
OF POOR QUALITY

MODE DATA

OMEGA= .1500E+02 M= 1 N= 1 XMN= .1371E+01
CUTOFF RATIO FOR MODE= .5873E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.15152306E+01 BMN = -0.42119467E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -1 N= 1 OMEGA= .1500E+02

MODE DATA

OMEGA= .1500E+02 M= 1 N= 2 XMN= .6390E+01
CUTOFF RATIO FOR MODE= .1260E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.15356938E+01 BMN = 0.32269858E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0

M= -1 N= 2 OMEGA= .1500E+02

LARGEST PROPAGATING N FOR THIS M = 2

MODE DATA

OMEGA= .1500E+02 M= 2 N= 1 XMN= .2708E+01
CUTOFF RATIO FOR MODE= .2973E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.20525722E+01 BMN = -0.42471320E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -2 N= 1 OMEGA= .1500E+02

MODE DATA

OMEGA= .1500E+02 M= 2 N= 2 XMN= .6920E+01
CUTOFF RATIO FOR MODE= .1164E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.36560636E+01 BMN = 0.74837045E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

```
M=   -2   N=    2   OMEGA= .1500E+02

LARGEST PROPAGATING N FOR THIS M =  2

MODE DATA

OMEGA= .1500E+02    M=  3   N=  1    XMN= .3987E+01
CUTOFF RATIO FOR MODE= .2020E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.26023450E+01     BMN = -0.32589376E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -3   N=    1   OMEGA= .1500E+02

MODE DATA

OMEGA= .1500E+02    M=  3   N=  2    XMN= .7747E+01
CUTOFF RATIO FOR MODE= .1039E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.35705322E+01     BMN = -0.89511162E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -3   N=    2   OMEGA= .1500E+02

LARGEST PROPAGATING N FOR THIS M =  2

MODE DATA

OMEGA= .1500E+02    M=  4   N=  1    XMN= .5200E+01
CUTOFF RATIO FOR MODE= .1549E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.31336392E+01     BMN = -0.20930877E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -4   N=    1   OMEGA= .1500E+02

LARGEST PROPAGATING N FOR THIS M =  1
```

```
MODE DATA

OMEGA= .1500E+02    M=  5   N=  1   XMN= .6355E+01
CUTOFF RATIO FOR MODE= .1267E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.36182370E+01    BMN = -0.11930907E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -5   N=    1   OMEGA= .1500E+02

LARGEST PROPAGATING N FOR THIS M =  1

MODE DATA

OMEGA= .1500E+02    M=  6   N=  1   XMN= .7473E+01
CUTOFF RATIO FOR MODE= .1077E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.40542452E+01    BMN = -0.61590101E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -6   N=    1   OMEGA= .1500E+02

LARGEST PROPAGATING N FOR THIS M =  1
NO MORE PROPAGATING MODES FOR THIS OMEGA
PROBLEM COMPLETED
STOP

END OF EXECUTION
CPU TIME: 29.82 ELAPSED TIME: 3:4.03
EXIT.
^C
@MAPPER
HUB RADIUS DIVIDED BY DUCT RADIUS =
.484
SIGMAR=  0.48400000E+00
MT =
.508
MT=  0.50800000E+00
MA =
.323
MA =  0.32300000E+00
ACCURACY OF BESSEL FN =
.0001
EB=  0.10000000E-03
```

```
ACCURACY OF CONVERGENCE TO ROOT XMN =
.0001
EC=  0.10000000E-03
NOISE FREQUENCY/SHAFT FREQUENCY=
30.
OMEGA= .3000E+02
XMN(MAX) FOR PROPAGATION =  0.16103146E+02

MODE DATA

OMEGA= .3000E+02    M=  0    N=  1    XMN= .0000E+00
PLANE WAVE MODE: CUTOFF RATIO IS + INFINITE
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.10000000E+01     BMN =  0.00000000E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

MODE DATA

OMEGA= .3000E+02    M=  0    N=  2    XMN= .6205E+01
CUTOFF RATIO FOR MODE= .2595E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN = -0.26166472E+01     BMN =  0.27171760E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

MODE DATA

OMEGA= .3000E+02    M=  0    N=  3    XMN= .1224E+02
CUTOFF RATIO FOR MODE= .1316E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN = -0.24929737E+01     BMN =  0.47091179E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

LARGEST PROPAGATING N FOR THIS M =  3
%FRSAPR Floating underflow      PC= 3100

%FRSAPR Floating underflow      PC= 3146


MODE DATA

OMEGA= .3000E+02    M=  1    N=  1    XMN= .1371E+01
CUTOFF RATIO FOR MODE= .1175E+02
SUM OF BESSEL FUNCTION ERROR CODES =  0
```

```
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.15152306E+01    BMN = -0.42119467E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -1   N=   1   OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  1   N=  2   XMN= .6390E+01
CUTOFF RATIO FOR MODE= .2520E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.19356938E+01    BMN =  0.32269858E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -1   N=   2   OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  1   N=  3   XMN= .1232E+02
CUTOFF RATIO FOR MODE= .1307E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.43514410E+01    BMN =  0.30726765E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -1   N=   3   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  3

MODE DATA

OMEGA= .3000E+02    M=  2   N=  1   XMN= .2708E+01
CUTOFF RATIO FOR MODE= .5946E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.20525722E+01    BMN = -0.42471320E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -2   N=   1   OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  2   N=  2   XMN= .6920E+01
CUTOFF RATIO FOR MODE= .2327E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
```

ORIGINAL PAGE IS
OF POOR QUALITY

ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.36560636E+01 BMN = 0.74837045E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -2 N= 2 OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02 M= 2 N= 3 XMN= .1258E+02
CUTOFF RATIO FOR MODE= .1280E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.44592966E+01 BMN = -0.29055613E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -2 N= 3 OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M = 3

MODE DATA

OMEGA= .3000E+02 M= 3 N= 1 XMN= .3987E+01
CUTOFF RATIO FOR MODE= .4039E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.26023450E+01 BMN = -0.32589376E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -3 N= 1 OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02 M= 3 N= 2 XMN= .7747E+01
CUTOFF RATIO FOR MODE= .2079E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.35705322E+01 BMN = -0.89511162E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -3 N= 2 OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02 M= 3 N= 3 XMN= .1301E+02
CUTOFF RATIO FOR MODE= .1238E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0

ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = -0.18754669E+00 BMN = 0.53089864E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -3 N= 3 OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M = 3

MODE DATA

OMEGA= .3000E+02 M= 4 N= 1 XMN= .5200E+01
CUTOFF RATIO FOR MODE= .3097E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.31336392E+01 BMN = -0.20930877E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -4 N= 1 OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02 M= 4 N= 2 XMN= .8798E+01
CUTOFF RATIO FOR MODE= .1830E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.33061669E+01 BMN = -0.15169850E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -4 N= 2 OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02 M= 4 N= 3 XMN= .1359E+02
CUTOFF RATIO FOR MODE= .1185E+01
SUM OF BESSEL FUNCTION ERROR CODES = 0
ERROR CODE FOR CONVERGENCE TO ROOT XMN = 0
AMN = 0.34317489E+01 BMN = 0.40298049E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M= -4 N= 3 OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M = 3

ORIGINAL PAGE IS
OF POOR QUALITY

```
MODE DATA

OMEGA= .3000E+02    M=  5    N=  1    XMN= .6355E+01
CUTOFF RATIO FOR MODE= .2534E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.36182370E+01    BMN = -0.11930907E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -5    N=    1    OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  5    N=  2    XMN= .9995E+01
CUTOFF RATIO FOR MODE= .1611E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.32789852E+01    BMN = -0.15982724E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -5    N=    2    OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  5    N=  3    XMN= .1432E+02
CUTOFF RATIO FOR MODE= .1125E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.50087871E+01    BMN =  0.16051647E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -5    N=    3    OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  3

MODE DATA

OMEGA= .3000E+02    M=  6    N=  1    XMN= .7473E+01
CUTOFF RATIO FOR MODE= .2155E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.40542452E+01    BMN = -0.61590101E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -6    N=    1    OMEGA= .3000E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
MODE DATA

OMEGA= .3000E+02    M=  6   N=  2   XMN= .1127E+02
CUTOFF RATIO FOR MODE= .1429E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.34447685E+01     BMN = -0.14296053E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -6   N=    2   OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  6   N=  3   XMN= .1518E+02
CUTOFF RATIO FOR MODE= .1061E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.51904865E+01     BMN = -0.42382837E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -6   N=    3   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  3

MODE DATA

OMEGA= .3000E+02    M=  7   N=  1   XMN= .8565E+01
CUTOFF RATIO FOR MODE= .1880E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.44478450E+01     BMN = -0.30159084E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -7   N=    1   OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02    M=  7   N=  2   XMN= .1257E+02
CUTOFF RATIO FOR MODE= .1281E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.37694693E+01     BMN = -0.11581721E+01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -7   N=    2   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  2
```

```
MODE DATA

OMEGA= .3000E+02     M=  8    N=  1    XMN= .9641E+01
CUTOFF RATIO FOR MODE= .1670E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.48101954E+01     BMN = -0.14625833E-01
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -8    N=     1    OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02     M=  8    N=  2    XMN= .1386E+02
CUTOFF RATIO FOR MODE= .1162E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.41235105E+01     BMN = -0.85984372E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -8    N=     2    OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  2

MODE DATA

OMEGA= .3000E+02     M=  9    N=  1    XMN= .1071E+02
CUTOFF RATIO FOR MODE= .1504E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.51513706E+01     BMN = -0.65355876E-02
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -9    N=     1    OMEGA= .3000E+02

MODE DATA

OMEGA= .3000E+02     M=  9    N=  2    XMN= .1512E+02
CUTOFF RATIO FOR MODE= .1065E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.44743524E+01     BMN = -0.59372922E+00
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=   -9    N=     2    OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  2
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
MODE DATA

OMEGA= .3000E+02    M= 10   N=  1   XMN= .1177E+02
CUTOFF RATIO FOR MODE= .1368E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.54757875E+01    BMN = -0.30348101E-02
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -10    N=    1   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  1

MODE DATA

OMEGA= .3000E+02    M= 11   N=  1   XMN= .1283E+02
CUTOFF RATIO FOR MODE= .1256E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.57877752E+01    BMN = -0.13177058E-02
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -11    N=    1   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  1

MODE DATA

OMEGA= .3000E+02    M= 12   N=  1   XMN= .1388E+02
CUTOFF RATIO FOR MODE= .1160E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.60893310E+01    BMN = -0.63729882E-03
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -12    N=    1   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  1

MODE DATA

OMEGA= .3000E+02    M= 13   N=  1   XMN= .1493E+02
CUTOFF RATIO FOR MODE= .1079E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.63824353E+01    BMN = -0.18130501E-03
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
M=  -13   N=    1   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  1

MODE DATA

OMEGA= .3000E+02    M= 14   N=  1   XMN= .1598E+02
CUTOFF RATIO FOR MODE= .1008E+01
SUM OF BESSEL FUNCTION ERROR CODES =  0
ERROR CODE FOR CONVERGENCE TO ROOT XMN =  0
AMN =  0.66676797E+01    BMN = -0.10550421E-03
ERROR CODE FOR BESSEL FNS IN AMN AND BMN CALC = 0 0 0 0 0 0 0 0

M=  -14   N=    1   OMEGA= .3000E+02

LARGEST PROPAGATING N FOR THIS M =  1
NO MORE PROPAGATING MODES FOR THIS OMEGA
PROBLEM COMPLETED
STOP

END OF EXECUTION
CPU TIME: 1:40.60       ELAPSED TIME: 8:6.24
EXIT.
```