NASA Technical Paper 1295

DYGABCD - A Program for Calculating Linear A, B, C, and D Matrices From a Nonlinear Dynamic Engine Simulation

Lucille C. Geyser

SEPTEMBER 1978

NASA

NASA Technical Paper 1295

DYGABCD - A Program for Calculating Linear A, B, C, and D Matrices From a Nonlinear Dynamic Engine Simulation

Lucille C. Geyser
Lewis Research Center
Cleveland, Ohio

National Aeronautics
and Space Administration

**Scientific and Technical
Information Office**

1978

CONTENTS

DYGABCD - A PROGRAM FOR CALCULATING LINEAR A, B, C, AND D MATRICES FROM A NONLINEAR DYNAMIC ENGINE SIMULATION

by Lucille C. Geyser

Lewis Research Center

SUMMARY

A digital computer program, DYGABCD, has been developed that will generate linearized, dynamic models of simulated turbofan and turbojet engines. DYGABCD is based on an earlier computer program, DYNGEN, that is capable of calculating simulated nonlinear steady-state and transient performance of one- and two-spool turbojet engines or two- and three-spool turbofan engines.

Most control design techniques require linear system descriptions. For multiple-input/multiple-output systems such as turbine engines, state space matrix descriptions of the system are often desirable. DYGABCD computes the state space matrices, commonly referred to as the A, B, C, and D matrices, required for a linear system description. The report discusses the analytical approach and provides a users manual, FORTRAN listings, and a sample case. NASA TN D-7901, describing DYNGEN, is a necessary adjunct to this report.

INTRODUCTION

Digital computers are more frequently being used to accurately simulate the steady-state and transient performance characteristics of turbojet and turbofan engines. These simulations are capable of modeling these engines over the complete range of power settings and flight conditions. DYNGEN (ref. 1) is one such engine simulation that is generalized to permit the user to model many different engine configurations. It is based on the generalized steady-state engine modeling codes GENENG (ref. 2) and GENENG II (ref. 3). These codes are based on a technique called SMOTE (refs. 4 and 5).

To accomplish the modeling accuracy being demonstrated by DYNGEN and other digital computer engine simulation codes, nonlinear steady-state and transient physical relationships must be included. Most of the analytical design approaches used by control designers, however, require linear descriptions of the steady-state and transient characteristics of the engine. Linear approximations to nonlinear systems can be obtained at a particular operating point (a particular equilibrium condition of power setting and flight condition) for small excursions about that particular point.

For a system as complex as an engine, this linear system description usually consists of a multitude of interdependent linear algebraic and differential equations that can be written in a compact form by using matrix notation. A widely accepted linear dynamic system representation (ref. 6) uses a system matrix A, a control matrix B, an output matrix C, and a feed-forward matrix D. To describe an engine on a linearized basis over its complete operating envelope, however, a whole family of suitably selected linear approximations would be required.

This report, then, describes DYGABCD, a digital computer program that calculates the A, B, C, and D matrices for any turbofan or turbojet engine that is capable of being simulated with the DYNGEN program. DYGABCD can be used (1) exactly as DYNGEN to generate engine steady-state and transient performance characteristics; or (2) to generate the linear model A, B, C, and D matrices. The method of calculation employed in DYGABCD to generate these matrices can easily be used in any transient simulation that uses a backward difference integration technique (ref. 2).

The report first presents (1) the theoretical or analytical bases for linear model approximations to nonlinear systems, (2) a definition of the compact matrix formulation, and (3) a derivation of the equations necessary for determining matrix element values from the transient simulation. Next presented is a users guide for DYGABCD, including complete descriptions of those subroutines that have computational techniques different from DYNGEN. The FORTRAN symbols used are defined in appendix A, complete listings are shown in appendix B, and a sample case is presented in appendix C.

DYGABCD is written in FORTRAN IV and was used on the IBM 360/67 with the time-sharing system (TSS). The program should run equally well with the operating system (OS) on an IBM 360 or 370 series computer.

ANALYTICAL BACKGROUND

The simulation of a complex turbine engine process involves both linear and nonlinear algebraic and differential equations. Multiple equations of this type can generally be expressed in compact vector notation as

$$\dot{x} = f(x, u) \tag{1}$$

$$y = g(x, u) \tag{2}$$

where x is a vector of the time-varying states of the engine, u is a vector of the system control inputs to the engine, and y is a vector of the engine's outputs. The vector $\dot{x}$ represents the time rate of change of each of the engine's states. The following table lists some typical states, inputs, and outputs for a turbine engine system:

States	Inputs	Outputs
Temperatures	Fuel flows	Engine airflows
Pressures	Nozzle area	Thrust
Speeds	Bleeds	Specific fuel consumption
		Fan surge margin
		Compressor surge margin

Since most control design techniques require a linear description of the process to be controlled, a linear approximation to the nonlinear system needs to be developed. At a particular operating point, equations (1) and (2) can be represented by a Taylor series expansion about that point. If the vectors x and u are assumed to be single elements (i.e., scalars), equation (1) may be written as follows:

$$\dot{x}_0 = f(x_0, u_0) \tag{3}$$

The Taylor series expansion about the point (x_0, u_0) is

$$\dot{x} = f(x_0, u_0) + \left.\frac{\partial f}{\partial x}\right|_{\substack{x=x_0 \\ u=u_0}} (x - x_0) + \left.\frac{\partial f}{\partial u}\right|_{\substack{x=x_0 \\ u=u_0}} (u - u_0)$$

$$+ \left.\frac{\partial^2 f}{\partial x^2}\right|_{\substack{x=x_0 \\ u=u_0}} (x - x_0)^2 + \left.\frac{\partial^2 f}{\partial u^2}\right|_{\substack{x=x_0 \\ u=u_0}} (u - u_0)^2$$

$$+ \text{Other higher order terms} \tag{4}$$

If $(x = x_0,\ u = u_0)$ is a steady-state operating point, the system is in equilibrium and hence $\dot{x}_0 = 0$. Thus, by using equations (1) and (3), equation (4) can be rewritten as

$$\dot{x} = \left.\frac{\partial \dot{x}}{\partial x}\right|_{op} (x - x_0) + \left.\frac{\partial \dot{x}}{\partial u}\right|_{op} (u - u_0)$$

$$+ \left.\frac{\partial^2 x}{\partial x^2}\right|_{op} (x - x_0)^2 + \left.\frac{\partial^2 u}{\partial u^2}\right|_{op} (u - u_0)^2$$

$$+ \text{Other higher order terms} \tag{5}$$

where op denotes operating point.

In a suitable neighborhood of the steady-state operating point (x_0, u_0), the higher order terms of equation (4) are negligible, and equation (1) can be approximated by using only the first partial derivative terms. By letting $\Delta x = x - x_0$, $\Delta u = u - u_0$, and $\Delta\dot{x} = \dot{x} - \dot{x}_0$, equation (5) becomes the linear equation

$$\Delta\dot{x} = \left.\frac{\partial \dot{x}}{\partial x}\right|_{op} \Delta x + \left.\frac{\partial \dot{x}}{\partial u}\right|_{op} \Delta u \tag{6}$$

Equation (2) may be linearized in similar fashion for the output y, resulting in

$$\Delta y = \left.\frac{\partial y}{\partial x}\right|_{op} \Delta x + \left.\frac{\partial y}{\partial u}\right|_{op} \Delta u \tag{7}$$

Definitions of A, B, C, and D Matrices

An actual system with a multitude of states, inputs, and outputs, when linearized about an operating point, can be described by the following linear vector-matrix equations:

$$\Delta\dot{x} = A\,\Delta x + B\,\Delta u \tag{8}$$

$$\Delta y = C\,\Delta x + D\,\Delta u \tag{9}$$

where

Δx n-dimensioned vector of state deviations

Δu c-dimensioned vector of system control input deviations

Δy o-dimensioned vector of system output deviations

A n- by n-dimensioned state system matrix

B n- by c-dimensioned control input matrix

C o- by n-dimensioned output matrix

D o- by c-dimensioned feed-forward matrix

n number of states

c number of control inputs

o number of outputs

The system matrix A is a matrix of numbers that are the values of particular partial derivatives at a specific operating point. When the partial derivative notation of equations (6) and (7) is used, equations (10) to (13) show the elemental arrangements of the matrices A, B, C, and D.

$$A = \begin{bmatrix} \left.\frac{\partial \dot{x}_1}{\partial x_1}\right|_{op} & \left.\frac{\partial \dot{x}_1}{\partial x_2}\right|_{op} & \cdots & \left.\frac{\partial \dot{x}_1}{\partial x_n}\right|_{op} \\ \left.\frac{\partial \dot{x}_2}{\partial x_1}\right|_{op} & & & \\ \vdots & & & \vdots \\ \left.\frac{\partial \dot{x}_n}{\partial x_1}\right|_{op} & & & \end{bmatrix} \tag{10}$$

$$B = \begin{bmatrix} \left.\frac{\partial \dot{x}_1}{\partial u_1}\right|_{op} & \left.\frac{\partial \dot{x}_1}{\partial u_2}\right|_{op} & \cdots & \left.\frac{\partial \dot{x}_1}{\partial u_c}\right|_{op} \\ \left.\frac{\partial \dot{x}_2}{\partial u_1}\right|_{op} & & & \\ \vdots & & & \vdots \\ \left.\frac{\partial \dot{x}_n}{\partial u_1}\right|_{op} & & & \end{bmatrix} \tag{11}$$

$$C = \begin{bmatrix} \left.\frac{\partial y_1}{\partial x_1}\right|_{op} & \left.\frac{\partial y_1}{\partial x_2}\right|_{op} & \cdots & \left.\frac{\partial y_1}{\partial x_n}\right|_{op} \\ \left.\frac{\partial y_2}{\partial x_1}\right|_{op} & & & \\ \vdots & & & \vdots \\ \left.\frac{\partial y_o}{\partial x_1}\right|_{op} & & & \end{bmatrix} \tag{12}$$

$$D = \begin{bmatrix} \left.\frac{\partial y_1}{\partial u_1}\right|_{op} & \left.\frac{\partial y_1}{\partial u_2}\right|_{op} & \cdots & \left.\frac{\partial y_1}{\partial u_c}\right|_{op} \\ \left.\frac{\partial y_2}{\partial u_1}\right|_{op} & & & \\ \vdots & & & \vdots \\ \left.\frac{\partial y_o}{\partial u_1}\right|_{op} & & & \end{bmatrix} \tag{13}$$

Solutions for Full Forms of A, B, C, and D Matrices

It is our intent to take a DYNGEN digital computer simulation with its nonlinear dynamic equations and to develop a method for extracting from it the elements of the A, B, C, and D matrices at particular engine operating points.

Because of the backward finite-difference integration technique required by DYNGEN (SMOTE technique), it is necessary to calculate the inverse of the state system matrix A. To accomplish this, equation (8) can be rewritten as

$$\Delta x = A^{-1} \Delta \dot{x} - A^{-1} B \Delta u \tag{14}$$

To solve for A^{-1}, all system control input deviations Δu from the operating point are set to zero. Thus,

$$\Delta x = A^{-1} \Delta \dot{x} \tag{15}$$

Each element of A^{-1} is then approximated as

$$\left(A^{-1}\right)_{ij} \approx \frac{\Delta x_i}{\Delta \dot{x}_j} \tag{16}$$

where the state derivatives $\dot{x}_j$ are perturbed, one at a time, as described in the following paragraph.

A steady-state vector of the states x is calculated. The state derivatives $\dot{x}$ at this steady-state point are therefore zero. The first $\dot{x}$ is set to some suitable value, $\Delta \dot{x}$, with the rest of the $\dot{x}$'s held to zero. A new state vector x^* is calculated. The difference between x^* and x is the vector Δx associated with this first $\Delta \dot{x}$. Each of the other $\dot{x}$'s is then set to some suitable value, $\Delta \dot{x}$, one at a time, with all other $\dot{x}$'s set to zero each time. A new vector x^* and a new vector Δx are calculated for each $\Delta \dot{x}$. A matrix of these Δx column vectors is formed. Each column vector is divided by the $\Delta \dot{x}$ that produced it, thereby forming the A^{-1} matrix referred to in equation (16). The A^{-1} matrix is then inverted to obtain the A matrix.

Solve for the control matrix B as follows: Under equilibrium conditions, $\dot{x} = 0$, equation (8) becomes

$$\Delta x = -A^{-1} B \Delta u \tag{17}$$

Each element of $-A^{-1}B$ is then approximated as

$$\left(-A^{-1}B\right)_{ij} \approx \frac{\Delta x_i}{\Delta u_j} \tag{18}$$

where the system control inputs u_j are perturbed, one at a time, as described in the following paragraph.

A steady-state vector of the states x is calculated. The first system control input u is changed by some suitable value Δu with the remainder of the u's held to their steady-state values. A new state vector x^* is calculated. The difference between x^* and x is the vector Δx associated with this first Δu. Each of the other u's is then perturbed by some suitable Δu, one at a time, with all other u's remaining constant each time. A new vector x^* and a new vector Δx are calculated for each Δu. A matrix of these Δx column vectors is then formed. Each column vector is divided by the Δu that produced it, thereby forming the $-A^{-1}B$ matrix approximated by equation (18). This matrix is then premultiplied by $-A$ to obtain the B matrix.

To solve for the output matrix C, substitute the equation for Δx (eq. (14)) in the equation for Δy (eq. (9)), yielding

$$\Delta y = CA^{-1}\,\Delta \dot{x} - CA^{-1}B\,\Delta u + D\,\Delta u \tag{19}$$

When the system control inputs are held fixed, equation (19) becomes

$$\Delta y = CA^{-1}\,\Delta \dot{x} \tag{20}$$

Each element of CA^{-1} is then approximated as

$$\left(CA^{-1}\right)_{ij} \approx \frac{\Delta y_i}{\Delta \dot{x}_j} \tag{21}$$

where the state derivatives $\dot{x}_j$ are perturbed one at a time, as was done in obtaining A^{-1}.

A steady-state output vector y is calculated. The state derivatives $\dot{x}$ at this steady-state point are therefore zero. The first $\dot{x}$ is set to some suitable value, $\Delta \dot{x}$, with the rest of the $\dot{x}$'s held to zero. A new output vector y^* is calculated. The difference between y^* and y is the vector Δy associated with this first $\Delta \dot{x}$. Each of the other $\dot{x}$'s is then set to some suitable value $\Delta \dot{x}$, one at a time, with all other $\dot{x}$'s

set to zero each time. A new vector y^* and a new vector Δy are calculated for each $\Delta\dot{x}$. A matrix of these Δy column vectors is then formed. Each column vector is divided by the $\Delta\dot{x}$ that produced it, thereby forming the CA^{-1} matrix approximated by equation (21). This matrix is then postmultiplied by A to obtain the C matrix.

Solving equation (9) for the feed-forward matrix D yields the approximation

$$D_{ij} \approx \frac{\Delta y_i}{\Delta u_j} - C\frac{\Delta x_i}{\Delta u_j} \tag{22}$$

where the system control inputs u_j are perturbed, one at a time, as described in the following paragraph.

Two steady-state vectors, x (the state vector) and y (the output vector), are calculated. The first system control input u is changed by some suitable value Δu, with the remainder of the u's held to their steady-state value. A new state vector x^* is calculated. And a new output vector y^* is calculated. The difference between x^* and x is the vector Δx associated with this first Δu. The difference between y^* and y is the vector Δy associated with this first Δu. Each of the other u's is then perturbed by some suitable value Δu, one at a time, with all other u's remaining constant each time. A new vector x^* and a new vector Δx are calculated for each Δu. A new vector y^* and a new vector Δy are also calculated for each Δu. A matrix of the Δx column vectors is then formed, and each column vector is divided by the Δu that produced it. A matrix of the Δy column vectors is also formed, and each column vector is again divided by the Δu that produced it. By substituting these two matrices in equation (22), we can solve for D.

Solutions for Reduced Forms of A, B, C, and D Matrices

Once the complete linear model of the dynamic system is obtained, it may be desirable, for certain reasons, to reduce the complexity of the linear description. For example, the eigenvalues of the full system either may be beyond the frequency range of interest or may have little overall effect on the system dynamic characteristics. This reduction can be done by several methods.

By the method of Weinberg and Adams (ref. 7), equation (14) is rewritten in partitioned form as

$$\frac{\Delta x_1}{\Delta x_2} = \left[\begin{array}{c|c} \hat{A}_{11} & \hat{A}_{12} \\ \hline \hat{A}_{21} & \hat{A}_{22} \end{array}\right] \frac{\Delta \dot{x}_1}{\Delta \dot{x}_2} - \left[\begin{array}{c|c} \hat{A}_{11} & \hat{A}_{21} \\ \hline \hat{A}_{12} & \hat{A}_{22} \end{array}\right] \frac{B_1}{B_2} \Delta u \tag{23}$$

where

Δx_1 r-dimensioned part of state deviation vector

Δx_2 (n-r)-dimensioned part of state deviation vector

Δu c-dimensioned system control input deviation vector

$\hat{A}_{11}$ r- by r-dimensioned upper left partition of A^{-1} matrix

$\hat{A}_{12}$ r- by (n-r)-dimensioned upper right partition of A^{-1} matrix

$\hat{A}_{21}$ (n-r)- by r-dimensioned lower left partition of A^{-1} matrix

$\hat{A}_{22}$ (n-r)- by (n-r)-dimensioned lower right partition of A^{-1} matrix

B_1 r- by c-dimensioned part of control input matrix

B_2 (n-r)- by c-dimensioned part of control input matrix

n total number of states

r number of states desired

c number of control inputs

Before calculating the partitioned $\hat{A}$ matrix, permute the order of the Δx's and the corresponding $\Delta \dot{x}$'s, so that the Δx's to be retained are all included in Δx_1. Those remaining make up the elements of Δx_2. The elements of Δx_2 are assumed to be negligible in determining the system transient response. Hence, setting $\Delta \dot{x}_2 = 0$ in equation (23) yields

$$\Delta x_1 = \hat{A}_{11} \Delta \dot{x}_1 - \left(\hat{A}_{11} B_1 + \hat{A}_{12} B_2\right) \Delta u \tag{24}$$

Solving for $\Delta \dot{x}_1$ gives

$$\Delta \dot{x}_1 = \left(\hat{A}_{11}\right)^{-1} \Delta x_1 + \left(\hat{A}_{11}\right)^{-1} \left(\hat{A}_{11} B_1 + \hat{A}_{12} B_2\right) \Delta u \tag{25}$$

or

$$\Delta \dot{x}_1 = A_R \Delta x_1 + B_R \Delta u \tag{26}$$

where

$$A_R = \left(\hat{A}_{11}\right)^{-1} \tag{27}$$

and

$$B_R = \left(\hat{A}_{11}\right)^{-1} \left(\hat{A}_{11}B_1 + \hat{A}_{12}B_2\right) \tag{28}$$

Note that it is necessary to compute only the upper left quadrant of the A^{-1} matrix (the r-by-r $\hat{A}_{11}$) and then to invert it to obtain A_R. The quantity $\hat{A}_{11}B_1 + \hat{A}_{12}B_2$ consists of the first r rows of the $A^{-1}B$ matrix from equation (18). Thus, only the first r rows of $-A^{-1}B$ need to be calculated. These rows are then premultiplied by $-A_R$ to obtain B_R.

In solving for a reduced C matrix and the D matrix used with the reduced A, B, and C matrices, there is no simple discarding of rows and columns. Starting with equation (23), if $\Delta\dot{x}_2$ is zero, then

$$\Delta x_2 = \hat{A}_{21}\,\Delta\dot{x}_1 - \left(\hat{A}_{21}B_1 + \hat{A}_{22}B_2\right)\Delta u \tag{29}$$

Substituting equations (25) and (27) into equation (29) yields

$$\Delta x_2 = \hat{A}_{21}A_R\,\Delta x_1 + \left(\hat{A}_{21}A_R\hat{A}_{12} - \hat{A}_{22}\right)B_2\,\Delta u \tag{30}$$

The partitioned form of equation (9) is

$$\Delta y = \left[C_1 \mid C_2\right]\frac{\Delta x_1}{\Delta x_2} + D\,\Delta u \tag{31}$$

where

Δy o-dimensioned output deviation vector

Δx_1 r-dimensioned part of state deviation vector

Δx_2 (n-r)-dimensioned part of state deviation vector

Δu c-dimensioned system control input deviation vector

C_1 o- by r-dimensioned part of output matrix

C_2 o- by (n-r)-dimensioned part of output matrix

D o- by c-dimensioned feed-forward matrix

Equation (31) yields

$$\Delta y = C_1 \Delta x_1 + C_2 \Delta x_2 + D \Delta u \tag{32}$$

Using equation (30) to eliminate Δx_2 gives

$$\Delta y = C_1 \Delta x_1 + C_2 \hat{A}_{21} A_R \Delta x_1 + \left[C_2 \left(\hat{A}_{21} A_R \hat{A}_{12} - \hat{A}_{22} \right) B_2 + D \right] \Delta u \tag{33}$$

or

$$\Delta y = C_R \Delta x_1 + D_R \Delta u \tag{34}$$

where

$$C_R = C_1 + C_2 \hat{A}_{21} A_R \tag{35}$$

and

$$D_R = C_2 \left(\hat{A}_{21} A_R \hat{A}_{12} - \hat{A}_{22} \right) B_2 + D \tag{36}$$

USERS GUIDE

Changes to DYNGEN

The IBM 360/67 computer with TSS was chosen because of the storage size of the DYGABCD program. Certain changes have been made in the conversion of DYNGEN to DYGABCD for accuracy and speed of calculation. All real variables are now double precision. Each convergence loop now uses a convergence criterion of TOLALL multiplied by some constant, where TOLALL is input in the namelist data as before. Although the number of passes through some loops has been increased, the relationships among the loops have been kept constant, so that the accuracy of the convergence criteria can be increased. All BLOCK DATA subroutines are combined into one subroutine with the named common COMMON/COMDAT/COMD(5423). All program variables are com-

bined into one named common COMMON/COMALL/COM(1062). Within each subroutine, only those variables actually referenced are equivalenced to either COMD and/or COM, with one exception. In the main routine DYGABCD, each variable referenced anywhere in the program is equivalenced, so that all may be seen at a glance.

The DYGABCD program provides a maximum of 23 possible derivative variables for use in calculating the A and C matrices. The specific number of possible variables for any particular engine depends on the configuration of that engine. In calculating the B matrix, only two control variables are used. More parameters could be chosen. In calculating the C matrix, any variables calculated in the program, including the states and/or inputs, may be used. Variables that are both states and outputs are used in calculating the D matrix.

Minor changes have been made in the main routine DYGABCD and the subroutines ENGBAL and PUTIN (underlined in the listings) to accommodate the addition of the A, B, C, and D matrix calculations. An addition has been made to function DERIV to allow either the original calculations of DYNGEN or the calculations necessary for the A and/or C matrices to be performed. Since DISTRB was already a user-written subroutin in DYNGEN, the bulk of the A, B, C, and D matrix calculations are done in it.

Method

DYGABCD may be run exactly as DYNGEN if DYNGEN data are used. It may also be used to generate the A, B, C, and D matrices when it is used with a different set of data and a new subroutine DISTRB that is modified by the user. To use the A, B, C, and D matrix capability, the following tasks must be performed:

(1) A steady-state point is run and the resulting main fuel flow noted. This is the steady-state point about which the small perturbations (generating A, B, C, and D) will be made. When the A, B, C, and D matrices are generated, they will define a linear model that is applicable in the region of this steady-state point.

(2) A new run is made with the addition of one data card that includes the following parameters: MODE=2, WFB=(the steady-state value noted previously in task 1), ITRAN=1, and IAMTRX=1.

IAMTRX is a new variable that is input as 1 if the A, B, C, and D matrices are to be calculated. The default value is 0, which is the value when DYGABCD is used as DYNGEN. If IAMTRX is 1, the program runs a "pseudo-transient" to calculate the A and C matrices, a set of steady-state points to calculate the B matrix, and a set of steady-state points plus the C matrix to calculate the D matrix.

The term "pseudo-transient" is used to describe the following procedure: since ITRAN = 1 on the data card in task 2, the program runs in the transient mode. However, during every time step, when the time index, NSTEP, is increased by 1 in sub-

routine ENGINE, it is promptly decreased by 1 back to 0 by the next statement, if IAMTRX is 1. This "fool the program" pseudo-transient is necessary to calculate the A and C matrices in subroutine DISTRB. Just before calculating the B and D matrices, which depend on a set of steady-state points, DISTRB sets ITRAN to 0.

Changes to Subroutine DISTRB

DATA statements and program statements in subroutine DISTRB control the calculations of the A, B, C, and D matrices. DYGABCD provides a maximum of 23 possible time-varying state variables. These variables and their original order are as follows:

List number	Variable name
1	XNHP
2	XNIP
3	XNLP
4	P22
5	U22
6	P21
7	U21
8	P3
9	U3
10	P4
11	U4
12	P50
13	U50
14	P5
15	U5
16	P55
17	U55
18	P7
19	U7
20	P24
21	U24
22	P37
23	U37

The DATA statements and program statements include parameters that describe the particular case the user wishes to run.

INV	total number of states possible for particular engine being modeled, less than or equal to 23
INVRED	total number of states actually used in model (full or reduced), less than or equal to INV
IVARB	order in which states are to be done (If this is a reduced model, the states to be included are listed before those not included.)
INB	number of system control inputs desired, less than or equal to 5
BPER	percent (BPER=.02 means 2 percent) (The steady-state value of each input will be multiplied by this percent to form the Δu of each input in order to calculate the B and D matrices.)
INC	number of outputs desired, less than or equal to 50
NUCOM	list of equivalenced "COM" numbers of outputs desired (All of the equivalences are in the main routine DYGABCD.)
PVRDOT	vector of percents (PVRDOT=.02 means 2 percent) in list number order (The steady-state value of each state will be multiplied by the percent associated with that state to form the $\Delta \dot{x}$ for that particular state in order to calculate the A and C matrices.)

There is one other necessary parameter that is set by a program statement - ICHOIC. If ICHOIC is 0, subroutine DISTRB calculates the A and C matrices by using the percentages of the states as set by the initial conditions of the PVRDOT's. If ICHOIC is set to 1, the percentages will be calculated and reset by the program to form an A^{-1} matrix with no zero columns or rows. A run with ICHOIC set to 1 should always be made first to find values for the PVRDOT's that do not result in an A^{-1} matrix containing any zero columns or rows. Obviously, an A^{-1} matrix that has zero columns or rows cannot be inverted.

Subroutine DISTRB is programmed to use inputs WFB (main fuel flow) and A8 (nozzle area). However, more and/or other inputs could be used. The number of inputs is limited to 5, but that limit could be changed by the user.

A line-by-line study of subroutine DISTRB should be made before attempting to make any runs other than the sample case. Subroutine DISTRB includes many comment cards that can be helpful. The listing starts on page 98.

Changes to Function DERIV

Function DERIV has been changed so that, if IAMTRX=1, a different set of statements is used. These statements set all derivatives to zero except for the particular

state being used. The derivative of that state is set to its steady-state value multiplied by the PVRDOT of that state. If IAMTRX=0, function DERIV behaves in the same manner as it did in DYNGEN.

Changes to Subroutine PUTIN

Subroutine PUTIN has been changed so that it will accept one more variable in the namelist DATAIN. This variable is IAMTRX, which is defaulted to zero if DYGABCD is to be used as DYNGEN and is set to 1 if DYGABCD is to be used to generate the A, B, C, and D matrices.

Changes to Subroutine ENGBAL

Two changes have been made to subroutine ENGBAL. One change resets the time index if IAMTRX=1. This has been explained in the section Method. The other change causes subroutines DISTRB and COINLT to be called during the steady-state calculations for the B and D matrices if IAMTRX=1. If IAMTRX=0, subroutine ENGBAL behaves in the same manner as it did in DYNGEN.

Changes to Main Routine DYGABCD

The routine DYGABCD sets the default value of IAMTRX to zero and initializes the parameter IDOT to zero. IDOT is used internally in subroutine DISTRB to keep a record on the states being used to generate the A and C matrices and later to keep a record on the inputs used in the calculations for the B and D matrices.

SAMPLE CASE

The sample case is a two-spool, two-stream engine simulation that has a maximum of 16 states. The system control inputs used were WFB (main fuel flow) and A8 (nozzle area). The outputs specified were SFC (specific fuel consumption), FG (gross thrust), and FN (net thrust).

A steady-state point was run and the main fuel flow was noted to be 2.75 lbm/sec. The data card for the A, B, C, and D matrix calculations was inserted. In subroutine DISTRB, the following parameters were set in DATA statements: the number of states possible for this engine (16), in INV; the number of states for a reduced model (9), in INVRED; the order of states, in IVARB; the number of system control inputs (2), in

INB; the percent delta of the inputs for the calculation of the B and D matrices (0.01), in BPER; the number of outputs (3), in INC; and the list of the equivalenced "COM" numbers of the outputs, in NUCOM. Subroutine statements were as follows: ICHOIC was set to 1 so that subroutine DISTRB would calculate the PVRDOT's. These PVRDOT's were all initialized to 0.02. The A, B, C, and D matrices generated were used to form a linear model.

Two transient runs were made using DYGABCD as DYNGEN. The first run had a 3-percent step change in main fuel flow, and the other had a 3-percent step change in nozzle area. The results from these two full-state, nonlinear transients were the time histories of the states, inputs, and outputs. These nonlinear results (squares on the graphs) were then used as the standard against which the results from the various runs using the linear model (circles on the graphs) were plotted.

Since one of the state eigenvalues was extremely large, 15th order (instead of 16th) was the largest order linear model that could be used to generate time responses. A run was made using the 15th-order (full) linear model with a 3-percent step change in main fuel flow. Another run was made using a second-order (reduced) linear model with the same step change. These two runs were then repeated with a 3-percent step change in nozzle area.

The responses of the states and outputs from these four linear cases are compared with the responses from the nonlinear cases in figures 1 to 19. The results of the full-order linear runs and the nonlinear runs for the two step inputs are shown in the upper quadrants of each figure. The agreement is very good. The reduced (low order) linear runs and the nonlinear runs for the two step inputs are shown in the lower quadrants of each figure. Here, there is disagreement, especially in the first 0.1 second of the transient, because those parameters that cause the high-frequency dynamics are missing.

The FORTRAN symbols used are defined in appendix A. The program is listed in appendix B. Sample case input data and output are listed in appendix C.

CONCLUSIONS

DYGABCD is a computer program that produces the A, B, C, and D matrices from a nonlinear, generalized, transient engine simulation. These matrices are used to generate a linear model. The results from the nonlinear simulations and the full-order linear simulations show good agreement. This method also permits reduced-order

models to be formed that produce varying degrees of agreement. The method of calculating these matrices could be applied to many other dynamic engine simulations.

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, April 21, 1978,
505-05.

APPENDIX A

FORTRAN SYMBOLS

An asterisk denotes that the variable can be input.

	A24	cross-sectional area at station 24, m^2 (ft^2)
	A25	cross-sectional area at station 25, m^2 (ft^2)
*	A28	area of fan duct nozzle throat, station 28, m^2 (ft^2)
	A28SAV	saved area of fan duct nozzle throat, station 28, at design conditions, m^2 (ft^2)
	A29	cross-sectional area at station 29, m^2 (ft^2)
	A29SAV	saved cross-sectional area at station 29 at design conditions, m^2 (ft^2)
*	A38	area of wing duct nozzle throat, station 38, m^2 (ft^2)
	A39	cross-sectional area at station 39, m^2 (ft^2)
	A55	cross-sectional area at station 55, m^2 (ft^2)
*	A6	cross-sectional area of afterburner entrance, station 6, calculated from AM6, m^2 (ft^2)
	A7	cross-sectional area at station 7, m^2 (ft^2)
*	A8	area of main nozzle throat, station 8, (can be changed at off-design), m^2 (ft^2)
	A8SAV	saved area of main nozzle throat, station 8, at design conditions, m^2 (ft^2)
	A9	cross-sectional area at station 9, m^2 (ft^2)
	A9SAV	saved cross-sectional area at station 9 at design conditions, m^2 (ft^2)
*	AFTFAN	(logical) control for an aft-fan engine
*	ALTP	altitude of aircraft, m (ft)

*	AM	Mach number of aircraft
*	AM23	Input: design Mach number of ductburner entrance, station 23 Output: Mach number at station 23
	AM25	Mach number at station 25
	AM28	Mach number at station 28
	AM29	Mach number at station 29
	AM38	Mach number at station 38
	AM39	Mach number at station 39
*	AM55	Input: design Mach number at low-pressure-turbine exit, station 55 Output: Mach number at station 55
*	AM6	Input: design Mach number at afterburner entrance, station 6 Output: Mach number at station 6
	AM6DSV	saved Mach number at afterburner entrance at design conditions, station 6
	AM7	Mach number at station 7
	AM8	Mach number at station 8
	AM9	Mach number at station 9
	BLC	bleed flow out of compressor, kg/sec (lbm/sec)
	BLDU	bleed flow into fan duct, kg/sec (lbm/sec)
	BLF	bleed flow out of fan (dumped overboard), kg/sec (lbm/sec)
	BLHP	bleed flow into high-pressure turbine, kg/sec (lbm/sec)
	BLI	airflow into third stream, kg/sec (lbm/sec)
	BLIP	bleed flow into intermediate-pressure turbine, kg/sec (lbm/sec)
	BLLP	bleed flow into low-pressure turbine, kg/sec (lbm/sec)

	BLOB	bleed flow lost overboard (customer bleed), kg/sec (lbm/sec)
	BPRINT	ratio of airflow into wing duct to airflow into core
	BYPASS	ratio of airflow into fan duct to airflow into intermediate compressor
	CNC	corrected shaft speed of core compressor
	CNF	corrected shaft speed of fan compressor
	CNHP	corrected shaft speed of high-pressure turbine
	CNHPCF	correction factor of high-pressure-turbine corrected speed
*	CNHPDS	design corrected speed of high-pressure turbine
	CNI	corrected shaft speed of intermediate compressor
	CNIP	corrected shaft speed of intermediate-pressure turbine
	CNIPCF	correction factor of intermediate-pressure-turbine corrected speed
*	CNIPDS	design corrected speed of intermediate-pressure turbine
	CNLP	corrected shaft speed of low-pressure turbine
	CNLPCF	correction factor of low-pressure turbine corrected speed
*	CNLPDS	design corrected speed of low-pressure turbine
	CS	ambient speed of sound, m/sec (ft/sec)
*	CVDNOZ	velocity coefficient of duct nozzle thrust
*	CVDWNG	velocity coefficient of wing nozzle thrust
*	CVMNOZ	velocity coefficient of core nozzle thrust
*	DELFG	gross-thrust delta degradation multiplier
*	DELFN	net-thrust delta degradation multiplier

* DELSFC specific-fuel-consumption delta degradation multiplier

* DELT1 correction to standard-day temperature, K (^{o}R)

DHHPCF Δ enthalpy correction factor of high-pressure turbine

DHIPCF Δ enthalpy correction factor of intermediate-pressure turbine

DHLPCF Δ enthalpy correction factor of low-pressure turbine

DHTC work done by high-pressure turbine, J/kg (Btu/lbm)

DHTCHP enthalpy change of high-pressure turbine, temperature corrected, J/kg-K (Btu/lbm-^{o}R)

DHTCIP enthalpy change of intermediate-pressure turbine, temperature corrected, J/kg-K (Btu/lbm-^{o}R)

DHTCLP enthalpy change of low-pressure turbine, temperature corrected, J/kg-K (Btu/lbm-^{o}R)

DHTF work done by low-pressure turbine, J/kg (Btu/lbm)

DHTI work done by intermediate-pressure turbine, J/kg (Btu/lbm)

* DPAFDS design pressure drop (ΔP/P) of afterburner

DPAFT pressure drop (ΔP/P) of afterburner

* DPCODS design pressure drop (ΔP/P) of combustor

DPCOM pressure drop (ΔP/P) of combustor

DPDUC pressure drop (ΔP/P) of fan duct

* DPDUDS design pressure drop (ΔP/P) of fan duct

* DPWGDS design pressure drop (ΔP/P) of wing duct

DPWING pressure drop (ΔP/P) of wing duct

* DT solution time step for transient, sec

* DTPRNT time step for output listings, sec

DUMD1(15) dummy variables

*	DUMSPL	(logical) control for spool that does not change temperature or pressure of air
	ERRER	(logical) test of exceeding ITRYS
*	ETAA	efficiency of afterburner
*	ETAADS	design efficiency of afterburner
	ETAASV	saved efficiency of afterburner at design conditions
	ETAB	efficiency of combustor
	ETABCF	correction factor of combustor efficiency
*	ETABDS	design efficiency of combustor
	ETAC	adiabatic efficiency of core compressor
	ETACCF	correction factor of core-compressor efficiency
*	ETACDS	design adiabatic efficiency of core compressor
*	ETAD	efficiency of ductburner
	ETAF	adiabatic efficiency of fan compressor
	ETAFCF	correction factor of fan-compressor efficiency
*	ETAFDS	design adiabatic efficiency of fan compressor
	ETAI	adiabatic efficiency of intermediate compressor
	ETAICF	correction factor of intermediate-compressor efficiency
*	ETAIDS	design adiabatic efficiency of intermediate compressor
*	ETAR	pressure recovery of inlet (ram recovery), P2/P1
	ETATHP	adiabatic efficiency of high-pressure turbine
	ETATIP	adiabatic efficiency of intermediate-pressure turbine
	ETATLP	adiabatic efficiency of low-pressure turbine

	ETHPCF	correction factor of high-pressure-turbine efficiency
*	ETHPDS	design adiabatic efficiency of high-pressure turbine
	ETIPCF	correction factor of intermediate-pressure-turbine efficiency
*	ETIPDS	design adiabatic efficiency of intermediate-pressure turbine
	ETLPCF	correction factor of low-pressure-turbine efficiency
*	ETLPDS	design adiabatic efficiency of low-pressure turbine
*	FAN	(logical) control that indicates fan or turbojet
	FAR24	fuel-to-air ratio at station 24
	FAR4	fuel-to-air ratio at station 4
	FAR5	fuel-to-air ratio at station 5
	FAR50	fuel-to-air ratio at station 50
	FAR55	fuel-to-air ratio at station 55
	FAR7	fuel-to-air ratio at station 7
	FAR7SV	saved fuel-to-air ratio at station 7 at design conditions
	FCOVFN	ratio of core thrust to net thrust
	FFOVFN	ratio of fan thrust to net thrust
	FG	gross thrust, N (lbf)
	FGM	momentum thrust of all but wing, N (lbf)
	FGMD	fan duct momentum thrust of all but wing, N (lbf)
	FGMM	core nozzle momentum thrust of all but wing, N (lbf)
	FGMWNG	momentum thrust of wing, N (lbf)

	FGP	pressure thrust of all but wing, N (lbf)
	FGPD	fan duct pressure thrust of all but wing, N (lbf)
	FGPM	core nozzle pressure thrust of all but wing, N (lbf)
	FGPWNG	pressure thrust of wing, N (lbf)
	FMNOFN	ratio of fan plus core thrust to net thrust
	FN	net thrust, N (lbf)
	FNMAIN	net thrust of all but wing, N (lbf)
	FNOVFD	ratio of net thrust to design-point net thrust
	FNWING	net thrust of wing, N (lbf)
	FRD	ram drag, N (lbf)
	FWOVFN	ratio of net wing thrust to net thrust
*	FXFN2M	(logical) control for boosted fan
*	FXM2CP	(logical) control for supercharged compressor
	H1	enthalpy at station 1, J/kg (Btu/lbm)
	H2	enthalpy at station 2, J/kg (Btu/lbm)
	H21	enthalpy at station 21, J/kg (Btu/lbm)
	H22	enthalpy at station 22, J/kg (Btu/lbm)
	H23	enthalpy at station 23, J/kg (Btu/lbm)
	H24	enthalpy at station 24, J/kg (Btu/lbm)
	H25	enthalpy at station 25, J/kg (Btu/lbm)
	H28	enthalpy at station 28, J/kg (Btu/lbm)
	H29	enthalpy at station 29, J/kg (Btu/lbm)
	H3	enthalpy at station 3, J/kg (Btu/lbm)
	H38	enthalpy at station 38, J/kg (Btu/lbm)
	H39	enthalpy at station 39, J/kg (Btu/lbm)

	H4	enthalpy at station 4, J/kg (Btu/lbm)
	H5	enthalpy at station 5, J/kg (Btu/lbm)
	H50	enthalpy at station 50, J/kg (Btu/lbm)
	H55	enthalpy at station 55, J/kg (Btu/lbm)
	H6	enthalpy at station 6, J/kg (Btu/lbm)
	H7	enthalpy at station 7, J/kg (Btu/lbm)
	H8	enthalpy at station 8, J/kg (Btu/lbm)
	H9	enthalpy at station 9, J/kg (Btu/lbm)
*	HPEXT	power extracted, W (hp)
*	IAFTBN	index for afterburning desired; zeroes automatically; values of 0, 1, or 2
*	IAMTP	index for ram or inlet operation desired; values of 0 to 5
*	IAMTRX	index that indicates if A,B,C, and D matrices are to be calculated
	ICOAFB	index of error in subroutine COAFBN
	ICODUC	index of error in subroutine CODUCT
	ICOMIX	index of error in subroutine COMIX
*	IDBURN	index for duct burning desired; zeroes automatically; values of 0, 1, or 2
*	IDCD	duct nozzle convergent-divergent when value=1; convergent when value=0
*	IDES	index for design point; zeroes automatically; must be set to 1 to design engine
*	IDUMP	index for dumping of error matrix; values of 0, 1, or 2
*	IGASMX	index for mixed or no-mixed flow turbofans; values of -1, 0, 1, or 2
*	IMCD	main nozzle convergent-divergent when value=1; convergent when value=0

*	INIT	index for initializing guesses; zeroes automatically; 0 calls GUESS, 1 does not
*	ISPOOL	number of engine rotors
*	ITRAN	index for initiating transient
*	ITRYS	maximum number of iterations allowed through engine counter
	JTRAN	index that indicates a transient is in progress
	LOOPER	number of loops through engine counter
*	MODE	independent variable designator for engine operation
*	NOZFLT	index for floating main or duct nozzle; zeroes automatically; values of 0 to 3
	P1	standard pressure, N/m^2 (atm)
*	P2	total pressure at station 2, N/m^2 (atm)
	P21	total pressure at station 21, N/m^2 (atm)
	P22	total pressure at station 22, N/m^2 (atm)
	P23	total pressure at station 23, N/m^2 (atm)
	P24	total pressure at station 24, N/m^2 (atm)
	P25	total pressure at station 25, N/m^2 (atm)
	P28	total pressure at station 28, N/m^2 (atm)
	P29	total pressure at station 29, N/m^2 (atm)
	P3	total pressure at station 3, N/m^2 (atm)
	P37	total pressure at station 37, N/m^2 (atm)
	P38	total pressure at station 38, N/m^2 (atm)
	P39	total pressure at station 39, N/m^2 (atm)
	P4	total pressure at station 4, N/m^2 (atm)
	P5	total pressure at station 5, N/m^2 (atm)

	P50	total pressure at station 50, N/m^2 (atm)
	P55	total pressure at station 55, N/m^2 (atm)
	P6	total pressure at station 6, N/m^2 (atm)
	P6DSAV	saved total pressure at station 6 at design conditions, N/m^2 (atm)
	P7	total pressure at station 7, N/m^2 (atm)
	P8	total pressure at station 8, N/m^2 (atm)
	P9	total pressure at station 9, N/m^2 (atm)
*	PCBLC	ratio of compressor bleed to cool turbines to total compressor airflow
*	PCBLDU	ratio of compressor bleed leaked into fan duct to total compressor bleed flow
*	PCBLF	ratio of bleed from fan compressor to fan airflow dumped overboard (i.e., leakage)
*	PCBLHP	fraction of PCBLC used in high-pressure turbine for cooling
*	PCBLI	fraction of intermediate compressor air that goes into third stream
*	PCBLID	ratio of design value of air into wing to air into core; zero for two-stream engine
*	PCBLIP	fraction of PCBLC used in intermediate-pressure turbine for cooling
*	PCBLLP	fraction of PCBLC used in low-pressure turbine for cooling
*	PCBLOB	fraction of bleed air out of compressor lost overboard (for customer use)
*	PCNC	shaft speed of core compressor as a percent of design
*	PCNCDS	design corrected speed of core compressor as a percent of design
	PCNCGU	guessed shaft speed of core compressor as a percent of design

*	PCNF	shaft speed of fan compressor as a percent of design
*	PCNFDS	design corrected speed of fan compressor as a percent of design
	PCNFGU	guessed shaft speed of fan compressor as a percent of design
*	PCNI	shaft speed of intermediate compressor as a percent of design
*	PCNIDS	design corrected speed of intermediate compressor as a percent of design
	PCNIGU	guessed shaft speed of intermediate compressor as a percent of design
*	PMIHP	polar moment of inertia of high-pressure rotor, kg-m^2 (slug-ft^2)
*	PMIIP	polar moment of inertia of intermediate-pressure rotor, kg-m^2 (slug-ft^2)
*	PMILP	polar moment of inertia of low-pressure rotor, kg-m^2 (slug-ft^2)
	PRC	pressure ratio of core compressor
	PRCCF	correction factor of core-compressor pressure ratio
*	PRCDS	design pressure ratio of core compressor
	PRF	pressure ratio of fan compressor
	PRFCF	correction factor of fan-compressor pressure ratio
*	PRFDS	design pressure ratio of fan compressor
	PRI	pressure ratio of intermediate compressor
	PRICF	correction factor of intermediate-compressor pressure ratio
*	PRIDS	design pressure ratio of intermediate compressor
	PS28	static pressure at station 28, N/m^2 (atm)
	PS29	static pressure at station 29, N/m^2 (atm)

	PS38	static pressure at station 38, N/m^2 (atm)
	PS39	static pressure at station 39, N/m^2 (atm)
*	PS55	Input: static pressure at low-pressure-turbine exit, station 55, N/m^2 (atm) Output: static pressure at station 55, N/m^2 (atm)
	PS6	static pressure at station 6, N/m^2 (atm)
	PS7	static pressure at station 7, N/m^2 (atm)
	PS8	static pressure at station 8, N/m^2 (atm)
	PS9	static pressure at station 9, N/m^2 (atm)
	S1	entropy at station 1, J/kg-K (Btu/lbm-^oR)
	S2	entropy at station 2, J/kg-K (Btu/lbm-^oR)
	S21	entropy at station 21, J/kg-K (Btu/lbm-^oR)
	S22	entropy at station 22, J/kg-K (Btu/lbm-^oR)
	S23	entropy at station 23, J/kg-K (Btu/lbm-^oR)
	S24	entropy at station 24, J/kg-K (Btu/lbm-^oR)
	S25	entropy at station 25, J/kg-K (Btu/lbm-^oR)
	S28	entropy at station 28, J/kg-K (Btu/lbm-^oR)
	S29	entropy at station 29, J/kg-K (Btu/lbm-^oR)
	S3	entropy at station 3, J/kg-K (Btu/lbm-^oR)
	S4	entropy at station 4, J/kg-K (Btu/lbm-^oR)
	S5	entropy at station 5, J/kg-K (Btu/lbm-^oR)
	S50	entropy at station 50, J/kg-K (Btu/lbm-^oR)
	S55	entropy at station 55, J/kg-K (Btu/lbm-^oR)
	S6	entropy at station 6, J/kg-K (Btu/lbm-^oR)
	S7	entropy at station 7, J/kg-K (Btu/lbm-^oR)
	S8	entropy at station 8, J/kg-K (Btu/lbm-^oR)
	S9	entropy at station 9, J/kg-K (Btu/lbm-^oR)

	SFC	specific fuel consumption, kg/N-hr (lbm/lbf-hr)
*	SI	(logical) control for SI or U.S. customary (English) units
	T1	standard temperature, station 1, K (^{o}R)
*	T2	Input: total temperature at fan inlet, station 2, K (^{o}R) Output: total temperature at station 2, K (^{o}R)
	T21	total temperature at station 21, K (^{o}R)
	T21DS	total temperature at intermediate-compressor exit at design conditions, station 21, K (^{o}R)
	T22	total temperature at station 22, K (^{o}R)
	T22DS	design total temperature at fan exit, station 22, K (^{o}R)
	T23	total temperature at station 23, K (^{o}R)
*	T24	Input: total temperature at ductburner exit, station 24, K (^{o}R) Output: total temperature at station 24, K (^{o}R)
	T25	total temperature at station 25, K (^{o}R)
	T28	total temperature at station 28, K (^{o}R)
	T29	total temperature at station 29, K (^{o}R)
	T2DS	design total temperature at fan inlet, station 2, K (^{o}R)
	T3	total temperature at station 3, K (^{o}R)
	T38	total temperature at station 38, K (^{o}R)
	T39	total temperature at station 39, K (^{o}R)
*	T4	Input: total temperature at combustor exit, station 4, K (^{o}R) Output: total temperature at station 4, K (^{o}R)
*	T4DS	design total temperature at combustor exit, station 4, K (^{o}R)
	T4GU	guessed total temperature at station 4, K (^{o}R)

	T5	total temperature at station 5, K (^{o}R)
	T50	total temperature at station 50, K (^{o}R)
	T55	total temperature at station 55, K (^{o}R)
	T6	total temperature at station 6, K (^{o}R)
*	T7	Input: total temperature at afterburner exit, station 7, K (^{o}R) Output: total temperature at station 7, K (^{o}R)
*	T7DS	design total temperature at afterburner exit, station 7, K (^{o}R)
	T8	total temperature at station 8, K (^{o}R)
	T9	total temperature at station 9, K (^{o}R)
*	TF	final time for transient, sec
	TFAR	total fuel-to-air ratio
	TFFHP	flow function of high-pressure turbine, kg-$\sqrt{K}$-m^2/N-sec (lbm-$\sqrt{^{o}R}$-in^2/lbf-sec)
	TFFIP	flow function of intermediate-pressure turbine, kg-$\sqrt{K}$-m^2/N-sec (lbm-$\sqrt{^{o}R}$-in^2/lbf-sec)
	TFFLP	flow function of low-pressure turbine, kg-$\sqrt{K}$-m^2/N-sec (lbm-$\sqrt{^{o}R}$-in^2/lbf-sec)
	TFHPCF	correction factor of high-pressure turbine flow function
*	TFHPDS	design flow function of high-pressure turbine, kg-$\sqrt{K}$-m^2/N-sec (lbm-$\sqrt{^{o}R}$-in^2/lbf-sec)
	TFIPCF	correction factor of intermediate-pressure turbine flow function
*	TFIPDS	design flow function of intermediate-pressure turbine, kg-$\sqrt{K}$-m^2/N-sec (lbm-$\sqrt{^{o}R}$-in^2/lbf-sec)
	TFLPCF	correction factor of low-pressure turbine flow function
*	TFLPDS	design flow function of low-pressure turbine, kg-$\sqrt{K}$-m^2/N-sec (lbm-$\sqrt{^{o}R}$-in^2/lbf-sec)

	TIME	time, sec
*	TOLALL	tolerance on convergence
	TS28	static temperature at station 28, K (°R)
	TS29	static temperature at station 29, K (°R)
	TS38	static temperature at station 38, K (°R)
	TS39	static temperature at station 39, K (°R)
	TS7	static temperature at station 7, K (°R)
	TS8	static temperature at station 8, K (°R)
	TS9	static temperature at station 9, K (°R)
	U21	internal energy at station 21, J/kg (Btu/lbm)
	U22	internal energy at station 22, J/kg (Btu/lbm)
	U24	internal energy at station 24, J/kg (Btu/lbm)
	U3	internal energy at station 3, J/kg (Btu/lbm)
	U37	internal energy at station 37, J/kg (Btu/lbm)
	U4	internal energy at station 4, J/kg (Btu/lbm)
	U5	internal energy at station 5, J/kg (Btu/lbm)
	U50	internal energy at station 50, J/kg (Btu/lbm)
	U55	internal energy at station 55, J/kg (Btu/lbm)
	U7	internal energy at station 7, J/kg (Btu/lbm)
	V25	velocity at station 25, m/sec (ft/sec)
	V28	velocity at station 28, m/sec (ft/sec)
	V29	velocity at station 29, m/sec (ft/sec)
	V38	velocity at station 38, m/sec (ft/sec)
	V39	velocity at station 39, m/sec (ft/sec)
	V55	velocity at station 55, m/sec (ft/sec)
	V6	velocity at station 6, m/sec (ft/sec)

	V7	velocity at station 7, m/sec (ft/sec)
	V8	velocity at station 8, m/sec (ft/sec)
	V9	velocity at station 9, m/sec (ft/sec)
	VA	velocity of aircraft, m/sec (ft/sec)
*	VAFTBN	control volume associated with afterburner, m^3 (ft^3)
*	VCOMB	control volume associated with combustor, m^3 (ft^3)
*	VCOMP	control volume associated with core compressor, m^3 (ft^3)
*	VFAN	control volume associated with fan compressor, m^3 (ft^3)
*	VFDUCT	control volume associated with fan duct, m^3 (ft^3)
*	VHPTRB	control volume associated with high-pressure turbine, m^3 (ft^3)
*	VINTC	control volume associated with intermediate compressor, m^3 (ft^3)
*	VIPTRB	control volume associated with intermediate-pressure turbine, m^3 (ft^3)
	VJD	velocity of fan duct exhaust, m/sec (ft/sec)
	VJM	velocity of core exhaust, m/sec (ft/sec)
	VJW	velocity of wing duct exhaust, m/sec (ft/sec)
*	VLPTRB	control volume associated with low-pressure turbine, m^3 (ft^3)
*	VWDUCT	control volume associated with wing duct, m^3 (ft^3)
	WA21	airflow at station 21, kg/sec (lbm/sec)
	WA22	airflow at station 22, kg/sec (lbm/sec)
	WA23DS	airflow at station 23 at design conditions, kg/sec (lbm/sec)
	WA3	airflow at station 3, kg/sec (lbm/sec)

	WA32	airflow at station 32, kg/sec (lbm/sec)
	WA32DS	airflow at station 32 at design conditions, kg/sec (lbm/sec)
	WA3CDS	corrected airflow in combustor at design conditions, station 3, kg/sec (lbm/sec)
	WAC	airflow of core compressor, kg/sec (lbm/sec)
	WACC	corrected airflow of core compressor, kg/sec (lbm/sec)
*	WACCDS	design corrected airflow of core compressor, kg/sec (lbm/sec)
	WACCF	correction factor of core-compressor corrected airflow
	WACDS	airflow of core compressor at design conditions, kg/sec (lbm/sec)
	WACI	corrected airflow of intermediate compressor, kg/sec (lbm/sec)
	WACP	saved airflow of core compressor, kg/sec (lbm/sec)
	WAD	airflow of fan duct, kg/sec (lbm/sec)
	WAF	airflow of fan compressor, kg/sec (lbm/sec)
	WAFC	corrected airflow of fan compressor, kg/sec (lbm/sec)
*	WAFCDS	design corrected airflow of fan compressor, kg/sec (lbm/sec)
	WAFCF	correction factor of fan compressor corrected airflow
	WAFDS	airflow of fan compressor at design conditions, kg/sec (lbm/sec)
	WAFP	saved airflow of fan compressor, kg/sec (lbm/sec)
	WAI	airflow of intermediate compressor, kg/sec (lbm/sec)

*	WAICDS	design corrected airflow of intermediate compressor, kg/sec (lbm/sec)
	WAICF	correction factor of intermediate-compressor corrected airflow
	WAIDS	airflow of intermediate compressor at design conditions, kg/sec (lbm/sec)
	WAIP	saved airflow of intermediate compressor, kg/sec (lbm/sec)
*	WFA	fuel flow rate to afterburner (IAFTBN=2 only for input), kg/sec (lbm/sec)
*	WFB	fuel flow rate to main combustor (MODE=2 only for input), kg/sec (lbm/sec)
*	WFBDS	design fuel flow rate to main combustor (MODE=2 only for input), kg/sec (lbm/sec)
	WFD	fuel flow rate to ductburner, kg/sec (lbm/sec)
	WFT	total fuel flow rate, kg/sec (lbm/sec)
	WG24	gas flow at station 24, kg/sec (lbm/sec)
	WG37	gas flow at station 37, kg/sec (lbm/sec)
	WG4	gas flow at station 4, kg/sec (lbm/sec)
	WG5	gas flow at station 5, kg/sec (lbm/sec)
	WG50	gas flow at station 50, kg/sec (lbm/sec)
	WG55	gas flow at station 55, kg/sec (lbm/sec)
	WG6	gas flow at station 6, kg/sec (lbm/sec)
	WG6CDS	corrected gas flow at station 6 at design conditions, kg/sec (lbm/sec)
	WG7	gas flow at station 7, kg/sec (lbm/sec)
	WGT	total gas flow rate, kg/sec (lbm/sec)
	XBLDU	saved BLDU
	XBLF	saved BLF

	XFAR24	saved FAR24
	XFAR55	saved FAR55
	XH21	saved H21
	XH25	saved H25
	XH3	saved H3
	XH55	saved H55
	XNHP	speed of core compressor, rpm
*	XNHPDS	design speed of high-pressure rotor, rpm
	XNIP	speed of intermediate compressor, rpm
*	XNIPDS	design speed of intermediate-pressure rotor, rpm
	XNLDEM	commanded speed of fan compressor, rpm
	XNLP	speed of fan compressor, rpm
*	XNLPDS	design speed of low-pressure rotor, rpm
	XP1	saved P1
	XP21	saved P21
	XP25	saved P25
	XP55	saved P55
	XS21	saved S21
	XS25	saved S25
	XS55	saved S55
	XT21	saved T21
	XT25	saved T25
	XT55	saved T55
	XWAC	saved WAC
	XWAF	saved WAF
	XWFB	saved WFB

	XWFD	saved WFD
	XWG24	saved WG24
	XWG55	saved WG55
	XXP1	saved P1
	ZC	ratio of core-compressor pressure ratios
*	ZCDS	design ratio of core compressor; equals pressure ratio at design point on design speed line minus value of pressure ratio at lowest point on speed line, divided by high (surge) value minus low value of pressure ratio on design speed line
	ZF	ratio of fan-compressor pressure ratios
*	ZFDS	design ratio of fan compressor (see ZCDS for explanation)
	ZI	ratio of intermediate-compressor pressure ratios
*	ZIDS	design ratio of intermediate compressor (see ZCDS for explanation)

APPENDIX B

FORTRAN LISTINGS

Main Program DYGABCD

```
C     THIS IS THE MAIN PROGRAM                                                  ABCD0001
      IMPLICIT  REAL*8  (A-H,O-Z)                                               ABCD0002
      LOGICAL  SI, ERRER, DUMSPL, FXFN2M, FXM2CP, AFTFAN, FAN                   ABCD0003
      COMMON /COMALL/  COM(1062)                                                ABCD0004
      DIMENSION  WORD(2), ERR(9), DUMD1(15), FO(50,4), SO(10,6),                ABCD0005
     1 PDATA(5,50), TIMEPT(50), XS(23), PVRDOT(23)                              ABCD0006
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (JDES, COM(4)),           ABCD0007
     1 (KDES, COM(5)), (MODE, COM(6)), (INIT, COM(7)), (IDUMP, COM(8)),         ABCD0008
     2 (IAMTP, COM(9)), (IGASMX, COM(10)), (IDBURN, COM(11)),                   ABCD0009
     3 (IAFTBN, COM(12)), (IDCD, COM(13)), (IMCD, COM(14)),                     ABCD0010
     4 (IDSHOC, COM(15)), (IMSHOC, COM(16)), (NOZFLT, COM(17)),                 ABCD0011
     5 (ITRYS, COM(18)), (LOOPER, COM(19)), (NOMAP, COM(20)),                   ABCD0012
     6 (NUMMAP, COM(21)), (MAPEDG, COM(22)), (TOLALL, COM(23)),                 ABCD0013
     7 (ERR(1), COM(24)), (P1, COM(33)), (H22, COM(34)),                        ABCD0014
     8 (AM23, COM(35)), (WA23DS, COM(36)), (T23, COM(37)),                      ABCD0015
     9 (P23, COM(38)), (H23, COM(39)), (S23, COM(40)), (A24, COM(41)),          ABCD0016
     1 (T24, COM(42)), (H24, COM(43)), (S24, COM(44)), (AM25, COM(45)),         ABCD0017
     2 (T25, COM(46)), (P25, COM(47)), (H25, COM(48)), (S25, COM(49)),          ABCD0018
     3 (A28, COM(50)), (A28SAV, COM(51)), (AM28, COM(52)),                      ABCD0019
     4 (V28, COM(53)), (TS28, COM(54)), (PS28, COM(55)), (T28, COM(56)),        ABCD0020
     5 (P28, COM(57)), (H28, COM(58)), (S28, COM(59)), (A29, COM(60)),          ABCD0021
     6 (A29SAV, COM(61)), (AM29, COM(62)), (V29, COM(63)),                      ABCD0022
     7 (TS29, COM(64)), (PS29, COM(65)), (T29, COM(66)), (P29, COM(67)),        ABCD0023
     8 (H29, COM(68)), (S29, COM(69)), (BYPASS, COM(70)),                       ABCD0024
     9 (WAD, COM(71)), (WFD, COM(72)), (ETAD, COM(73)),                         ABCD0025
     1 (DPDUC, COM(74)), (DPDUDS, COM(75)), (XXP1, COM(76))                     ABCD0026
      EQUIVALENCE  (XWG24, COM(77)), (XFAR24, COM(78)), (XT25, COM(79)),        ABCD0027
     1 (XP25, COM(80)), (XH25, COM(81)), (XS25, COM(82)),                       ABCD0028
     2 (XWG55, COM(83)), (XFAR55, COM(84)), (XT55, COM(85)),                    ABCD0029
     3 (XP55, COM(86)), (XH55, COM(87)), (XS55, COM(88)),                       ABCD0030
     4 (XWFB, COM(89)), (XWFD, COM(90)), (T2DS, COM(91)), (T2, COM(92)),        ABCD0031
     5 (P2, COM(93)), (H2, COM(94)), (S2, COM(95)), (S22, COM(96)),             ABCD0032
     6 (T22DS, COM(97)), (A25, COM(98)), (V25, COM(99)),                        ABCD0033
     7 (T4GU, COM(100)), (T4DS, COM(101)), (T5, COM(102)),                      ABCD0034
     8 (H5, COM(103)), (S5, COM(104)), (WG5, COM(105)),                         ABCD0035
     9 (FAR5, COM(106)), (AM55, COM(107)), (V55, COM(108)),                     ABCD0036
     1 (A55, COM(109)), (PS55, COM(110)), (S6, COM(111)),                       ABCD0037
     2 (PS6, COM(112)), (V6, COM(113)), (TFHPCF, COM(114)),                     ABCD0038
     3 (CNHPCF, COM(115)), (ETHPCF, COM(116)), (DHHPCF, COM(117)),              ABCD0039
     4 (TFHPDS, COM(118)), (CNHPDS, COM(119)), (ETHPDS, COM(120)),              ABCD0040
     5 (PRFCF, COM(121)), (ETAFCF, COM(122)), (WAFCF, COM(123)),                ABCD0041
     6 (PCNFDS, COM(124)), (PRFDS, COM(125)), (ETAFDS, COM(126)),               ABCD0042
     7 (WAFDS, COM(127)), (PCNCGU, COM(128)), (HPEXT, COM(129)),                ABCD0043
     8 (WACCDS, COM(130)), (PPF, COM(131)), (ETAF, COM(132)),                   ABCD0044
     9 (ZCDS, COM(133)), (CNF, COM(134)), (WAFC, COM(135)),                     ABCD0045
     1 (ZF, COM(136)), (PCNF, COM(137)), (PCBLF, COM(138))                      ABCD0046
      EQUIVALENCE  (ZI, COM(139)), (PCNI, COM(140)), (TFFIP, COM(141)),         ABCD0047
     1 (CNIP, COM(142)), (ETATIP, COM(143)), (DHTCIP, COM(144)),                ABCD0048
     2 (DHTI, COM(145)), (ZIDS, COM(146)), (PCNIDS, COM(147)),                  ABCD0049
     3 (PCNIGU, COM(148)), (T1, COM(149)), (H1, COM(150)),                      ABCD0050
     4 (S1, COM(151)), (T3, COM(152)), (H3, COM(153)), (WA3, COM(154)),         ABCD0051
     5 (WA3CDS, COM(155)), (T4, COM(156)), (H4, COM(157)),                      ABCD0052
     6 (S4, COM(158)), (WG4, COM(159)), (FAR4, COM(160)),                       ABCD0053
     7 (T50, COM(161)), (H50, COM(162)), (S50, COM(163)),                       ABCD0054
```

```
8 (WG50, COM(164)), (FAR50, COM(165)), (PCBLHP, COM(166)),          ABCD0055
9 (PCBLIP, COM(167)), (PCBLLP, COM(168)), (PCBLDU, COM(169)),       ABCD0056
1 (PCBLOB, COM(170)), (CNHP, COM(171)), (ETATHP, COM(172)),         ABCD0057
2 (DHTCHP, COM(173)), (DHTC, COM(174)), (TFFHP, COM(175)),          ABCD0058
3 (PRICF, COM(176)), (ETAICF, COM(177)), (WAICF, COM(178)),         ABCD0059
4 (ETABCF, COM(179)), (PRIDS, COM(180)), (ETAIDS, COM(181)),        ABCD0060
5 (WAIDS, COM(182)), (WAICDS, COM(183)), (ETABDS, COM(184)),        ABCD0061
6 (WFBDS, COM(185)), (ZFDS, COM(186)), (ETAR, COM(187)),            ABCD0062
7 (ETAI, COM(188)), (PRI, COM(189)), (ETAB, COM(190)),              ABCD0063
8 (WAC, COM(191)), (WFB, COM(192)), (BLC, COM(193)),                ABCD0064
9 (CS, COM(194)), (AM, COM(195)), (ALTP, COM(196)),                 ABCD0065
1 (DPCODS, COM(197)), (DPCOM, COM(198)), (PCNFGU, COM(199))         ABCD0066
  EQUIVALENCE  (XP1, COM(200)), (XT21, COM(201)), (XP21, COM(202)), ABCD0067
1 (XH21, COM(203)), (XS21, COM(204)), (XH3, COM(205)),              ABCD0068
2 (XWAF, COM(206)), (XWAC, COM(207)), (XBLF, COM(208)),             ABCD0069
3 (XBLDU, COM(209)), (WG37, COM(210)), (A38, COM(211)),             ABCD0070
4 (AM38, COM(212)), (V38, COM(213)), (T38, COM(214)),               ABCD0071
5 (H38, COM(215)), (P38, COM(216)), (TS38, COM(217)),               ABCD0072
6 (PS38, COM(218)), (A39, COM(219)), (AM39, COM(220)),              ABCD0073
7 (V39, COM(221)), (T39, COM(222)), (H39, COM(223)),                ABCD0074
8 (P39, COM(224)), (TS39, COM(225)), (PS39, COM(226)),              ABCD0075
9 (WA32DS, COM(227)), (DPWING, COM(228)), (BPRINT, COM(229)),       ABCD0076
1 (FGMWNG, COM(230)), (FGPWNG, COM(231)), (FNWING, COM(232)),       ABCD0077
2 (FNMAIN, COM(233)), (FWOVFN, COM(234)), (DPWGDS, COM(235)),       ABCD0078
3 (DELFG, COM(236)), (DELFN, COM(237)), (DELSFC, COM(238)),         ABCD0079
4 (CVDWNG, COM(239)), (CVDNOZ, COM(240)), (CVMNOZ, COM(241)),       ABCD0080
5 (VA, COM(242)), (VJD, COM(243)), (VJW, COM(244)),                 ABCD0081
6 (VJM, COM(245)), (WFT, COM(246)), (WGT, COM(247)),                ABCD0082
7 (SFC, COM(248)), (TFAR, COM(249)), (FRD, COM(250)),               ABCD0083
8 (FGMD, COM(251)), (FGMM, COM(252)), (FGPD, COM(253)),             ABCD0084
9 (FGPM, COM(254)), (FGM, COM(255)), (FGP, COM(256)),               ABCD0085
1 (FG, COM(257)), (FN, COM(258)), (FFOVFN, COM(259))                ABCD0086
  EQUIVALENCE  (FCOVFN, COM(260)), (FMNOFN, COM(261)),              ABCD0087
1 (FNOVFD, COM(262)), (T21, COM(263)), (H21, COM(264)),             ABCD0088
2 (S21, COM(265)), (WA21, COM(266)), (T21DS, COM(267)),             ABCD0089
3 (T22, COM(268)), (WA22, COM(269)), (S3, COM(270)),                ABCD0090
4 (WA32, COM(271)), (T55, COM(272)), (H55, COM(273)),               ABCD0091
5 (S55, COM(274)), (TFLPDS, COM(275)), (CNLPDS, COM(276)),          ABCD0092
6 (ETLPDS, COM(277)), (TFIPDS, COM(278)), (CNIPDS, COM(279)),       ABCD0093
7 (ETIPDS, COM(280)), (TFLPCF, COM(281)), (CNLPCF, COM(282)),       ABCD0094
8 (ETLPCF, COM(283)), (DHLPCF, COM(284)), (TFIPCF, COM(285)),       ABCD0095
9 (CNIPCF, COM(286)), (ETIPCF, COM(287)), (DHIPCF, COM(288)),       ABCD0096
1 (TFFLP, COM(289)), (CNLP, COM(290)), (ETATLP, COM(291)),          ABCD0097
2 (DHTCLP, COM(292)), (DHTF, COM(293)), (PRCCF, COM(294)),          ABCD0098
3 (ETACCF, COM(295)), (WACCF, COM(296)), (PRCDS, COM(297)),         ABCD0099
4 (ETACDS, COM(298)), (WACDS, COM(299)), (ZC, COM(300)),            ABCD0100
5 (PCNC, COM(301)), (PCBLC, COM(302)), (PCNCDS, COM(303)),          ABCD0101
6 (PCBLI, COM(304)), (PCBLID, COM(305)), (CNC, COM(306)),           ABCD0102
7 (PRC, COM(307)), (ETAC, COM(308)), (CNI, COM(309)),               ABCD0103
8 (WACI, COM(310)), (WAI, COM(311)), (BLI, COM(312)),               ABCD0104
9 (BLHP, COM(313)), (BLIP, COM(314)), (BLLP, COM(315)),             ABCD0105
1 (BLF, COM(316)), (BLDU, COM(317)), (BLOB, COM(318))               ABCD0106
  EQUIVALENCE  (WAF, COM(319)), (WACC, COM(320)), (WG24, COM(321)), ABCD0107
1 (FAR24, COM(322)), (WG55, COM(323)), (FAR55, COM(324)),           ABCD0108
```

```
     2 (P6DSAV, COM(325)), (AM6DSV, COM(326)), (AM6, COM(327)),                ABCD0109
     3 (A6, COM(328)), (WG6CDS, COM(329)), (WG6, COM(330)),                   ABCD0110
     4 (T6, COM(331)), (P6, COM(332)), (H6, COM(333)), (WG7, COM(334)),       ABCD0111
     5 (FAR7, COM(335)), (FAR7SV, COM(336)), (TS7, COM(337)),                 ABCD0112
     6 (PS7, COM(338)), (V7, COM(339)), (AM7, COM(340)), (A7, COM(341)),      ABCD0113
     7 (T7DS, COM(342)), (T7, COM(343)), (H7, COM(344)), (S7, COM(345)),      ABCD0114
     8 (A8, COM(346)), (A8SAV, COM(347)), (TS8, COM(348)),                    ABCD0115
     9 (PS8, COM(349)), (V8, COM(350)), (AM8, COM(351)), (T8, COM(352)),      ABCD0116
     1 (P8, COM(353)), (H8, COM(354)), (S8, COM(355)), (A9, COM(356)),        ABCD0117
     2 (A9SAV, COM(357)), (TS9, COM(358)), (PS9, COM(359)),                   ABCD0118
     3 (V9, COM(360)), (AM9, COM(361)), (T9, COM(362)), (P9, COM(363)),       ABCD0119
     4 (H9, COM(364)), (S9, COM(365)), (ETAADS, COM(366)),                    ABCD0120
     5 (DPAFDS, COM(367)), (WFA, COM(368)), (ETAA, COM(369)),                 ABCD0121
     6 (ETAASV, COM(370)), (DPAFT, COM(371)), (XNHP, COM(372)),               ABCD0122
     7 (XNIP, COM(373)), (XNLP, COM(374)), (P22, COM(375)),                   ABCD0123
     8 (U22, COM(376)), (P21, COM(377)), (U21, COM(378)),                     ABCD0124
     9 (P3, COM(379)), (U3, COM(380)), (P4, COM(381)), (U4, COM(382)),        ABCD0125
     1 (P50, COM(383)), (U50, COM(384)), (P5, COM(385)), (U5, COM(386))       ABCD0126
      EQUIVALENCE  (P55, COM(387)), (U55, COM(388)), (P7, COM(389)),          ABCD0127
     1 (U7, COM(390)), (P24, COM(391)), (U24, COM(392)),                      ABCD0128
     2 (P37, COM(393)), (U37, COM(394)), (VFAN, COM(395)),                    ABCD0129
     3 (VINTC, COM(396)), (VCOMP, COM(397)), (VCOMB, COM(398)),               ABCD0130
     4 (VHPTRB, COM(399)), (VIPTRB, COM(400)), (VLPTRB, COM(401)),            ABCD0131
     5 (VAFTBN, COM(402)), (VFDUCT, COM(403)), (VWDUCT, COM(404)),            ABCD0132
     6 (DUMD1(1), COM(405)), (WAFP,COM(420)), (WAIP,COM(421)),                ABCD0133
     7 (WACP, COM(422)), (XNHPDS, COM(423)), (XNIPDS, COM(424)),              ABCD0134
     8 (XNLPDS, COM(425)), (PMIHP, COM(426)), (PMIIP, COM(427)),              ABCD0135
     9 (PMILP, COM(428)), (DELT1, COM(429)), (FO(1,1), COM(430)),             ABCD0136
     1 (XNLDEM, COM(630)), (SO(1,1), COM(631)), (PDATA(1,1), COM(691)),       ABCD0137
     2 (TIMEPT(1), COM(941)), (PRFNEW, COM(991)), (PRCNEW, COM(992)),         ABCD0138
     3 (TIME, COM(993)), (DT, COM(994)), (TF, COM(995)),                      ABCD0139
     4 (TPRINT, COM(996)), (DTPRNT, COM(997)), (PVRDOT(1), COM(998)),         ABCD0140
     5 (XS(1), COM(1021)), (ISPOOL, COM(1044)), (ICOAFB, COM(1045)),          ABCD0141
     6 (ICODUC, COM(1046)), (ICOMIX, COM(1047)), (KKGO, COM(1048)),           ABCD0142
     7 (ITRAN, COM(1049)), (JTRAN, COM(1050)), (NSTEP, COM(1051)),            ABCD0143
     8 (IVRDOT, COM(1052)), (IDOT, COM(1053)), (IAMTRX, COM(1054)),           ABCD0144
     9 (SI, COM(1055)), (ERRER, COM(1056)), (DUMSPL, COM(1057)),              ABCD0145
     1 (FXFN2M, COM(1058)), (FXM2CP, COM(1059)), (AFTFAN, COM(1060))          ABCD0146
      EQUIVALENCE  (FAN, COM(1061)), (WAFCDS, COM(1062))                      ABCD0147
      ERRER = .FALSE.                                                         ABCD0148
      ITRAN = 0                                                               ABCD0149
      JTRAN = 0                                                               ABCD0150
      NSTEP = 0                                                               ABCD0151
      TIME = 0.0D0                                                            ABCD0152
      TPRINT = 0.0D0                                                          ABCD0153
      DTPRNT = 0.0D0                                                          ABCD0154
      IDOT = 0                                                                ABCD0155
      IAMTRX = 0                                                              ABCD0156
      DO 1  J = 3,404                                                         ABCD0157
1     WORD(J) = 0.0D0                                                         ABCD0158
      DO 2  J = 1057,1062                                                     ABCD0159
2     WORD(J) = 0.0D0                                                         ABCD0160
      ISPOOL = 0                                                              ABCD0161
C     SET ARBITRARY VALUES FOR INTERMEDIATE SPOOL DESIGN PARAMETERS TO        ABCD0162
```

```
C       AVOID ERROR WHEN RUNNING A DUMMYSPOOL ENGINE                    ABCD0163
        PRIDS = 1.5D0                                                   ABCD0164
        ETAIDS = 1.0D0                                                  ABCD0165
        PCNIDS = 100.0D0                                                ABCD0166
        ZIDS = .75D0                                                    ABCD0167
        PCNCDS = 100.0D0                                                ABCD0168
        KKGO = 0                                                        ABCD0169
        CALL CONOUT(1)                                                  ABCD0170
        P6DSAV = 1.0D0                                                  ABCD0171
        AM6DSV = 1.0D0                                                  ABCD0172
        ETAASV = 1.0D0                                                  ABCD0173
        FAR7SV = 1.0D0                                                  ABCD0174
        CALL ENGBAL                                                     ABCD0175
        STOP                                                            ABCD0176
        END                                                             ABCD0177
```

Subroutine AFQUIR

```
        SUBROUTINE AFQUIR (X,AIND,DEPEND,ANS,AJ,TOL,DIR,ANEW,ICON)      ABCD0178
        IMPLICIT  REAL*8  (A-H,O-Z)                                     ABCD0179
        DIMENSION X(9)                                                  ABCD0180
C X(1)=NAME OF ARRAY TO USE                                             ABCD0181
C AIND=INDEPENDENT VARIABLE                                             ABCD0182
C DEPEND= DEPENDENT VARIABLE                                            ABCD0183
C ANS=ANSWER UPON WHICH TO CONVERGE                                     ABCD0184
C AJ=MAX NUMBER OF TRIES                                                ABCD0185
C TOL=PERCENT TOLERANCE FOR CONVERGENCE                                 ABCD0186
C DIR=DIRECTION AND PERCENTAGE FOR FIRST GUESS                          ABCD0187
C ANEW=CALCULATED VALUE OF NEXT TRY AT INDEPENDENT VARIABLE             ABCD0188
C ICON=CONTROL  =1 GO THRU LOOP AGAIN                                   ABCD0189
C               =2 YOU HAVE REACHED THE ANSWER                          ABCD0190
C               =3 COUNTER HAS HIT LIMITS                               ABCD0191
C X(2)=COUNTER STORAGE                                                  ABCD0192
C X(3)=CHOOSES METHOD OF CONVERGENCE                                    ABCD0193
C X(4)=THIRD DEPEND VAR                                                 ABCD0194
C X(5)=THIRD IND VAR                                                    ABCD0195
C X(6)=SECOND DEPEND VAR                                                ABCD0196
C X(7)=SECOND IND VAR                                                   ABCD0197
C X(8)=FIRST DEPEND VAR                                                 ABCD0198
C X(9)=FIRST IND VAR                                                    ABCD0199
C X(3)  MUST BE ZERO UPON FIRST ENTRY TO ROUTINE                        ABCD0200
        Y = 0.0D0                                                       ABCD0201
        IF (ANS .EQ. 0.0D0)  GO TO 2                                    ABCD0202
        DEP = DEPEND - ANS                                              ABCD0203
        TOLANS = TOL * ANS                                              ABCD0204
        GO TO 3                                                         ABCD0205
2       DEP = DEPEND                                                    ABCD0206
        TOLANS = TOL                                                    ABCD0207
3       IF (DABS(DEP) .LE. TOLANS) GO TO 5                              ABCD0208
        IF (X(2) - AJ)  8,8,7                                           ABCD0209
5       ANEW = AIND                                                     ABCD0210
        X(2) = 0.0D0                                                    ABCD0211
        ICON = 2                                                        ABCD0212
        RETURN                                                          ABCD0213
```

```
6       ANEW = Y                                                        ABCD0214
        X(2) = X(2) + 1.0D0                                             ABCD0215
        ICON = 1                                                        ABCD0216
        RETURN                                                          ABCD0217
7       ANEW = Y                                                        ABCD0218
        X(2) = 0.0D0                                                    ABCD0219
        ICON = 3                                                        ABCD0220
        RETURN                                                          ABCD0221
8       IF (X(3) .GT. 0.0D0)  GO TO 12                                  ABCD0222
C ***   FIRST GUESS USING DIR                                           ABCD0223
9       X(3) = 1.0D0                                                    ABCD0224
        X(8) = DEP                                                      ABCD0225
        X(9) = AIND                                                     ABCD0225
        IF (AIND .EQ. 0.0D0)  GO TO 11                                  ABCD0227
10      Y = DIR * AIND                                                  ABCD0228
        GO TO 6                                                         ABCD0229
11      Y = DIR                                                         ABCD0230
        GO TO 6                                                         ABCD0231
12      IF (X(3) .GT. 1.0D0)  GO TO 16                                  ABCD0232
C ***   LINEAR GUESS                                                    ABCD0233
13      X(3) = 2.0D0                                                    ABCD0234
        X(6) = DEP                                                      ABCD0235
        X(7) = AIND                                                     ABCD0235
        IF (X(8) .EQ. X(6) .OR. X(9) .EQ. X(7))  GO TO 9                ABCD0237
        A = (X(9) - X(7)) / (X(8) - X(6))                               ABCD0238
        Y = X(9) - A * X(8)                                             ABCD0239
        IF (DABS(10.0D0 * X(9)) - DABS(Y))  9,9,6                       ABCD0240
C ***   QUADRATIC GUESS                                                 ABCD0241
15      X(4) = DEP                                                      ABCD0242
        X(5) = AIND                                                     ABCD0243
        IF (X(7) .NE. X(5))  GO TO 18                                   ABCD0244
        IF (X(6) - X(4))  13,9,13                                       ABCD0245
18      IF (X(6) .EQ. X(4))  GO TO 13                                   ABCD0246
        IF (X(9) .NE. X(5))  GO TO 23                                   ABCD0247
        IF (X(8) .EQ. X(4))  GO TO 22                                   ABCD0248
21      X(9) = X(7)                                                     ABCD0249
        X(8) = X(6)                                                     ABCD0250
        GO TO 13                                                        ABCD0251
22      X(9) = X(7)                                                     ABCD0252
        X(8) = X(6)                                                     ABCD0253
        X(3) = 1.0D0                                                    ABCD0254
        IF (X(9))  10,11,10                                             ABCD0255
23      IF (X(8) .EQ. X(4))  GO TO 21                                   ABCD0256
        F = (X(6) - X(4)) / (X(7) - X(5))                               ABCD0257
        A = (X(8) - X(4) - F * (X(9) - X(5))) / ((X(9) - X(7)) *        ABCD0258
      1 (X(9) - X(5)))                                                  ABCD0259
        B = F - A * (X(5) + X(7))                                       ABCD0260
        C = X(4) + X(5) * (A * X(7) - F)                                ABCD0261
        IF (A .NE. 0.0D0)  GO TO 27                                     ABCD0262
        IF (B .EQ. 0.0D0)  GO TO 7                                      ABCD0263
        Y = -C / B                                                      ABCD0264
        GO TO 47                                                        ABCD0265
27      IF (B .NE. 0.0D0)  GO TO 32                                     ABCD0266
        IF (C .NE. 0.0D0)  GO TO 30                                     ABCD0267
```

```
      Y = 0.0D0                                                          ABCD0268
      GO TO 47                                                           ABCD0269
30    G = -C / A                                                         ABCD0270
      IF (G .LE. 0.0D0)  GO TO 7                                         ABCD0271
      Y = DSQRT(G)                                                       ABCD0272
      YY = -DSQRT(G)                                                     ABCD0273
      GO TO 37                                                           ABCD0274
32    IF (C .NE. 0.0D0)  GO TO 34                                        ABCD0275
      Y = -B / A                                                         ABCD0276
      YY = 0.0D0                                                         ABCD0277
      GO TO 37                                                           ABCD0278
34    D = 4.0D0 * A * C / B ** 2                                         ABCD0279
      IF (1.0D0 - D)  13,35,36                                           ABCD0280
35    Y = -B / (2.0D0 * A)                                               ABCD0281
      GO TO 47                                                           ABCD0282
36    E = DSQRT(1.0D0 - D)                                               ABCD0283
      Y = (-B / (2.0D0 * A)) * (1.0D0 + E)                               ABCD0284
      YY = (-B / (2.0D0 * A)) * (1.0D0 - E)                              ABCD0285
37    J = 4                                                              ABCD0286
      DEPMIN = DABS(X(4))                                                ABCD0287
      DO 39  I = 6,8,2                                                   ABCD0288
      IF (DEPMIN .LE. DABS(X(I)))  GO TO 39                              ABCD0289
      J = I                                                              ABCD0290
      DEPMIN = DABS(X(I))                                                ABCD0291
39    CONTINUE                                                           ABCD0292
      K = J + 1                                                          ABCD0293
      IF (((X(K) - Y) * (X(K) - YY)) .LE. 0.0D0)  GO TO 42               ABCD0294
      IF ((DABS(X(K) - Y) - DABS(X(K) - YY)) .LE. 0.0D0)  GO TO 47       ABCD0295
      Y = YY                                                             ABCD0296
      GO TO 47                                                           ABCD0297
42    IF (J .GE. 6)  GO TO 44                                            ABCD0298
      JJ = J + 2                                                         ABCD0299
      KK = K + 2                                                         ABCD0300
      GO TO 45                                                           ABCD0301
44    JJ = J - 2                                                         ABCD0302
      KK = K - 2                                                         ABCD0303
45    SLOPE = (X(KK) - X(K)) / (X(JJ) - X(J))                            ABCD0304
      IF ((SLOPE * X(J) * (X(K) - Y)) .GT. 0.0D0)  GO TO 47              ABCD0305
      Y = YY                                                             ABCD0306
47    X(9) = X(7)                                                        ABCD0307
      X(8) = X(6)                                                        ABCD0308
      X(7) = X(5)                                                        ABCD0309
      X(6) = X(4)                                                        ABCD0310
      GO TO 6                                                            ABCD0311
      END                                                                ABCD0312
```

Subroutine ATMOS

```
      SUBROUTINE ATMOS (ZFT,TM,SIGMA,RHO,THETA,DELTA,CA,AMU,K)           ABCD0313
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD0314
      LOGICAL  SI                                                        ABCD0315
C   THIS IS A SUBROUTINE TO COMPUTE CERTAIN ELEMENTS OF THE 1962         ABCD0316
C   U.S. STANDARD ATMOSPHERE UP TO 90 KILOMETERS.                        ABCD0317
C    CALLING SEQUENCE...                                               ABCD0318
```

```
C                                                                       ABCD0319
C     CALL ATMOS (ZFT, TM, SIGMA, RHO, THETA, DELTA, CA, AMU, K)        ABCD0320
C          ZFT   =   GEOMETRIC ALTITUDE (FEET)                          ABCD0321
C          TM    =   MOLECULAR SCALE TEMPERATURE (DEGREES RANKINE)      ABCD0322
C          SIGMA =   RATIO OF DENSITY TO THAT AT SEA LEVEL             ABCD0323
C          RHO   =   DENSITY(LB-SEC**2-FT**(-4)   OR   SLUG-FT**(-3))  ABCD0324
C          THETA =   RATIO OF TEMPERATURE TO THAT AT SEA LEVEL         ABCD0325
C          DELTA =   RATIO OF PRESSURE TO THAT AT SEA LEVEL            ABCD0326
C          CA    =   SPEED OF SOUND (FT/SEC)                           ABCD0327
C          AMU   =   VISCOSITY COEFFICIENT (LB-SEC/FT**2)              ABCD0328
C                                                                       ABCD0329
C          K = 1 NORMAL                                                 ABCD0330
C            = 2 ALTITUDE LESS THAN -5000 METERS OR GREATER THAN 90 KM  ABCD0331
C            = 3 FLOATING POINT OVERFLOW                                ABCD0332
C                                                                       ABCD0333
C   ALL DATA AND FUNDAMENTAL CONSTANTS ARE IN THE METRIC SYSTEM AS      ABCD0334
C   THESE QUANTITIES ARE DEFINED AS EXACT IN THIS SYSTEM.               ABCD0335
C                                                                       ABCD0336
C   THE RADIUS OF THE EARTH (REFT59) IS THE VALUE ASSOCIATED WITH THE   ABCD0337
C   1959 ARDC ATMOSPHERE SO THAT PROGRAMS CURRENTLY USING THE LIBRARY   ABCD0338
C   ROUTINE WILL NOT REQUIRE ALTERATION TO USE THIS ROUTINE.            ABCD0339
      COMMON /COMALL/  COM(1062)                                        ABCD0340
      DIMENSION  HB(10), TMB(10), DELTAB(10), ALM(10)                   ABCD0341
      EQUIVALENCE  (SI, COM(1055))                                      ABCD0342
      DATA  HB / -5.0D0, 0.0D0, 11.0D0, 20.0D0, 32.0D0, 47.0D0, 52.0D0, ABCD0343
     1 61.0D0, 79.0D0, 88.743D0 /                                       ABCD0344
      DATA  TMB / 320.65D0, 288.15D0, 2 * 216.65D0, 228.65D0,           ABCD0345
     1 2 * 270.65D0, 252.65D0, 2 * 180.65D0 /                           ABCD0346
      DATA  DELTAB / 1.75363D+00, 1.00000D+00, 2.23361D-01, 5.40328D-02,ABCD0347
     1 8.56663D-03, 1.09455D-03, 5.82289D-04, 1.79718D-04, 1.0241D-05,  ABCD0348
     2 1.6223D-06 /                                                     ABCD0349
      DATA  ALM / 2 * -6.5D0, 0.0D0, 1.0D0, 2.8D0, 0.0D0, -2.0D0,       ABCD0350
     1 -4.0D0, 2 * 0.0D0 /                                              ABCD0351
      DATA  REFT59/2.0855531D+07/, GZ    /9.80665D+00/,                 ABCD0352
     1      AMZ   /2.89644D+01  /, RSTAR /8.31432D+00/,                 ABCD0353
     2      FTTOKM/3.048D-04    /, S     /1.104D+02  /,                 ABCD0354
     3      AMUZ  /1.2024D-05   /, CAZ   /1.11645D+03/,                 ABCD0355
     4      RHOZ  /7.6474D-02   /, GZENG /3.21741D+01/                  ABCD0356
C  CONVERT GEOMETRIC ALTITUDE TO GEOPOTENTIAL ALTITUDE                   ABCD0357
C     IF IN SI UNITS, CHANGE ZFT TO FEET                                ABCD0358
      IF (SI)  ZFT = ZFT * 3.280833D0                                   ABCD0359
      HFT = (REFT59 / (REFT59 + ZFT)) * ZFT                             ABCD0360
C  CONVERT HFT AND ZFT TO KILOMETERS                                    ABCD0361
      Z = FTTOKM * ZFT                                                  ABCD0362
      H = FTTOKM * HFT                                                  ABCD0363
      K = 1                                                             ABCD0364
      TMZ = TMB(2)                                                      ABCD0365
      IF (H .LT. -5.0D0 .OR. Z .GT. 90.0D0)  GO TO 7                    ABCD0366
      DO 1  M = 1,10                                                    ABCD0367
      IF (H - HB(M))  2,3,1                                             ABCD0368
1     CONTINUE                                                          ABCD0369
      GO TO 7                                                           ABCD0370
2     M = M - 1                                                         ABCD0371
3     DELH = H - HB(M)                                                  ABCD0372
```

```
      IF (ALM(M) .EQ. 0.0D0)  GO TO 4                                   ABCD0373
      TMK = TMB(M) + ALM(M) * DELH                                      ABCD0374
C  GRADIENT IS NON ZERO, PAGE 10, EQUATION I.2.10-(3)                   ABCD0375
      DELTA = DELTAB(M) * ((TMB(M) / TMK) ** (GZ * AMZ /                ABCD0376
     1 (RSTAR * ALM(M))))                                               ABCD0377
      GO TO 5                                                           ABCD0378
4     TMK = TMB(M)                                                      ABCD0379
C  GRADIENT IS ZERO, PAGE 10, EQUATION I.2.10-(4)                        ABCD0380
      DELTA = DELTAB(M) * DEXP(-GZ * AMZ * DELH / (RSTAR * TMB(M)))     ABCD0381
5     THETA = TMK / TMZ                                                 ABCD0382
      SIGMA = DELTA / THETA                                             ABCD0383
      ALPHA = DSQRT(THETA ** 3) * ((TMZ + S) / (TMK + S))               ABCD0384
C  CONVERSION TO ENGLISH UNITS                                          ABCD0385
      TM = 1.8D0 * TMK                                                  ABCD0386
      RHO = RHOZ * SIGMA / GZENG                                        ABCD0387
      CA = CAZ * DSQRT(THETA)                                           ABCD0388
      AMU = AMUZ * ALPHA / GZENG                                        ABCD0389
      IF (SI)  GO TO 100                                                ABCD0390
      GO TO 101                                                         ABCD0391
100   TM = TM / 1.8D0                                                   ABCD0392
      RHO = RHO * 515.379D0                                             ABCD0393
      CA = CA * .3048D0                                                 ABCD0394
      AMU = AMU * 47.880258D0                                           ABCD0395
      ZFT = ZFT / 3.280833D0                                            ABCD0396
C      IF IN SI UNITS:                                                  ABCD0397
C            TM         DEGREES KELVIN                                 ABCD0398
C            RHO        KG/M**2                                        ABCD0399
C            CA         M/SEC                                          ABCD0400
C            AMU        (N-SEC)/M**2                                   ABCD0401
C            ZFT        M                                              ABCD0402
101   CALL OVERFL (J)                                                   ABCD0403
      IF (J .EQ. 2)  RETURN                                             ABCD0404
      K = K + 2                                                         ABCD0405
      RETURN                                                            ABCD0406
7     K = 2                                                             ABCD0407
      RETURN                                                            ABCD0408
      END                                                               ABCD0409
```

Block Data

```
      BLOCK DATA                                                        ABCD0410
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD0411
C     THESE ARE THE GENERALIZED MAPS FOR AN UNREALISTIC SUPERSONIC      ABCD0412
C     ENGINE.  THEY INCLUDE:  FAN, INTERMEDIATE FAN, COMPRESSOR,        ABCD0413
C     COMBUSTOR, AND HIGH, INTERMEDIATE AND LOW TURBINES.               ABCD0414
      COMMON /COMDAT/  COMD(5423)                                       ABCD0415
      DIMENSION  CNXF(15), PRXF(15,15), WACXF(15,15), ETAXF(15,15),     ABCD0416
     1 NPTF(15)                                                         ABCD0417
      DIMENSION  CNXIN(15), PRXIN(15,15), WACXIN(15,15), ETAXIN(15,15), ABCD0418
     1 NPTI(15)                                                         ABCD0419
      DIMENSION  CNXP(15), PRXP(15,15), WACXP(15,15), ETAXP(15,15),     ABCD0420
     1 NPTP(15)                                                         ABCD0421
      DIMENSION  PSIXB(15), DELXB(15,15), ETAXB(15,15), NPTB(15)        ABCD0422
      DIMENSION  TFFXH(15), CNXH(15,15), DHTCXH(15,15), ETATXH(15,15),  ABCD0423
```

```
1 NPTTFH(15)                                                            ABCD0424
 DIMENSION  TFFXI(15), CNXI(15,15), DHTCXI(15,15), ETATXI(15,15),       ABCD0425
1 NPTTFI(15)                                                            ABCD0426
 DIMENSION CNXI1(15,14), CNXI2(15,1)                                    ABCD0427
 DIMENSION  TFFXL(15), CNXL(15,15), DHTCXL(15,15), ETATXL(15,15),       ABCD0428
1 NPTTFL(15)                                                            ABCD0429
 DIMENSION CNXL1(15,14), CNXL2(15,1)                                    ABCD0430
 EQUIVALENCE  (CNXF(1), COMD(1)), (PRXF(1,1), COMD(16)),                ABCD0431
1 (WACXF(1,1), COMD(241)), (ETAXF(1,1), COMD(466)),                     ABCD0432
2 (NCNF, COMD(5296)), (NPTF(1), COMD(5297))                             ABCD0433
 EQUIVALENCE  (CNXIN(1), COMD(691)), (PRXIN(1,1), COMD(706)),           ABCD0434
1 (WACXIN(1,1), COMD(931)), (ETAXIN(1,1), COMD(1156)),                  ABCD0435
2 (NCNI, COMD(5312)), (NPTI(1), COMD(5313))                             ABCD0436
 EQUIVALENCE  (CNXP(1), COMD(2071)), (PRXP(1,1), COMD(2086)),           ABCD0437
1 (WACXP(1,1), COMD(2311)), (ETAXP(1,1), COMD(2536)),                   ABCD0438
2 (NCNP, COMD(5344)), (NPTP(1), COMD(5345))                             ABCD0439
 EQUIVALENCE  (PSIXB(1), COMD(2761)), (DELXB(1,1), COMD(2776)),         ABCD0440
1 (ETAXB(1,1), COMD(3001)), (NPSB, COMD(5360)),                         ABCD0441
2 (NPTB(1), COMD(5361))                                                 ABCD0442
 EQUIVALENCE  (TFFXH(1), COMD(3226)), (CNXH(1,1), COMD(3241)),          ABCD0443
1 (DHTCXH(1,1), COMD(3466)), (ETATXH(1,1), COMD(3691)),                 ABCD0444
2 (NTFFSH, COMD(5376)), (NPITFH(1), COMD(5377))                         ABCD0445
 EQUIVALENCE  (TFFXI(1), COMD(3916)), (CNXI(1,1), COMD(3931)),          ABCD0446
1 (DHTCXI(1,1), COMD(4156)), (ETATXI(1,1), COMD(4381)),                 ABCD0447
2 (NTFFSI, COMD(5392)), (NPITFI(1), COMD(5393))                         ABCD0448
 EQUIVALENCE (CNXI(1,1), CNXI1(1,1)), (CNXI(1,15), CNXI2(1,1))          ABCD0449
 EQUIVALENCE  (TFFXL(1), COMD(4606)), (CNXL(1,1), COMD(4621)),          ABCD0450
1 (DHTCXL(1,1), COMD(4846)), (ETATXL(1,1), COMD(5071)),                 ABCD0451
2 (NTFFSL, COMD(5408)), (NPITFL(1), COMD(5409))                         ABCD0452
 EQUIVALENCE (CNXL(1,1), CNXL1(1,1)), (CNXL(1,15), CNXL2(1,1))          ABCD0453
 DATA NCNF, NPTF / 10, 6, 3 * 7, 5 * 10, 8, 5 * 0 /                     ABCD0454
 DATA CNXF / .3, .4, .5, .6, .7, .8, .9, 1.0, 1.1, 1.2, 5 * 0.0 /       ABCD0455
 DATA PRXF /                                                            ABCD0456
1 10 * 1.0, 5 * 0.0, 1.012, 1.02, 1.0256, 1.0368, 1.064, 1.1,          ABCD0457
2 1.076, 1.044, 1.104, 1.1632, 5 * 0.0, 1.028, 1.04, 1.0512, 1.088,    ABCD0458
3 1.1184, 1.16, 1.152, 1.1352, 1.22, 1.312, 5 * 0.0, 1.0384,           ABCD0459
4 1.0584, 1.08, 1.124, 1.148, 1.2, 1.2192, 1.2208, 1.324, 1.4,         ABCD0460
5 5 * 0.0, 1.0048, 1.0752, 1.116, 1.16, 1.184, 1.228, 1.26, 1.2944,    ABCD0461
6 1.4, 1.48, 5 * 0.0, 1.048, 1.092, 1.132, 1.1896, 1.2096, 1.2552,     ABCD0462
7 1.2896, 1.34, 1.448, 1.5, 6 * 0.0, 1.1, 1.148, 1.1952, 1.2176,       ABCD0463
8 1.272, 1.3312, 1.4, 1.5, 1.58, 9 * 0.0, 1.224, 1.2864, 1.3616,       ABCD0464
9 1.428, 1.5336, 1.66, 9 * 0.0, 1.244, 1.3024, 1.3912, 1.448,          ABCD0465
1 1.568, 10 * 0.0, 1.2672, 1.332, 1.4, 1.48, 1.584, 81 * 0.0 /         ABCD0466
 DATA WACXF /                                                           ABCD0467
1 243.6, 286.8, 333.6, 383.4, 439.8, 499.8, 566.4, 633.6, 700.2,       ABCD0468
2 750.0, 5 * 0.0, 229.8, 270.0, 322.8, 376.2, 436.8, 499.8, 566.4,     ABCD0469
3 633.6, 700.2, 750.0, 5 * 0.0, 199.8, 253.2, 310.2, 358.2, 428.4,     ABCD0470
4 493.2, 566.4, 633.6, 700.2, 750.0, 5 * 0.0, 166.8, 233.4, 291.6,     ABCD0471
5 340.2, 420.6, 485.4, 559.8, 633.0, 700.2, 750.0, 5 * 0.0, 133.2,     ABCD0472
6 209.4, 259.8, 313.2, 406.8, 476.4, 553.2, 625.8, 700.2, 750.0,       ABCD0473
7 5 * 0.0, 86.4, 183.6, 240.0, 276.6, 393.6, 466.8, 544.8, 616.8,      ABCD0474
8 693.4, 750.0, 6 * 0.0, 156.6, 213.6, 266.4, 388.2, 456.6, 528.6,     ABCD0475
9 600.0, 693.6, 749.4, 9 * 0.0, 383.4, 448.2, 509.4, 586.8, 683.4,     ABCD0476
1 736.8, 9 * 0.0, 368.4, 433.2, 483.6, 576.6, 666.6, 10 * 0.0,         ABCD0477
```

```
2 342.6, 406.8, 474.0, 553.2, 656.4, 81 * 0.0 /                          ABCD0478
 DATA ETAXF /                                                            ABCD0479
1 2 * .75592, .75064, .74536, .72512, .68816, .64152, .60016,            ABCD0480
2 .56936, .51744, 5 * 0.0, .7612, 5 * .77616, .72512, 3 * .64152,        ABCD0481
3 5 * 0.0, .76648, .792, .80256, 3 * .82808, .77616, 3 * .72512,         ABCD0482
4 5 * 0.0, .75592, .79728, .82808, 3 * .85448, .82808, 2 * .77616,       ABCD0483
5 .75592, 5 * 0.0, .72512, .80256, .84392, 3 * .88, .85888, .82808,      ABCD0484
6 .80256, .7612, 5 * 0.0, .64152, .77616, 2 * .82808, 2 * .90112,        ABCD0485
7 .88, .85888, .80784, .75064, 5 * 0.0, .74008, .77616, .80784,          ABCD0486
8 .90376, .9108, .90112, .88, .80256, .72512, 9 * 0.0, 2 * .90112,       ABCD0487
9 .88, .85888, .77616, .64152, 9 * 0.0, 2 * .88, 2 * .82808,             ABCD0488
1 .74536, 10 * 0.0, .82808, .8272, .81752, .78672, .72512,               ABCD0489
2 81 * 0.0 /                                                             ABCD0490
 DATA NCNI, NPTI / 10, 6, 3 * 7, 5 * 10, 8, 5 * 0 /                      ABCD0491
 DATA CNXIN / .3, .4, .5, .6, .7, .8, .9, 1.0, 1.1, 1.2, 5 * 0.0 /       ABCD0492
 DATA PRXIN /                                                            ABCD0493
1 10 * 1.0, 5 * 0.0, 1.018, 1.03, 1.0384, 1.0552, 1.096, 1.15,           ABCD0494
2 1.114, 1.066, 1.156, 1.2448, 5 * 0.0, 1.042, 1.06, 1.0768,             ABCD0495
3 1.132, 1.1776, 1.24, 1.228, 1.2028, 1.33, 1.468, 5 * 0.0, 1.0576,      ABCD0496
4 1.0876, 1.12, 1.186, 1.222, 1.3, 1.3288, 1.3312, 1.486, 1.6,           ABCD0497
5 5 * 0.0, 1.0672, 1.1128, 1.174, 1.24, 1.276, 1.342, 1.39, 1.4416,      ABCD0498
6 1.6, 1.72, 5 * 0.0, 1.072, 1.138, 1.198, 1.2844, 1.3144, 1.3828,       ABCD0499
7 1.4344, 1.51, 1.672, 1.81, 6 * 0.0, 1.15, 1.222, 1.2928, 1.3264,       ABCD0500
8 1.408, 1.4968, 1.6, 1.75, 1.87, 9 * 0.0, 1.336, 1.4296, 1.542,         ABCD0501
9 1.642, 1.8004, 1.99, 9 * 0.0, 1.366, 1.4536, 1.5868, 1.672,            ABCD0502
1 1.852, 10 * 0.0, 1.4008, 1.493, 1.6, 1.72, 1.876, 81 * 0.0 /           ABCD0503
 DATA WACXIN /                                                           ABCD0504
1 121.8, 143.4, 166.8, 191.7, 219.9, 249.9, 283.2, 316.8, 350.1,         ABCD0505
2 375.0, 5 * 0.0, 114.9, 135.0, 161.4, 188.1, 218.4, 249.9, 283.2,       ABCD0506
3 316.8, 350.1, 375.0, 5 * 0.0, 99.9, 126.6, 155.1, 179.1, 214.2,        ABCD0507
4 246.6, 283.2, 316.8, 350.1, 375.0, 5 * 0.0, 83.4, 116.7, 145.8,        ABCD0508
5 170.1, 210.3, 242.7, 279.9, 316.5, 350.1, 375.0, 5 * 0.0, 66.6,        ABCD0509
6 104.7, 129.9, 156.6, 203.4, 238.2, 276.6, 312.9, 350.1, 375.0,         ABCD0510
7 5 * 0.0, 43.2, 91.8, 120.0, 133.3, 196.8, 233.4, 272.4, 308.4,         ABCD0511
8 349.2, 375.0, 6 * 0.0, 78.3, 106.8, 133.2, 194.1, 228.3, 264.3,        ABCD0512
9 300.0, 346.8, 374.7, 9 * 0.0, 191.7, 224.1, 254.7, 293.4, 341.7,       ABCD0513
1 368.4, 9 * 0.0, 184.2, 216.6, 241.8, 288.3, 333.3, 10 * 0.0,           ABCD0514
2 171.3, 203.4, 237.0, 276.6, 328.2, 81 * 0.0 /                          ABCD0515
 DATA ETAXIN /                                                           ABCD0516
1 2 * .75592, .75064, .74536, .72512, .68816, .64152, .60016,            ABCD0517
2 .56936, .51744, 5 * 0.0, .7612, 5 * .77616, .72512, 3 * .64152,        ABCD0518
3 5 * 0.0, .76648, .792, .80256, 3 * .82808, .76616, 3 * .72512,         ABCD0519
4 5 * 0.0, .75592, .79728, .82808, 3 * .85448, .82808, 2 * .77616,       ABCD0520
5 .75592, 5 * 0.0, .72512, .80256, .84392, 3 * .88, .85888, .82808,      ABCD0521
6 .80256, .7612, 5 * 0.0, .64152, .77616, 2 * .82808, 2 * .90112,        ABCD0522
7 .88, .85888, .80784, .75064, 5 * 0.0, .74008, .77616, .80784,          ABCD0523
8 .90376, .9108, .90112, .88, .80256, .72512, 9 * 0.0, 2 * .90112,       ABCD0524
9 .88, .85888, .77616, .64152, 9 * 0.0, 2 * .88, 2 * .82808,             ABCD0525
1 .74536, 10 * 0.0, .82808, .8272, .81752, .78672, .72512,               ABCD0526
2 81 * 0.0 /                                                             ABCD0527
 DATA NCNP, NPTP / 10, 2 * 6, 2 * 8, 4 * 10, 2 * 8, 5 * 0 /              ABCD0528
 DATA CNXP / .562, .674, .787, .899, 1.0, 1.034, 1.067, 1.124,           ABCD0529
1 1.236, 1.292, 5 * 0.0 /                                                ABCD0530
 DATA PRXP /                                                             ABCD0531
```

```
     1  10 * 1.0, 5 * 0.0, 1.84, 1.966, 1.84, 2.008, 2.519, 2.855, 3.261,ABCD0532
     2  1.686, 4.353, 3.765, 5 * 0.0, 2.428, 3.093, 2.68, 3.429, 3.982,  ABCD0533
     3  4.297, 4.759, 3.849, 7.622, 6.481, 5 * 0.0, 2.869, 3.933, 3.408, ABCD0534
     4  4.605, 5.277, 5.613, 6.117, 5.466, 10.219, 9.176, 5 * 0.0, 3.835,ABCD0535
     5  4.689, 4.521, 5.697, 6.488, 6.936, 7.454, 6.866, 11.059, 10.219, ABCD0536
     6  5 * 0.0, 4.549, 5.529, 5.445, 6.614, 7.202, 7.622, 8.308, 8.371, ABCD0537
     7  11.899, 11.479, 7 * 0.0, 6.313, 7.538, 8.0, 8.546, 9.218, 8.966, ABCD0538
     8  13.159, 12.711, 7 * 0.0, 6.523, 7.958, 8.567, 9.134, 9.638,      ABCD0539
     9  9.883, 13.656, 14.412, 9 * 0.0, 9.386, 9.925, 10.513, 10.912,    ABCD0540
     1  11 * 0.0, 9.596, 10.219, 10.936, 11.815, 82 * 0.0 /              ABCD0541
      DATA WACXP /                                                       ABCD0542
     1  51.0, 59.3, 70.0, 84.8, 101.7, 108.1, 114.5, 122.9, 139.8, 146.2,ABCD0543
     2  5 * 0.0, 50.2, 59.3, 70.0, 84.8, 101.7, 108.1, 114.5, 122.9,     ABCD0544
     3  139.8, 146.2, 5 * 0.0, 49.5, 58.8, 70.0, 84.8 ,101.7, 108.1,     ABCD0545
     4  114.5, 122.9, 139.8, 146.2, 5 * 0.0, 48.8, 57.9, 69.5, 84.8,     ABCD0546
     5  101.7, 108.1, 114.5, 122.9, 139.8, 146.2, 5 * 0.0, 46.7, 56.7,   ABCD0547
     6  68.8, 84.0, 101.2, 107.6, 114.5, 122.9, 139.8, 146.2, 5 * 0.0,   ABCD0548
     7  44.5, 55.0, 67.9, 83.3, 101.0, 107.1, 114.3, 122.9, 139.5, 146.2,ABCD0549
     8  7 * 0.0, 66.4, 81.7, 100.0, 106.7, 113.6, 122.6, 139.3, 146.2,   ABCD0550
     9  7 * 0.0, 65.7, 80.5, 99.5, 106.0, 113.3, 122.1, 139.0, 146.2,    ABCD0551
     1  9 * 0.0, 98.1, 104.5, 112.6, 121.7, 11 * 0.0, 97.4, 104.0, 112.4,ABCD0552
     2  120.7, 82 * 0.0 /                                                 ABCD0553
      DATA ETAXP /                                                       ABCD0554
     1  2 * .59082, .58566, .5805, .5719, .57018, .55986, .53922, .47644,ABCD0555
     2  .46612, 5 * 0.0, .62178, 6 * .64242, .57018, .60114, .57018,      ABCD0556
     3  5 * 0.0, .64242, .69402, .6837, 4 * .72498, .64242, .72498,       ABCD0557
     4  .64242, 5 * 0.0, .65274, 2 * .72498, 4 * .77748, .72498, .77744,  ABCD0558
     5  .72498, 5 * 0.0, .67338, .74552, .77744, 4 * .8084, .77744,       ABCD0559
     6  .7826, .75078, 5 * 0.0, .64242, .72498, .79292, .82904,           ABCD0560
     7  3 * .83936, .8084, .77744, .75078, 7 * 0.0, .77744, .8084,        ABCD0561
     8  2 * .86, .84968, .82388, 2 * .72498, 7 * 0.0, .7697, .79292,      ABCD0562
     9  4 * .83936, .69918, .64242, 9 * 0.0, 4 * .8084, 11 * 0.0, .80582,ABCD0563
     1  .8041, .79808, .77744, 82 * 0.0 /                                 ABCD0564
      DATA NPSB, NPTB / 15, 15 * 15 /                                    ABCD0565
      DATA PSIXB / 4.9116, 9.8232, 14.735, 19.646, 24.558, 29.47,        ABCD0566
     1  34.381, 39.293, 44.207, 73.674, 100.0, 200.0, 300.0, 400.0,      ABCD0567
     2  500.0 /                                                           ABCD0568
      DATA DELXB / 15 * 200.0, 15 * 300.0, 15 * 400.0, 15 * 500.0,       ABCD0569
     1  15 * 600.0, 15 * 700.0, 15 * 800.0, 15 * 900.0, 15 * 1000.0,     ABCD0570
     2  15 * 1100.0, 15 * 1200.0, 15 * 1300.0, 15 * 1400.0, 15 * 1500.0, ABCD0571
     3  15 * 1600.0 /                                                     ABCD0572
      DATA ETAXB/                                                        ABCD0573
     1  .6, .758, .868, .925, .96, .983, 9 * 1.0, .726, .825, .893, .936,ABCD0574
     2  .966, .991, 9 * 1.0, .777, .858, .911, .946, .972, .992, 9 * 1.0,ABCD0575
     3  .806, .875, .925, .955, .977, .994, 9 * 1.0, .826, .888, .935,    ABCD0576
     4  .933, .982, .995, 9 * 1.0, .843, .898, .942, .969, .985, .997,    ABCD0577
     5  9 * 1.0, .855, .906, .947, .974, .99, .998, 9 * 1.0, .865, .912,  ABCD0578
     6  .951, .977, .992, .999, 9 * 1.0, .87, .914, .953, .978, .993,     ABCD0579
     7  2 * .999, 8 * 1.0, .87, .915, .953, .979, .995, 2 * .999,         ABCD0580
     8  8 * 1.0, .87, .915, .953, .979, .995, 2 * .999, 8 * 1.0, .87,     ABCD0581
     9  .915, .953, .979, .995, 2 * .999, 8 * 1.0, .87, .915, .953, .979,ABCD0582
     1  .995, 2 * .999, 8 * 1.0, .87, .915, .953, .979, .995, 2 * .999,   ABCD0583
     2  8 * 1.0, .87, .915, .953, .979, .995, 2 * .999, 8 * 1.0 /         ABCD0584
      DATA NTFFSH, NPTTFH / 10, 9 * 15, 12, 5 * 0 /                      ABCD0585
```

```
 DATA TFFXH / 39.67, 42.99, 47.46, 48.61, 49.175, 49.6, 50.0,       ABCD0586
1 50.425, 50.92, 51.575, 5 * 0.0 /                                  ABCD0587
 DATA CNXH /                                                        ABCD0588
1 10 * .1872, 5 * 0.0, .3372, .3942, .4362, .255, .3, .3568, .4314,ABCD0589
2 .4834, .3372, .2814, 5 * 0.0, .5156, .5814, .6568, .4784, .5254, ABCD0590
3 .6196, .6844, .7314, .5344, .3304, 5* 0.0, 2 * .7128, .8726,     ABCD0591
4 .6942, .75, .8628, .9568, .8814, .6754, .4686, 5 * 0.0, .9382,   ABCD0592
5 .8442, 1.0696, .9148, .9754, 1.0932, 1.2010, 1.0226, .8068,      ABCD0593
6 .5628, 5 * 0.0, 1.1442, .9804, 1.2382, 1.1442, 1.2754, 1.2852,   ABCD0594
7 1.3834, 1.1442, .9196, .6382, 5 * 0.0, 1.3138, 1.1068, 1.4638,   ABCD0595
8 1.3882, 1.4824, 1.501, 1.5108, 1.2804, 1.0128, .6892, 5 * 0.0,   ABCD0596
9 1.5382, 1.2754, 1.6882, 1.5618, 1.7638, 1.6882, 1.6186, 1.3696,  ABCD0597
1 1.1254, .7362, 5 * 0.0, 1.7264, 1.445, 1.9696, 1.801, 2.045,     ABCD0598
2 1.9138, 1.745, 1.4638, 1.2196, .7696, 5 * 0.0, 1.9324, 1.7068,   ABCD0599
3 2.2138, 1.9794, 2.3362, 2.1246, 1.8618, 1.595, 1.3138, .8068,    ABCD0600
4 5 * 0.0, 2.15, 1.9696, 2.552, 2.2794, 2.645, 2.2706, 1.9558,     ABCD0601
5 1.6746, 1.3696, .8254, 5 * 0.0, 2.4058, 2.2706, 2.805, 2.5138,   ABCD0602
6 2.8706, 2.4226, 2.0, 1.745, 1.4068, .8304, 5 * 0.0, 2.5892,      ABCD0603
7 2.697, 3.0392, 2.8334, 3.0764, 2.495, 2.045, 1.801, 1.445,       ABCD0604
8 6 * 0.0, 2.7862, 3.096, 3.2648, 3.1422, 3.152, 2.5372, 2.0824,   ABCD0605
9 1.8156, 1.4638, 6 * 0.0, 2.945, 3 * 3.3774, 3.1618, 2.5558,      ABCD0606
1 2.101, 1.8196, 1.4676, 6 * 0.0 /                                  ABCD0607
 DATA DHTCXH /                                                      ABCD0608
1 .0032, .0038, .0046, .0052, .0056, .0068, .008, .0088, .0093,    ABCD0609
2 .0132, 5 * 0.0, .0057, .008, .01, .0068, .0088, .012, .0164,     ABCD0610
3 .0196, .0159, .018, 5 * 0.0, .0084, .0113, .0144, .012, .0140,   ABCD0611
4 .0192, .0236, .0272, .0232, .0228, 5 * 0.0, .0108, .0136, .0184, ABCD0612
5 .0164, .0192, .0252, .0308, .0316, .0284, .0268, 5 * 0.0, .0133, ABCD0613
6 .0156, .0216, .0204, .0236, .03, .0372, .0356, .033, .0314,      ABCD0614
7 5 * 0.0, .0152, .0176, .024, .0244, .0288, .034, .0416, .0392,   ABCD0615
8 .0368, .0352, 5 * 0.0, .0164, .0192, .0268, .028, .0321, .0380,  ABCD0616
9 .0448, .0432, .04, .038, 5 * 0.0, .0174, .0212, .0292, .0304,    ABCD0617
1 .036, .0421, .0476, .046, .0442, .0412, 5 * 0.0, .0179, .0228,   ABCD0618
2 .0316, .0336, .04, .0472, .051, .0488, .048, .044, 5 * 0.0,      ABCD0619
3 .0176, .0248, .0331, .0356, .0444, .0524, .0544, .0528, .0524,   ABCD0620
4 .0476, 5 * 0.0, .0167, .026, .0344, .0388, .0496, .0564, .0576,  ABCD0621
5 .056, .0556, .0504, 5 * 0.0, .0144, .0261, .0346, .0412, .054,   ABCD0622
6 .0612, .06, .0596, .058, .053, 5 * 0.0, .012, .0241, .034, .0401,ABCD0623
7 .0596, .064, .0624, .064, .0612, 6 * 0.0, .0082, .0188, .0324,   ABCD0624
8 .0472, .064, .0668, .066, .0654, .064, 6 * 0.0, .0034, .0128,    ABCD0625
9 .0312, .0494, .0661, .0698, .07, .0693, .0668, 6 * 0.0 /         ABCD0626
 DATA ETATXH /                                                      ABCD0627
1 .6219, .6068, .5764, .5643, .5562, .5309, .5062, .5051, .4909,   ABCD0628
2 .4257, 5 * 0.0, 3 * .7078, 5 * .6068, .538, .4747, 5 * 0.0,      ABCD0629
3 .7868, 2 * .809, 5 * .7078, .5068, .5056, 5 * 0.0, .809, .8292,  ABCD0630
4 .8494, 4 * .809, .7665, .6573, .5359, 5 * 0.0, .809, .8363,      ABCD0631
5 .8543, 4 * .8494, .809, .7078, .5683, 5 * 0.0, .7963, .8393,     ABCD0632
6 .8515, .8596, 3 * .8697, .8292, .7463, .5941, 5 * 0.0, .7779,    ABCD0633
7 .8368, .8494, .8596, .8696, .8319, .8797, .8494, .7776, .6068,   ABCD0634
8 5 * 0.0, .7422, .8302, .8409, .8575, .8662, 2 * .8899, .8596,    ABCD0635
9 .809, .6178, 5 * 0.0, .7078, .8254, .8262, .8535, .8615, .894,   ABCD0636
1 .8954, .8697, .8191, .624, 5 * 0.0, .7635, 2 * .809, .8494,      ABCD0637
2 .8555, .8969, .9, .8808, .8302, .631, 5 * 0.0, .6068, .7696,     ABCD0638
3 .7579, .8363, .852, .8975, .901, .8848, .8347, .6265, 5 * 0.0,   ABCD0639
```

```
4 .5309, 2 * .7078, .8262, .8494, .8976, .9, .8848, .8363, .6118,  ABCD0640
5 5 * 0.0, .4773, .6068, .5652, .309, .8494, .8968, .898, .8788,   ABCD0641
6 .8322, 6 * 0.0, .4045, .5056, .6068, .7797, .8532, .8937, .8925, ABCD0642
7 .8697, .8241, 6 * 0.0, .3034, .4197, .5865, .7584, .857, .8896,  ABCD0643
8 .8793, .859, .809, 6 * 0.0 /                                     ABCD0644
 DATA NTFFSI, NPTTFI / 11, 9 * 15, 12, 9, 4 * 0 /                  ABCD0645
 DATA TFFXI / 70.776, 82.236, 93.468, 103.464, 112.836, 116.58,    ABCD0646
1 120.0, 122.676, 125.124, 127.324, 130.536, 4 * 0.0 /             ABCD0647
 DATA CNXI1 /                                                      ABCD0648
1 11 * .3522, 4 * 0.0, .5104, .5278, .5654, .4052, .4844, .5896,   ABCD0649
2 .7392, .5808, .5278, .4574, .4226, 4 * 0.0, .7044, .7575, .8279, ABCD0650
3 .6514, .7044, .8008, .9689, .7575, .634, .6167, .5278, 4 * 0.0,  ABCD0651
4 .933, 1.0208, 1.0296, .8452, .933, 1.0567, 1.2109, .933, .7922,  ABCD0652
5 .7218, .6167, 4 * 0.0, 1.1618, 1.2322, 1.1975, 1.0567,2 * 1.2322,ABCD0653
6 1.4089, 1.1801, .9689, .8279, .7044, 4 * 0.0, 1.3556, 1.3818,    ABCD0654
7 1.373, 1.2322, 1.4967, 1.4619, 1.6056, 1.3915, 1.1183, .933,     ABCD0655
8 .7922, 4 * 0.0, 1.5497, 1.6201, 1.5497, 1.4434, 1.6548, 1.6722,  ABCD0656
9 1.7609, 1.5671, 1.1801, 1.0567, .8452, 4 * 0.0, 1.6905, 1.813,   ABCD0657
1 1.7609, 1.6722, 1.8834, 1.866, 1.9367, 1.7609, 1.3209, 1.1493,   ABCD0658
2 .8983, 4 * 0.0, 1.9367, 1.9723, 1.9367, 1.954, 2.0071, 2.1171,   ABCD0659
3 2.0948, 1.866, 1.4619, 1.2148, .9293, 4 * 0.0, 2.1835, 2.1305,   ABCD0660
4 2.1479, 2.1131, 2.1652, 2.3245, 2.2, 1.9897, 1.5497, 1.2505,     ABCD0661
5 5 * 0.0, 2.3593, 2.2715, 2.3245, 2.2715, 2.3274, 2.5357, 2.2889, ABCD0662
6 2.0601, 1.6722, 1.2784, 5 * 0.0, 2.5001, 2.5089, 2.4827, 2.4915, ABCD0663
7 2.5531, 2.7375, 2.3949, 2.1131, 1.7609, 1.2824, 5 * 0.0, 2.6941, ABCD0664
8 2.7471, 2.6583, 2.7471, 2.8175, 3.0019, 2.4471, 2.1652, 1.813,   ABCD0665
9 6 * 0.0, 2.8175, 2 * 2.9227, 2.9931, 3.0461, 3.1167, 2.5001,     ABCD0666
1 2.2009, 1.8315, 6 * 0.0 /                                        ABCD0667
 DATA CNXI2 /                                                      ABCD0668
1 6 * 3.1698, 2.5175, 2.2048, 1.8401, 6 * 0.0/                     ABCD0669
 DATA DHTCXI /                                                     ABCD0670
1 .0016, .0023, .0027, .0029, .0031, .0034, .0038, .0042, .0047,   ABCD0671
2 .0054, .0061, 4 * 0.0, .0023, .0035, .0045, .0034, .0043, .0057, ABCD0672
3 .0078, .0069, .007, .0069, .0075, 4 * 0.0, .0031, .0047, .0063,  ABCD0673
4 .0054, .0062, .0076, .0101, .009, .0084, .0092, .0093, 4 * 0.0,  ABCD0674
5 .0038, .0061, .0076, .0069, .0081, .01, .0124, .0109, .0104,     ABCD0675
6 .0107, .0108, 4 * 0.0, .0045, .007, .0087, .0084, .0105, .0114,  ABCD0676
7 .0142, .0135, .0124, .0123, .0124, 4 * 0.0, .0049, .0076, .0098, ABCD0677
8 .0097, .0124, .0134, .0159, .0156, .0141, .0138, .014, 4 * 0.0,  ABCD0678
9 .0052, .0084, .0107, .0111, .0136, .015, .0173, .0177, .0148,    ABCD0679
1 .0159, .0151, 4 * 0.0, .0054, .0089, .0118, .0124, .0152, .0165, ABCD0680
2 .019, .0199, .0166, .0177, .0164, 4 * 0.0, .0055, .0092, .0126,  ABCD0681
3 .014, .0159, .0184, .0207, .0213, .0184, .0191, .0177, 4 * 0.0,  ABCD0682
4 .0054, .0094, .0134, .0146, .0169, .0199, .022, .023, .0196,     ABCD0683
5 .0202, 5 * 0.0, .0051, .0095, .0139, .0153, .0178, .0214, .0233, ABCD0684
6 .0241, 2 * .0214, 5 * 0.0, .0047, .0093, .0142, .0161, .0189,    ABCD0685
7 .0228, .025, .0251, .0232, .0221, 5 * 0.0, .0038, .0089, .0146,  ABCD0686
8 .0168, .0199, .0251, .0261, .0263, .0245, 6 * 0.0, .0031, .0083, ABCD0687
9 .0147, .0172, .0207, .0267, 2 * .0276, .0255, 6 * 0.0, .0001,    ABCD0688
1 .0068, .0145, .0173, .021, .028, .029, .0283, .0267, 6 * 0.0 /   ABCD0689
 DATA ETATXI /                                                     ABCD0690
1 .712, 2 * .8, .7995, .775, .76, .731, .71, .678, .638, .6,       ABCD0691
2 4 * 0.0, .73, .81, .83, 4 * .8, .745, .7, .655, .6, 4 * 0.0,     ABCD0692
3 .7472, .82, .86, .84, .848, .345, .83, .768, .7125, .67, .612,   ABCD0693
```

```
4 4 * 0.0, .73, .83, .863, 4 * .86, .8, .735, .685, .617, 4 * 0.0, ABCD0694
5 .714, .83, .867, .868, .875, .873, .875, .838, .769, .7, .621,   ABCD0695
6 4 * 0.0, .7, .829, .87, .873, 3 * .89, .86, .8, .711, .6258,     ABCD0696
7 4 * 0.0, .685, .81, .872, .88, .8912, .895, .893, .8712, .806,   ABCD0697
8 .718, .625, 4 * 0.0, .673, .8, .872, .883, .894, .9, .8975, .878,ABCD0698
9 .8225, .718, .623, 4 * 0.0, .6452, .785, .87, .8835, .8955,      ABCD0699
1 .9005, .8999, .88, .8395, .717, .6009, 4 * 0.0, .62, .76, .867,  ABCD0700
2 .883, .897, .901, .9, .8775, .845, .714, 5 * 0.0, .6, .745, .86, ABCD0701
3 .88, .8961, .9004, .898, .876, .847, .7, 5 * 0.0, .575, .7, .85, ABCD0702
4 .874, .89, .9, .8937, .8722, .8445, .689, 5 * 0.0, .531, .68,    ABCD0703
5 .83, .86, .879, 2 * .89, .866, .833, 6 * 0.0, .5, .645, .8, .835,ABCD0704
6 .8671, .88, .8799, .86, .8235, 6 * 0.0, .385, .59, .76, .82, .86,ABCD0705
7 .8735, .871, .848, .808, 6 * 0.0 /                               ABCD0706
 DATA NTFFSL, NPTTFL / 11, 9 * 15, 12, 9, 4 * 0 /                  ABCD0707
 DATA TFFXL / 88.47, 102.795, 116.835, 129.33, 141.045, 145.725,   ABCD0708
1 150.0, 153.345, 156.405, 159.78, 163.17, 4 * 0.0 /               ABCD0709
 DATA CNXL1 /                                                      ABCD0710
1 11 * .3682, 4 * 0.0, .5336, .5518, .5911, .4237, .5065, .6164,   ABCD0711
2 .7728, .6072, .5518, .4782, .4418, 4 * 0.0, .7365, .7919, .8655, ABCD0712
3 .681, .7365, .8372, 1.0129, .7919, .6629, .6447, .5518, 4 * 0.0, ABCD0713
4 .9754, 1.0672, 1.0764, .3837, .9754, 1.1047, 1.2659, .9754,      ABCD0714
5 .8282, .7546, .6447, 4 * 0.0, 1.2146, 1.2882, 1.2519, 1.1047,    ABCD0715
6 2 * 1.2882, 1.4729, 1.2337, 1.0129, .8655, .7365, 4 * 0.0,       ABCD0716
7 1.4173, 1.4446, 1.4354, 1.2882, 1.5647, 1.5283, 1.6785, 1.4548,  ABCD0717
8 1.1691, .9754, .8282, 4 * 0.0, 1.6201, 1.6937, 1.6201, 1.509,    ABCD0718
9 1.7301, 1.7482, 1.8409, 1.6383, 1.2337, 1.1047, .8837, 4 * 0.0,  ABCD0719
1 1.7673, 1.8954, 1.8409, 1.7482, 1.969, 1.9509, 2.0247, 1.8409,   ABCD0720
2 1.3809, 1.2015, .9391, 4 * 0.0, 2.0247, 2.0619, 2.0247, 2.0429,  ABCD0721
3 2.0983, 2.2133, 2.1901, 1.9509, 1.5283, 1.2701, .9715, 4 * 0.0,  ABCD0722
4 2.2827, 2.2273, 2.2455, 2.2091, 2.2637, 2.4302, 2.3, 2.0801,     ABCD0723
5 1.6201, 1.3073, 5 * 0.0, 2.4655, 2.3747, 2.4302, 2.3747, 2.4332, ABCD0724
6 2.651, 2.3929, 2.1537, 1.7482, 1.3365, 5 * 0.0, 2.6137, 2.6229,  ABCD0725
7 2.5956, 2.6047, 2.6691, 2.8619, 2.5038, 2.2091, 1.8409, 1.3407,  ABCD0726
8 5 * 0.0, 2.8166, 2.872, 2.7791, 2.872, 2.9456, 3.1384, 2.5583,   ABCD0727
9 2.2637, 1.8954, 6 * 0.0, 2.9456, 2 * 3.0555, 3.1291, 3.1846,     ABCD0728
1 3.2584, 2.6137, 2.3009, 1.9147, 6 * 0.0 /                        ABCD0729
 DATA CNXL2 /                                                      ABCD0730
1 6 * 3.3138, 2.6319, 2.3051, 1.9237, 6 * 0.0 /                    ABCD0731
 DATA DHTCXL /                                                     ABCD0732
1 .0018, .0026, .0031, .0033, .0036, .0038, .0044, .0048, .0054,   ABCD0733
2 .0061, .0069, 4 * 0.0, .0026, .0039, .0051, .0038, .0049, .0064, ABCD0734
3 .0089, .0078, .008, .0073, .0036, 4 * 0.0, .0035, .0054, .0071,  ABCD0735
4 .0061, .0071, .0087, .0115, .0102, .0096, .0104, .0106, 4 * 0.0, ABCD0736
5 .0044, .0069, .0087, .0078, .0092, .0113, .0141, .0124, .0119,   ABCD0737
6 .0122, .0123, 4 * 0.0, .0051, .008, .0099, .0096, .0119, .013,   ABCD0738
7 .0162, .0153, .0141, .0139, .0141, 4 * 0.0, .0056, .0087, .0111, ABCD0739
8 .011, .0141, .0152, .0181, .0177, .016, .0157, .0159, 4 * 0.0,   ABCD0740
9 .0059, .0096, .0122, .0126, .0155, .0171, .0197, .0201, .0169,   ABCD0741
1 .0181, .0172, 4 * 0.0, .0061, .0101, .0134, .0141, .0172, .0187, ABCD0742
2 .0216, .0226, .0188, .0201, .0186, 4 * 0.0, .0062, .0104, .0143, ABCD0743
3 .0159, .0181, .0209, .0235, .0242, .0209, .0217, .0201, 4 * 0.0, ABCD0744
4 .0061, .0107, .0152, .0166, .0192, .0226, .025, .0261, .0223,    ABCD0745
5 .023, 5 * 0.0, .0057, .0108, .0157, .0174, .0202, .0244, .0265,  ABCD0746
6 .0274, 2 * .0244, 5 * 0.0, .0053, .0106, .0162, .0183, .0214,    ABCD0747
```

```
7 .0259, .0284, .0285, .0263, .0251, 5 * 0.0, .0044, .0101, .0166, ABCD0748
8 .0191, .0266, .0286, .0296, .0299, .0279, 6 * 0.0, .0035, .0094, ABCD0749
9 .0167, .0195, .0235, .0303, 2 * .0314, .0289, 6 * 0.0, .0001,    ABCD0750
1 .0077, .0164, .0197, .0239, .0319, .0329, .0321, .0303, 5 * 0.0 /ABCD0751
 DATA ETATXL /                                                     ABCD0752
1 .712, 2 * .8, .7995, .775, .76, .731, .71, .678, .638, .6,       ABCD0753
2 4 * 0.0, .73, .81, .83, 4 * .8, .745, .7, .655, .6, 4 * 0.0,     ABCD0754
3 .7472, .82, .86, .84, .848, .845, .83, .768, .7125, .67, .612,   ABCD0755
4 4 * 0.0, .73, .83, .863, 4 * .86, .8, .735, .685, .617, 4 * 0.0, ABCD0756
5 .714, .83, .867, .868, .875, .873, .875, .838, .769, .7, .621,   ABCD0757
6 4 * 0.0, .7, .829, .87, .873, 3 * .89, .86, .8, .711, .6258,     ABCD0758
7 4 * 0.0, .685, .81, .872, .88, .8912, .895, .893, .8712, .806,   ABCD0759
8 .718, .625, 4 * 0.0, .673, .8, .872, .883, .894, .9, .8975, .878,ABCD0760
9 .8225, .718, .623, 4 * 0.0, .6452, .785, .87, .8835, .8955,      ABCD0761
1 .9005, .8999, .88, .8395, .717, .6009, 4 * 0.0, .52, .76, .867,  ABCD0762
2 .883, .897, .901, .9, .8775, .845, .714, 5 * 0.0, .6, .745, .86, ABCD0763
3 .88, .8961, .9004, .898, .875, .847, .7, 5 * 0.0, .575, .7, .85, ABCD0764
4 .874, .89, .9, .8937, .8722, .8445, .689, 5 * 0.0, .531, .68,    ABCD0765
5 .83, .86, .879, 2 * .89, .866, .833, 6 * 0.0, .5, .645, .8, .835,ABCD0766
6 .8671, .88, .8799, .86, .8235, 6 * 0.0, .385, .59, .76, .82, .86,ABCD0767
7 .8735, .871, .848, .808, 5 * 0.0 /                               ABCD0768
 END                                                               ABCD0769
```

Subroutine COAFBN

```
 SUBROUTINE COAFBN                                                 ABCD0770
 IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD0771
 LOGICAL  SI                                                       ABCD0772
 COMMON /COMALL/  COM(1062)                                        ABCD0773
 DIMENSION  WORD(2)                                                ABCD0774
 DIMENSION  Q(9), AWORD(2)                                         ABCD0775
 EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (IGASMX, COM(10)),ABCD0776
1 (IAFTBN, COM(12)), (TOLALL,COM(23)), (WG24, COM(321)), (FAR24,   ABCD0777
2 COM(322)), (WG55, COM(323)), (FAR55, COM(324)), (P6DSAV,         ABCD0778
3 COM(325)), (AM6DSV, COM(326)), (AM6, COM(327)), (A6, COM(328)),  ABCD0779
4 (WG6CDS, COM(329)), (WG6, COM(330)), (T6, COM(331)),             ABCD0780
5 (P6, COM(332)), (H6, COM(333)), (WG7, COM(334)),                 ABCD0781
6 (FAR7, COM(335)), (FAR7SV, COM(336)), (TS7, COM(337)),           ABCD0782
7 (PS7, COM(338)), (V7, COM(339)), (AM7, COM(340)), (A7, COM(341)),ABCD0783
8 (T7DS, COM(342)), (T7, COM(343)), (H7, COM(344)), (S7, COM(345)),ABCD0784
9 (ETAADS, COM(366)), (DPAFDS, COM(367)), (WFA, COM(368)),         ABCD0785
1 (ETAA, COM(369)), (ETAASV, COM(370)), (DPAFT, COM(371)),         ABCD0786
2 (P7, COM(389)), (U7, COM(390)), (VAFTBN, COM(402)),              ABCD0787
3 (ICOAFB, COM(1045)), (SI, COM(1055))                             ABCD0788
 DATA  AWORD /4HCOAF, 4HBN  /                                      ABCD0789
 WORD(1) = AWORD(1)                                                ABCD0790
 WORD(2) = AWORD(2)                                                ABCD0791
 Q(2) = 0.0D0                                                      ABCD0792
 Q(3) = 0.0D0                                                      ABCD0793
 IF (SI) GO TO 100                                                 ABCD0794
 AJ = 778.26D0                                                     ABCD0795
 AJX = 2.719D0                                                     ABCD0796
 CAPSF = 2116.2170D0                                               ABCD0797
 G = 32.174049D0                                                   ABCD0798
```

```
      PRATM = 14.696D0                                                  ABCD0799
      TDEL = 2000.0D0                                                   ABCD0800
      T7MAX = 4000.0D0                                                  ABCD0801
      RA = .0252D0                                                      ABCD0802
      GO TO 101                                                         ABCD0803
100   AJ = 1.0D0                                                        ABCD0804
      AJX = 1.0D0                                                       ABCD0805
      CAPSF = 101325.0D0                                                ABCD0806
      G = 1.0D0                                                         ABCD0807
      PRATM = 14.696D0 / 101324.6D0                                     ABCD0808
      TDEL = 1111.0D0                                                   ABCD0809
      T7MAX = 2222.0D0                                                  ABCD0810
      RA = 286.9D0                                                      ABCD0811
101   GAJ2 = 2.0D0* G * AJ                                              ABCD0812
      ICOAFB = 0                                                        ABCD0813
C***  P6DS AND AM6DS ARE SET FOR GENERALIZATION OF AFTERBURNER          ABCD0814
C***  EFFICIENCY MAP GENERALIZATION                                     ABCD0815
      IF (IDES .NE. 1)  GO TO 102                                       ABCD0816
      P6DS = P6 * PRATM                                                 ABCD0817
      AM6DS = AM6                                                       ABCD0818
102   WF6 = FAR55 * WG55 / (FAR55 + 1.0D0)                              ABCD0819
      IF (IGASMX .GT. 0)  WF6 = WF6 + FAR24 * WG24 / (FAR24 + 1.0D0)    ABCD0820
      WA6 = WG6 - WF6                                                   ABCD0821
C ***  DRY LOSS                                                         ABCD0822
      WG6C = WG6 * DSQRT(T6) / P6                                       ABCD0823
      IF (IDES .EQ. 1)  WG6CDS = WG6C                                   ABCD0824
      DPAFT = DPAFDS * (WG6C / WG6CDS)                                  ABCD0825
      IF (DPAFT .GT. 1.0D0)  DPAFT = 1.0D0                              ABCD0826
      P7 = P6 * (1.0D0 - DPAFT)                                         ABCD0827
      A7 = A6                                                           ABCD0828
      FAR6 = WF6 / WA6                                                  ABCD0829
      CALL PPOCOM (FAR6,T6,XX1,XX2,XX3,XX4,PHI6,XX6)                    ABCD0830
      WQA = WG6 / A7                                                    ABCD0831
      C1 = P7 * DSQRT(G / (T6 * AJ)) * CAPSF                            ABCD0832
      AM7 = AM6                                                         ABCD0833
      TS7 = 0.875D0 * T6                                                ABCD0834
1     DO 2  I = 1,50                                                    ABCD0835
      CALL PROCOM (FAR6,TS7,CS7,AK7,CP7,REX7,PHIS7,HS7)                 ABCD0836
      V7 = AM7 * CS7                                                    ABCD0837
      HSCAL = H6 - V7 ** 2 / GAJ2                                       ABCD0838
      DELHS = HSCAL - HS7                                               ABCD0839
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 3              ABCD0840
2     TS7 = TS7 + DELHS / CP7                                           ABCD0841
      ICOAFB = 1                                                        ABCD0842
      GO TO 14                                                          ABCD0843
3     WQAT = C1 * DSQRT(AK7 / REX7) * AM7 / (1.0D0 + (AK7 - 1.0D0) *    ABCD0844
     1 AM7 ** 2 / 2.0D0) ** ((AK7 + 1.0D0) / (2.0D0 * (AK7 - 1.0D0)))   ABCD0845
      DIR = WQA / WQAT                                                  ABCD0846
      EW = (WQA - WQAT) / WQA                                           ABCD0847
      CALL AFQUIR (Q(1),AM7,EW,0.0D0,40.0D0,1.0D0*TOLALL,DIR,AM7T,IGO)  ABCD0848
      ICOAFB = 2                                                        ABCD0849
      GO TO (4,5,14), IGO                                               ABCD0850
4     AM7 = AM7T                                                        ABCD0851
      IF (AM7 .LE. 0.0D0)  AM7 = 1.0D-10                                ABCD0852
```

```
      IF (AM7 .GE. 1.0D0)  AM7 = 0.9D0                                  ABCD0853
      GO TO 1                                                           ABCD0854
5     PS7 = P7 / DEXP((PHI6 - PHIS7) / REX7)                            ABCD0855
      IF (IAFTBN .GT. 0)  GO TO 7                                       ABCD0856
C *** NON-AFTERBURNING                                                  ABCD0857
6     T7 = T6                                                           ABCD0858
      WFA = 0.0D0                                                       ABCD0859
      FAR7 = FAR6                                                       ABCD0860
      WG7 = WG6                                                         ABCD0861
      IF (IDES .NE. 1 .OR. T7DS .EQ. 0.0D0)  GO TO 20                   ABCD0862
C *** AFTERBURNING                                                      ABCD0863
7     IF (IAFTBN .EQ. 2)  T7 = T6 + TDEL                                ABCD0864
      IF (IDES .EQ. 1)  T7 = T7DS                                       ABCD0865
      IF (T7 .LE. T6)  GO TO 6                                          ABCD0866
      RHO65 = CAPSF * PS7 / (AJ * REX7 * TS7)                           ABCD0867
      PS65 = PS7                                                        ABCD0868
      V65 = V7                                                          ABCD0869
      Q(2) = 0.0D0                                                      ABCD0870
      Q(3) = 0.0D0                                                      ABCD0871
8     IF (T7 .GT. T7MAX)  T7 = T7MAX                                    ABCD0872
      IF (T7 .LT. T6)  T7 = T6 * 1.001D0                                ABCD0873
      IF (SI)  T7 = T7 * 9.0D0 / 5.0D0                                  ABCD0874
      HV = ((((( - .4594317D-19 * T7) - .2034116D-15) * T7 +            ABCD0875
     1 .2783643D-11) * T7 + .2051501D-07) * T7 - .2453116D-03) * T7 -   ABCD0876
     2 .9433296D-01) * T7 + .1845537D+05                                ABCD0877
      IF (.NOT. SI)  GO TO 103                                          ABCD0878
      T7 = T7 * 5.0D0 / 9.0D0                                           ABCD0879
      HV = HV * 2325.4295D0                                             ABCD0880
103   CALL THERMO (P7,HA,T7,XX1,XX2,1,FAR6,0)                           ABCD0881
C***  TO ALTER DESIGN ABETAA MAP FROM GENERAL TO SPECIFIC MAP           ABCD0882
      IF (IDES .NE. 1)  GO TO 9                                         ABCD0883
      FAR7DS = (HA - H6) / (HV * ETAADS)                                ABCD0884
      CALL ETAAB (0.0D0,0.0D0,0.0D0,0.0D0,ETAADS,ETAASV,P6DS,P6DSAV,    ABCD0885
     1 AM6DS,AM6DSV,IDES,FAR7DS,FAR7SV)                                 ABCD0886
      T7 = T6                                                           ABCD0887
      GO TO 20                                                          ABCD0888
9     P6GS = P6 * PRATM                                                 ABCD0889
      FAR7GS = (HA - H6) / (HV * ETAADS)                                ABCD0890
      DO 10  II = 1,50                                                  ABCD0891
      CALL ETAAB (FAR7GS,AM6,P6GS,ETAA,ETAADS,ETAASV,P6DS,P6DSAV,AM6DS, ABCD0892
     1 AM6DSV,IDES,FAR7DS,FAR7SV)                                       ABCD0893
      FAR7 = (HA - H6) / (HV * ETAA)                                    ABCD0894
      DELFA7 = DABS(FAR7 - FAR7GS)                                      ABCD0895
      IF (DELFA7 .LE. 1.0D0 * TOLALL * FAR7)  GO TO 11                  ABCD0896
10    FAR7GS = FAR7                                                     ABCD0897
11    IF (FAR7 .GT. 0.0D0)  GO TO 12                                    ABCD0898
      ICOAFB = 3                                                        ABCD0899
      CALL ERROR                                                        ABCD0900
12    WFAX = FAR7 * WG6                                                 ABCD0901
      IF (IAFTBN .EQ. 1)  GO TO 15                                      ABCD0902
      ERRW = (WFA - WFAX) / WFA                                         ABCD0903
      DIR = DSQRT(WFA / WFAX)                                           ABCD0904
      CALL AFQUIR (Q(1),T7,ERRW,0.0D0,30.0D0,.5D0*TOLALL,DIR,T7T,IGO)   ABCD0905
      ICOAFB = 4                                                        ABCD0906
```

```
      GO TO (13,16,14), IGO                                                    ABCD0907
13    T7 = T7T                                                                 ABCD0908
      GO TO 8                                                                  ABCD0909
14    CALL ERROR                                                               ABCD0910
15    WFA = WFAX                                                               ABCD0911
16    FAR7 = (WF6 + WFA) / WA6                                                 ABCD0912
      WG7 = WG6 + WFA                                                          ABCD0913
C *** MOMENTUM LOSS                                                            ABCD0914
      CALL PROCOM (FAR7,T7,XX1,XX2,XX3,REX7,PHI7,H7)                           ABCD0915
      RHO7 = CAPSF * P7 / (AJ * REX7 * T7)                                     ABCD0916
      V7 = WG7 / (RHO7 * A7)                                                   ABCD0917
      Q(2) = 0.0D0                                                             ABCD0918
      Q(3) = 0.0D0                                                             ABCD0919
      PS7 = PS65 - 0.01D0                                                      ABCD0920
17    RHO7 = WG7 / (V7 * A7)                                                   ABCD0921
      HS7 = H7 - V7 ** 2 / GAJ2                                                ABCD0922
      CALL THERMO (1.0D0,HS7,TS7,PHIS7,XX2,1,FAR7,1)                           ABCD0923
      IF (TS7 .GE. 301.0D0)  GO TO 18                                          ABCD0924
      CALL THERMO (1.0D0,HS7,400.0D0,PHIS7,XX2,1,FAR7,0)                       ABCD0925
      V7 = DSQRT(GAJ2 * (H7 - HS7))                                            ABCD0926
      GO TO 17                                                                 ABCD0927
18    PS7 = RHO7 * AJ * REX7 * TS7 / CAPSF                                     ABCD0928
      PS7A = PS65 + (RHO65 * V65 ** 2 - RHO7 * V7 ** 2) / (G * CAPSF)          ABCD0929
      DIR = DSQRT(DABS(PS7 / PS7A))                                            ABCD0930
      EP = (PS7 - PS7A) / PS7                                                  ABCD0931
      CALL AFQUIR (Q(1),V7,EP,0.0D0,50.0D0,1.0D0*TOLALL,DIR,V7T,IGO)           ABCD0932
      V7 = V7T                                                                 ABCD0933
      IF (V7 .LT. 100.0D0)  V7 = 100.0D0                                       ABCD0934
      ICOAFB = 5                                                               ABCD0935
      GO TO (17,19,14), IGO                                                    ABCD0936
19    P7 = PS7 * DEXP((PHI7 - PHIS7) / REX7)                                   ABCD0937
      CALL PROCOM (FAR7,TS7,CS7,XX2,XX3,XX4,XX5,XX6)                           ABCD0938
      AM7 = V7 / CS7                                                           ABCD0939
20    CALL THERMO (P7,H7,T7,S7,XX2,1,FAR7,0)                                   ABCD0940
      IF (VAFTBN .EQ. 0.0D0)  GO TO 31                                         ABCD0941
      Q(2) = 0.0D0                                                             ABCD0942
      Q(3) = 0.0D0                                                             ABCD0943
      WG7P = WG7                                                               ABCD0944
      H7P = H7                                                                 ABCD0945
      P7DOT = DERIV(18,P7)                                                     ABCD0946
28    CALL THERMO(P7,H7,T7,S7,XX2,1,FAR7,0)                                    ABCD0947
      WG7 = WG7P - P7DOT * VAFTBN / T7 / (1.4D0 * RA)                          ABCD0948
      U7 = H7 - AJX * RA * T7                                                  ABCD0949
      U7DOT = DERIV(19,U7)                                                     ABCD0950
      H7X = (WG7P * H7P - (WG7P - WG7) * U7 - U7DOT * P7 * VAFTBN /            ABCD0951
     1 T7 / RA) / WG7                                                          ABCD0952
      ERRW = (H7 - H7X) / H7                                                   ABCD0953
      DIR = DSQRT(DABS(H7 / H7X))                                              ABCD0954
      CALL AFQUIR (Q(1),T7,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T7T,IGO)          ABCD0955
      ICOAFB = 6                                                               ABCD0956
      GO TO (29,31,30), IGO                                                    ABCD0957
29    T7 = T7T                                                                 ABCD0958
      GO TO 28                                                                 ABCD0959
30    CALL ERROR                                                               ABCD0960
```

```
31    ICOAFB = 0                                                        ABCD0961
      CALL COMNOZ                                                       ABCD0962
      RETURN                                                            ABCD0953
C                                                                       ABCD0964
C                                                                       ABCD0965
      END                                                               ABCD0966
```

Subroutine COCOMB

```
      SUBROUTINE COCOMB                                                 ABCD0967
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD0968
      LOGICAL  SI, FXM2CP                                               ABCD0969
      COMMON /COMALL/  COM(1062)                                        ABCD0970
      COMMON /COMDAT/  COMD(5423)                                       ABCD0971
      DIMENSION  WORD(2), PSIXB(15), DELXB(15,15), ETAXB(15,15),        ABCD0972
     1 NPTB(15)                                                         ABCD0973
      DIMENSION  Q(9), DUMBO(15,15), AWORD(2)                           ABCD0974
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MODE, COM(6)),   ABCD0975
     1 (MAPEDG, COM(22)), (TOLALL, COM(23)), (T3, COM(152)), (H3,       ABCD0976
     2 COM(153)), (WA3, COM(154)), (WA3CDS, COM(155)), (T4, COM(156)),  ABCD0977
     3 (H4, COM(157)), (S4, COM(158)), (WG4, COM(159)),                 ABCD0978
     4 (FAR4, COM(160)), (T50, COM(161)), (H50, COM(162)),              ABCD0979
     5 (S50, COM(163)), (WG50, COM(164)), (FAR50, COM(165)),            ABCD0980
     6 (PCBLHP, COM(166)), (PCBLIP, COM(167)), (PCBLLP, COM(168)),      ABCD0981
     7 (PCBLDU, COM(169)), (PCBLOB, COM(170)), (CNHP, COM(171)),        ABCD0982
     8 (ETATHP, COM(172)), (DHTCHP, COM(173)), (DHTC, COM(174)),        ABCD0983
     9 (TFFHP, COM(175)), (ETABCF, COM(179)), (ETABDS, COM(184)),       ABCD0984
     1 (WFBDS, COM(185)), (ETAB, COM(190)), (WAC, COM(191)),            ABCD0985
     2 (WFB, COM(192)), (BLC, COM(193)), (DPCODS, COM(197)),            ABCD0986
     3 (DPCOM, COM(198)), (P3, COM(379)), (P4, COM(381)),               ABCD0987
     4 (U4, COM(382)), (P50, COM(383)), (VCOMB, COM(398)),              ABCD0988
     5 (ISPOOL, COM(1044)), (ITRAN, COM(1049)), (SI, COM(1055)),        ABCD0989
     6 (FXM2CP, COM(1059))                                              ABCD0990
      EQUIVALENCE  (PSIXB(1), COMD(2761)), (DELXB(1,1), COMD(2776)),    ABCD0991
     1 (ETAXB(1,1), COMD(3001)), (NPSB, COMD(5360)),                    ABCD0992
     2 (NPTB(1), COMD(5361))                                            ABCD0993
      DATA  AWORD /4HCOCO, 4HMB  /                                      ABCD0994
      WORD(1) = AWORD(1)                                                ABCD0995
      WORD(2) = AWORD(2)                                                ABCD0996
      IF(SI) GO TO 100                                                  ABCD0997
      RA = .0252D0                                                      ABCD0998
      AJ = 2.719D0                                                      ABCD0999
      TMAX = 4000.0D0                                                   ABCD1000
      TMIN = 1000.0D0                                                   ABCD1001
      GO TO 101                                                         ABCD1002
100   RA = 286.9D0                                                      ABCD1003
      AJ = 1.0D0                                                        ABCD1004
      TMAX = 2222.0D0                                                   ABCD1005
      TMIN = 555.5D0                                                    ABCD1006
101   Q(2) = 0.0D0                                                      ABCD1007
      Q(3) = 0.0D0                                                      ABCD1008
      P3PSI = 14.696D0 * P3                                             ABCD1009
      IF (SI)  P3PSI = .14504D-3 * P3                                   ABCD1010
      WA3C = WA3 * DSQRT(T3) / P3PSI                                    ABCD1011
```

```
      IF (SI)  WA3C = WA3 * DSQRT(T3) / P3                                  ABCD1012
      IF (IDES .EQ. 1)  WA3CDS = WA3C                                       ABCD1013
      DPCOM = DPCODS * (WA3C / WA3CDS)                                      ABCD1014
      IF (DPCOM .GT. 1.0D0)  DPCOM = 1.0D0                                  ABCD1015
      P4 = P3 * (1.0D0 - DPCOM)                                             ABCD1016
      IF (IDES .EQ. 1 .AND. MODE .EQ. 2)  T4 = (TMAX + TMIN) / 2.0          ABCD1017
      IF (ITRAN .EQ. 1 .AND. MODE .EQ. 2)  CALL FCNTRL                      ABCD1018
1     IF (T4 .GT. TMAX)  T4 = TMAX                                          ABCD1019
      IF (T4 .GE. TMIN)  GO TO 2                                            ABCD1020
      T4 = TMIN                                                             ABCD1021
      IF (MODE .EQ. 1)  MAPEDG = 1                                          ABCD1022
2     DTCO = T4 - T3                                                        ABCD1023
      IF (SI)  DTCO = DTCO * 9.0D0 / 5.0D0                                  ABCD1024
      P3PSIN = P3PSI                                                        ABCD1025
      CALL SEARCH (-1.0D0,P3PSIN,DTCO,ETAB,DUMMY,PSIXB(1),NPSB,             ABCD1026
     1 DELXB(1,1),ETAXB(1,1),DUMBO(1,1),NPTB(1),15,15,IGO)                  ABCD1027
      IF (IGO .EQ. 7)  CALL ERROR                                           ABCD1028
      IF (IDES .NE. 1)  GO TO 4                                             ABCD1029
      ETABCF = ETABDS / ETAB                                                ABCD1030
4     ETAB = ETABCF * ETAB                                                  ABCD1031
      IF (SI)  T4 = T4 * 9.0D0 / 5.0D0                                      ABCD1032
      HV = ((((( - .4594317D-19 * T4) - .2034116D-15) * T4 +                ABCD1033
     1 .2783643D-11) * T4 + .2051501D-07) * T4 - .2453116D-03) * T4 -       ABCD1034
     2 .9433296D-01) * T4 + .1845537D+05                                    ABCD1035
      IF (.NOT. SI)  GO TO 3                                                ABCD1036
      T4 = T4 * 5.0D0 / 9.0D0                                               ABCD1037
      HV = HV * 2325.4295D0                                                 ABCD1038
3     CALL THERMO (P4,HA,T4,XX1,XX2,0,0.0D0,0)                              ABCD1039
      FAR4 = (HA - H3) / (HV * ETAB)                                        ABCD1040
      IF (FAR4 .LT. 0.0D0)  FAR4 = 0.0D0                                    ABCD1041
      WFBX = FAR4 * WA3                                                     ABCD1042
      IF (MODE .NE. 2)  GO TO 7                                             ABCD1043
      ERRW = (WFB - WFBX) / WFB                                             ABCD1044
      DIR = DSQRT(WFB / WFBX)                                               ABCD1045
      CALL AFQUIR (Q(1),T4,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T4T,IGO)       ABCD1046
      GO TO (5,8,6), IGO                                                    ABCD1047
5     T4 = T4T                                                              ABCD1048
      GO TO 1                                                               ABCD1049
6     CALL ERROR                                                            ABCD1050
7     WFB = WFBX                                                            ABCD1051
      IF (IDES .EQ. 1)  WFBDS = WFB                                         ABCD1052
8     CALL THERMO (P4,H4,T4,S4,XX2,1,FAR4,0)                                ABCD1053
      WG4 = WFB + WA3                                                       ABCD1054
      IF (VCOMB .EQ. 0.0D0)  GO TO 21                                       ABCD1055
      Q(2) = 0.0D0                                                          ABCD1056
      Q(3) = 0.0D0                                                          ABCD1057
      WG4P = WG4                                                            ABCD1058
      H4P = H4                                                              ABCD1059
      P4DOT = DERIV(10,P4)                                                  ABCD1060
18    CALL THERMO(P4,H4,T4,S4,XX2,1,FAR4,0)                                 ABCD1061
      WG4 = WG4P - P4DOT * VCOMB / T4 / 1.4D0 / RA                          ABCD1062
      U4 = H4 - AJ * RA * T4                                                ABCD1063
      U4DOT = DERIV(11,U4)                                                  ABCD1064
      H4X = (WG4P * H4P - (WG4P - WG4) * U4 - U4DOT * P4 * VCOMB / T4 /     ABCD1065
```

```
     1 RA) / WG4                                                          ABCD1066
       ERRW = (H4 - H4X) / H4                                             ABCD1067
       DIR = DSQRT(DABS(H4 / H4X))                                        ABCD1068
       CALL AFQUIR (Q(1),T4,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T4T,IGO)    ABCD1069
       GO TO (19,21,20), IGO                                              ABCD1070
19     T4 = T4T                                                           ABCD1071
       GO TO 18                                                           ABCD1072
20     CALL ERROR                                                         ABCD1073
21     PCBLUS = 1.0 - PCBLHP - PCBLIP - PCBLLP - PCBLOB - PCBLDU          ABCD1074
       IF (PCBLUS .LT. .00001D0)  GO TO 22                                ABCD1075
       BLUS = BLC * PCBLUS                                                ABCD1076
       IF (BLUS .LT. (.0000001D0 * WA3))  GO TO 22                        ABCD1077
       FAR4 = FAR4 * WG4 / (WG4 + BLUS * (1.0D0 + FAR4))                  ABCD1078
       H4 = (H4 * WG4 + H3 * BLUS) / (WG4 + BLUS)                         ABCD1079
       WG4 = WG4 + BLUS                                                   ABCD1080
       CALL THERMO  (P4,H4,T4,S4,XX2,1,FAR4,1)                            ABCD1081
22     IF (IDES .EQ. 1)  WRITE (6,10) WA3CDS,ETABCF                       ABCD1082
       IF (FXM2CP .OR. ISPOOL .EQ. 1)  GO TO 9                            ABCD1083
       CALL COHPTB                                                        ABCD1084
       RETURN                                                             ABCD1085
9      P50 = P4                                                           ABCD1086
       H50 = H4                                                           ABCD1087
       T50 = T4                                                           ABCD1088
       S50 = S4                                                           ABCD1089
       FAR50 = FAR4                                                       ABCD1090
       WG50 = WG4                                                         ABCD1091
C      SET HIGH PRESSURE TURBINE PARAMETERS TO ZERO, NOT USED             ABCD1092
       TFFHP = 0.0D0                                                      ABCD1093
       CNHP = 0.0D0                                                       ABCD1094
       DHTC = 0.0D0                                                       ABCD1095
       DHTCHP = 0.0D0                                                     ABCD1096
       ETATHP = 0.0D0                                                     ABCD1097
       IF (FXM2CP)  CALL COIPTB                                           ABCD1098
C      IF RUNNING 1 SPOOL TJ GO TO COHPTB TO ZERO OUT COIPTB              ABCD1099
       IF (.NOT. FXM2CP)  CALL COHPTB                                     ABCD1100
       RETURN                                                             ABCD1101
C                                                                         ABCD1102
C                                                                         ABCD1103
C                                                                         ABCD1104
10     FORMAT (17H0COMBUSTOR DESIGN, 7X, 8H WA3CDS=, E15.8, 8H ETABCF=,   ABCD1105
     1 E15.8)                                                             ABCD1106
       END                                                                ABCD1107
```

Subroutine COCOMP

```
       SUBROUTINE COCOMP                                                  ABCD1108
       IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD1109
       LOGICAL  SI, DUMSPL, FXM2CP, AFTFAN, FAN                           ABCD1110
       COMMON /COMALL/  COM(1062)                                         ABCD1111
       COMMON /COMDAT/  COMD(5423)                                        ABCD1112
       DIMENSION  WORD(2), ERR(9), CNXP(15), PRXP(15,15), WA3XP(15,15),   ABCD1113
     1 ETAXP(15,15), NPTP(15)                                             ABCD1114
       DIMENSION  Q(9), WLH(2,2), AWORD(2)                                ABCD1115
       EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MODE, COM(6)),    ABCD1116
```

```
     1 (MAPEDG, COM(22)), (TOLALL, COM(23)), (ERR(1), COM(24)), (T2,     ABCD1117
     2 COM(92)), (T3, COM(152)), (H3, COM(153)), (WA3, COM(154)),       ABCD1118
     3 (PCBLHP, COM(166)), (PCBLIP, COM(167)), (PCBLLP, COM(168)),      ABCD1119
     4 (PCBLDU, COM(169)), (PCBLOB, COM(170)), (WAC, COM(191)),         ABCD1120
     5 (BLC, COM(193)), (T21, COM(263)), (H21, COM(264)),               ABCD1121
     6 (S21, COM(265)), (WA21, COM(266)), (T21DS, COM(267)),            ABCD1122
     7 (T22, COM(268)), (WA22, COM(269)), (S3, COM(270)),               ABCD1123
     8 (WA32, COM(271)), (PRCCF, COM(294)), (ETACCF, COM(295)),         ABCD1124
     9 (WACCF, COM(296)), (PRCDS, COM(297)), (ETACDS, COM(298)),        ABCD1125
     1 (WACDS, COM(299)), (ZC, COM(300)), (PCNC, COM(301)),             ABCD1126
     2 (PCBLC, COM(302)), (PCNCDS, COM(303)), (PCBLI, COM(304)),        ABCD1127
     3 (PCBLID, COM(305)), (CNC, COM(306)), (PRC, COM(307)),            ABCD1128
     4 (ETAC, COM(308)), (CNI, COM(309)), (WACI, COM(310)),             ABCD1129
     5 (WAI, COM(311)), (BLI, COM(312)), (BLHP, COM(313)),              ABCD1130
     6 (BLIP, COM(314)), (BLLP, COM(315)), (BLF, COM(316)),             ABCD1131
     7 (BLDU, COM(317)), (BLOB, COM(318)), (WAF, COM(319)),             ABCD1132
     8 (WACC, COM(320)), (P21, COM(377)), (P3, COM(379)),               ABCD1133
     9 (U3, COM(380)), (VCOMP, COM(397)), (WACP, COM(422)),             ABCD1134
     1 (ISPOOL, COM(1044)), (SI, COM(1055)), (DUMSPL, COM(1057))        ABCD1135
      EQUIVALENCE  (FXM2CP, COM(1059)), (AFTFAN, COM(1060)),            ABCD1136
     1 (FAN, COM(1061))                                                 ABCD1137
      EQUIVALENCE  (CNXP(1), COMD(2071)), (PRXP(1,1), COMD(2086)),      ABCD1138
     1 (WACXP(1,1), COMD(2311)), (ETAXP(1,1), COMD(2536)),              ABCD1139
     2 (NCNP, COMD(5344)), (NPTP(1), COMD(5345))                        ABCD1140
      DATA  AWORD, WLH /4HCOCO, 4HMP  , 4H (LO, 4H)    , 4H (HI, 4H)    / ABCD1141
      WORD(1) = AWORD(1)                                                ABCD1142
      WORD(2) = AWORD(2)                                                ABCD1143
      IF (SI) GO TO 100                                                 ABCD1144
      TSTD = 518.668D0                                                  ABCD1145
      PSTD = 1.0D0                                                      ABCD1146
      RA = .0252D0                                                      ABCD1147
      AJ = 2.719D0                                                      ABCD1148
      GO TO 101                                                         ABCD1149
100   TSTD = 288.149D0                                                  ABCD1150
      PSTD = 101325.0D0                                                 ABCD1151
      RA = 286.9D0                                                      ABCD1152
      AJ = 1.0D0                                                        ABCD1153
101   THETA = DSQRT(T21 / TSTD)                                         ABCD1154
      DELTA = P21 / PSTD                                                ABCD1155
      IF (IDES .NE. 1 .AND. .NOT. FXM2CP)  GO TO 2                      ABCD1156
      IF (IDES .NE. 1)  GO TO 1                                         ABCD1157
      WACDS = WAC                                                       ABCD1158
      WACC = WAC * THETA / DELTA                                        ABCD1159
      IF (FXM2CP)  GO TO 1                                              ABCD1160
      PCNC = PCNCDS * THETA                                             ABCD1161
      GO TO 2                                                           ABCD1162
C     SPEEDS OF MIDDLE AND INNER SPOOL ARE THE SAME                     ABCD1163
1     SPDMID = CNI * DSQRT(T22 / TSTD)                                  ABCD1164
      IF (AFTFAN)  SPDMID = CNI * DSQRT(T2 / TSTD)                      ABCD1165
      CNC = SPDMID / THETA                                              ABCD1166
      PCNC = 100.0D0 * THETA * CNC                                      ABCD1167
      IF (IDES .EQ. 1)  PCNCDS = PCNC / THETA                           ABCD1168
2     CNC = PCNC / (100.0D0 * THETA)                                    ABCD1169
      IF (ZC .LT. 0.0D0)  ZC = 0.0D0                                    ABCD1170
```

```
      IF (ZC .GT. 1.0D0)  ZC = 1.0D0                                    ABCD1171
      CNCS = CNC                                                        ABCD1172
      IF (ISPOOL .EQ. 1)  GO TO 12                                      ABCD1173
      CALL SEARCH (ZC,CNC,PRC,WACC,ETAC,CNXP(1),NCNP,PRXP(1,1),         ABCD1174
     1 WACXP(1,1),ETAXP(1,1),NPTP(1),15,15,IGO)                         ABCD1175
      GO TO 13                                                          ABCD1176
12    PRC = 1.0D0                                                       ABCD1177
      ETAC = 1.0D0                                                      ABCD1178
      WAC = WA21                                                        ABCD1179
      WACC = WAC * THETA / DELTA                                        ABCD1180
      CNC = 1.0D0                                                       ABCD1181
      PRCCF = 1.0D0                                                     ABCD1182
13    IF (MODE .EQ. 1)  GO TO 4                                         ABCD1183
      IF ((CNC - CNCS) .GT. .5D0 * TOLALL * CNC)  MAPEDG = 1            ABCD1184
4     IF (IGO .EQ. 1 .OR. IGO .EQ. 2)  WRITE (8,9) CNCS,WLH(1,IGO),     ABCD1185
     1 WLH(2,IGO)                                                       ABCD1186
      WAC = WACC * DELTA / THETA                                        ABCD1187
      IF (IDES .NE. 1)  GO TO 5                                         ABCD1188
      T21DS = T21                                                       ABCD1189
      IF (ISPOOL .GE. 2)  PRCCF = (PRCDS - 1.0D0) / (PRC - 1.0D0)       ABCD1190
      ETACCF = ETACDS / ETAC                                            ABCD1191
      IF (ISPOOL .EQ. 1)  ETACCF = 1.0D0                                ABCD1192
      WACCF = WACDS / WAC                                               ABCD1193
      WRITE (6,10) PRCCF,ETACCF,WACCF,T21DS                             ABCD1194
5     PRC = PRCCF * (PRC - 1.0D0) + 1.0D0                               ABCD1195
      ETAC = ETACCF * ETAC                                              ABCD1196
      WAC = WACCF * WAC                                                 ABCD1197
      WACP = WAC                                                        ABCD1198
      IF (.NOT. DUMSPL .OR. PCBLID .NE. 0.0D0 .OR. .NOT. FAN)  GO TO 6  ABCD1199
      WA22 = WAC                                                        ABCD1200
      WAI = WA22                                                        ABCD1201
      WACI = WACC * WACCF                                               ABCD1202
6     WA32 = WAI - WAC                                                  ABCD1203
      BLI = WA32                                                        ABCD1204
      WA21 = WAC                                                        ABCD1205
      WACC = WACC * WACCF                                               ABCD1206
      PCBLI = BLI / WAI                                                 ABCD1207
      CALL WDUCT1                                                       ABCD1208
      IF (PCBLID .EQ. 0.0D0)  ERR(7) = (WAC - WAI) / WAC                ABCD1209
      IF (.NOT. FAN)  ERR(5) = (WAP - WAC - BLP) / WAC                  ABCD1210
      IF (IDES .EQ. 1 .AND. PCBLID .EQ. 0.0D0)  ERR(7) = 1.0D-4         ABCD1211
      CALL THCOMP (PRC,ETAC,T21,H21,S21,P21,T3,H3,S3,P3)                ABCD1212
      IF (VCOMP .EQ. 0.0D0)  GO TO 21                                   ABCD1213
      Q(2) = 0.0D0                                                      ABCD1214
      Q(3) = 0.0D0                                                      ABCD1215
      H3P = H3                                                          ABCD1216
      P3DOT = DERIV(8,P3)                                               ABCD1217
18    CALL THERMO (P3,H3,T3,S3,XX2,0,0.0D0,0)                           ABCD1218
      WAC = WACP - P3DOT * VCOMP / T3 / 1.4D0 / RA                      ABCD1219
      U3 = H3 - AJ * RA * T3                                            ABCD1220
      U3DOT = DERIV(9,U3)                                               ABCD1221
      H3X = (WACP * H3P - (WACP - WAC) * U3 - U3DOT * P3 * VCOMP / T3 / ABCD1222
     1 RA) / WAC                                                        ABCD1223
      ERRW = (H3 - H3X) / H3                                            ABCD1224
```

```
      DIR = DSQRT(DABS(H3 / H3X))                                          ABCD1225
      CALL AFQUIR (Q(1),T3,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T3T,IGO)      ABCD1226
      GO TO (19,21,20), IGO                                                ABCD1227
19    T3 = T3T                                                             ABCD1228
      GO TO 18                                                             ABCD1229
20    CALL ERROR                                                           ABCD1230
21    IF (PCBLC .GT. 0.0D0)  BLC = PCBLC * WAC                             ABCD1231
      WA3 = WAC - BLC                                                      ABCD1232
      BLDU = PCBLDU * BLC                                                  ABCD1233
      BLOB = PCBLOB * BLC                                                  ABCD1234
      BLHP = PCBLHP * BLC                                                  ABCD1235
      BLIP = PCBLIP * BLC                                                  ABCD1236
      BLLP = PCBLLP * BLC                                                  ABCD1237
      IF (MODE .NE. 1)  GO TO 7                                            ABCD1238
      IF (DABS(CNC - CNCS) .LE. 1.0D0 * TOLALL * CNCS)  GO TO 8            ABCD1239
      WRITE (8,11) CNCS,CNC                                                ABCD1240
      CALL ERROR                                                           ABCD1241
7     PCNC = 100.0D0 * THETA * CNC                                         ABCD1242
8     CALL COCOMB                                                          ABCD1243
      RETURN                                                               ABCD1244
C                                                                          ABCD1245
C                                                                          ABCD1246
C                                                                          ABCD1247
9     FORMAT (19H0* * *  CNC OFF MAP,F10.4,2X,A4,A2,11H* * *$$$$$$)       ABCD1248
10    FORMAT(18H0COMPRESSOR DESIGN,6X,8H  PRCCF=,E15.8,8H ETACCF=,         ABCD1249
     1 E15.8,8H  WACCF=,E15.8,8H  T21DS=,E15.8)                            ABCD1250
11    FORMAT (10H0CNC WAS= ,E15.8,11H  AND NOW= ,E15.8,12H  CHECK PCNC,    ABCD1251
     1 12H INPUT$$$$$$)                                                    ABCD1252
      END                                                                  ABCD1253
```

Subroutine CODUCT

```
      SUBROUTINE CODUCT                                                    ABCD1254
      IMPLICIT  REAL*8  (A-H,O-Z)                                          ABCD1255
      LOGICAL  SI, AFTFAN                                                  ABCD1256
      COMMON /COMALL/  COM(1062)                                           ABCD1257
      DIMENSION  WORD(2), ERR(9)                                           ABCD1258
      DIMENSION  Q(9), AWORD1(2), AWORD2(2)                                ABCD1259
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (IGASMX, COM(10)),ABCD1260
     1 (IDBURN, COM(11)), (IDCD, COM(13)), (IDSHOC, COM(15)),              ABCD1261
     2 (NOZFLT, COM(17)), (TOLALL, COM(23)), (ERR(1), COM(24)), (P1,       ABCD1262
     3 COM(33)), (H22, COM(34)), (AM23, COM(35)), (WA23DS, COM(36)),       ABCD1263
     4 (T23, COM(37)), (P23, COM(38)), (H23, COM(39)), (S23, COM(40)),     ABCD1264
     5 (A24, COM(41)), (T24, COM(42)), (H24, COM(43)), (S24, COM(44)),     ABCD1265
     6 (AM25, COM(45)), (T25, COM(46)), (P25, COM(47)), (H25, COM(48)),    ABCD1266
     7 (S25, COM(49)), (A28, COM(50)), (A28SAV, COM(51)),                  ABCD1267
     8 (AM28, COM(52)), (V28, COM(53)), (TS28, COM(54)),                   ABCD1268
     9 (PS28, COM(55)), (T28, COM(56)), (P28, COM(57)), (H28, COM(58)),    ABCD1269
     1 (S28, COM(59)), (A29, COM(60)), (A29SAV, COM(61)),                  ABCD1270
     2 (AM29, COM(62)), (V29, COM(63)), (TS29, COM(64)),                   ABCD1271
     3 (PS29, COM(65)), (T29, COM(66)), (P29, COM(67)), (H29, COM(68)),    ABCD1272
     4 (S29, COM(69)), (BYPASS, COM(70)), (WAD, COM(71)),                  ABCD1273
     5 (WFD, COM(72)), (ETAD, COM(73)), (DPDUC, COM(74)),                  ABCD1274
     6 (DPDUDS, COM(75)), (H3, COM(153)), (WAC, COM(191)),                 ABCD1275
```

```
     7 (PCBLID, COM(305)), (WAI, COM(311)), (BLF, COM(316)),                ABCD1276
     8 (BLDU, COM(317)), (WAF, COM(319)), (WG24, COM(321)),                 ABCD1277
     9 (FAR24, COM(322)), (P22, COM(375)), (P24, COM(391)),                 ABCD1278
     1 (U24, COM(392)), (VFDUCT, COM(403)), (ICODUC, COM(1046))             ABCD1279
      EQUIVALENCE  (ITRAN, COM(1049)), (SI, COM(1055)),                     ABCD1280
     1 (AFTFAN, COM(1060))                                                  ABCD1281
      DATA  AWORD1, AWORD2 /4HCODU, 4HCT  , 4HDNO7, 4HZL  /                 ABCD1282
      WORD(1) = AWORD1(1)                                                   ABCD1283
      WORD(2) = AWORD1(2)                                                   ABCD1284
      Q(2) = 0.0D0                                                          ABCD1285
      Q(3) = 0.0D0                                                          ABCD1286
      GOGO = 0.0D0                                                          ABCD1287
      IF (SI)  GO TO 100                                                    ABCD1288
      AJ = 778.26D0                                                         ABCD1289
      AJX = 2.719D0                                                         ABCD1290
      CAPSF = 2116.217D0                                                    ABCD1291
      G = 32.174049D0                                                       ABCD1292
      TSID = 518.67D0                                                       ABCD1293
      TDEL = 2000.0D0                                                       ABCD1294
      TMAX = 4000.0D0                                                       ABCD1295
      RA = .0252D0                                                          ABCD1296
      GO TO 101                                                             ABCD1297
  100 AJ = 1.0D0                                                            ABCD1298
      AJX = 1.0D0                                                           ABCD1299
      CAPSF = 101325.0D0                                                    ABCD1300
      G = 1.0D0                                                             ABCD1301
      TSTD = 288.15D0                                                       ABCD1302
      TDEL = 1111.0D0                                                       ABCD1303
      TMAX = 2222.0D0                                                       ABCD1304
      RA = 286.9D0                                                          ABCD1305
  101 GAJ2 = 2.0D0 * G * AJ                                                 ABCD1306
      ICODUC = 0                                                            ABCD1307
      WAX = WAF - WAI - BLF                                                 ABCD1308
      IF (PCBLID .EQ. 0.0D0)  WAX = WAF - WAC - BLF                         ABCD1309
      IF (AFTFAN)  WAX = WAF - BLF                                          ABCD1310
      WAD = WAX + BLDU                                                      ABCD1311
      P23 = P22                                                             ABCD1312
C***  DRY LOSS                                                              ABCD1313
      H23 = (BLDU * H3 + WAX * H22) / WAD                                   ABCD1314
      CALL THERMO (P23,H23,T23,S23,XX2,1,0.0D0,1)                           ABCD1315
      WA23C = WAD * DSQRT(T23) / P23                                        ABCD1316
      IF (IDES .EQ. 1)  WA23DS = WA23C                                      ABCD1317
      BYPASS = (WAF - WAI) / WAI                                            ABCD1318
      IF (AFTFAN)  BYPASS = WAF / WAI                                       ABCD1319
      DPDUC = DPDUDS * (WA23C / WA23DS)                                     ABCD1320
      IF (DPDUC .GT. 1.0D0)  DPDUC = 1.0D0                                  ABCD1321
      P24 = P23 * (1.0D0 - DPDUC)                                           ABCD1322
      CALL PROCOM (0.0D0,T23,XX1,XX2,XX3,XX4,PHI23,XX6)                     ABCD1323
      IF (IGASMX .GT. 0)  IDBURN = 0                                        ABCD1324
      AM24 = AM23                                                           ABCD1325
      TS24 = T23 * 0.875D0                                                  ABCD1326
    1 DO 2  I = 1,50                                                        ABCD1327
      CALL PROCOM (0.0D0,TS24,CS24,AK24,CP24,REX24,PHIS24,HS24)             ABCD1328
      V24 = AM24 * CS24                                                     ABCD1329
```

```
      HSCAL = H23 - V24 ** 2 / GAJ2                                          ABCD1330
      DELHS = HSCAL - HS24                                                   ABCD1331
      IF (DABS(DELHS) .LE. 1.0D0 * TOLALL * HSCAL)  GO TO 3                  ABCD1332
2     TS24 = TS24 + DELHS / CP24                                             ABCD1333
      ICODUC = 1                                                             ABCD1334
      GO TO 11                                                               ABCD1335
3     C1 = P24 * DSQRT(G / (T23 * AJ)) * CAPSF                               ABCD1336
      AK24M1 = AK24 - 1.0D0                                                  ABCD1337
      AK24P1 = AK24 + 1.0D0                                                  ABCD1338
      AKM1 = AK24M1 / 2.0D0                                                  ABCD1339
      AKP1 = AK24P1 / 2.0D0                                                  ABCD1340
      IF (IDES .NE. 1)  GO TO 4                                              ABCD1341
      IF (GOGO .GT. 0.0D0)  GO TO 4                                          ABCD1342
      ASTOA = AKP1 ** (AKP1 / AK24M1) * AM24 * (1.0D0 + AKM1 *               ABCD1343
     1 AM24 ** 2) ** (- AKP1 / AK24M1)                                       ABCD1344
      EQWCR = DSQRT(G * AK24 / REX24 / AJ) / (DSQRT(TSTD) / CAPSF) *         ABCD1345
     1 (2.0D0 / AK24P1) ** (AKP1 / AK24M1)                                   ABCD1346
      WA23CC = WA23C / DSQRT(TSTD)                                           ABCD1347
      A24 = 1.0D0 / ASTOA * WA23CC / EQWCR                                   ABCD1348
      GOGO = 1.0D0                                                           ABCD1349
4     WQA = WAD / A24                                                        ABCD1350
      WQAT = C1 * DSQRT(AK24 / REX24) * AM24 / (1.0D0 + AKM1 * AM24 **       ABCD1351
     1 2) ** (AKP1 / AK24M1)                                                 ABCD1352
      DIR = WQA / WQAT                                                       ABCD1353
      EW = (WQA - WQAT) / WQA                                                ABCD1354
      CALL AFQUIR (Q(1),AM24,EW,0.0D0,30.0D0,1.0D0*TOLALL,DIR,AM24T,IGO)     ABCD1355
      ICODUC = 2                                                             ABCD1356
      GO TO (5,6,11), IGO                                                    ABCD1357
5     AM24 = AM24T                                                           ABCD1358
      IF (AM24 .GT. 1.0D0)  AM24 = 0.5D0                                     ABCD1359
      GO TO 1                                                                ABCD1360
6     PS24 = P24 / DEXP((PHI23 - PHIS24) / REX24)                            ABCD1361
7     IF (IDBURN .GT. 0)  GO TO 8                                            ABCD1362
C***  NON-DUCT BURNING                                                       ABCD1363
      T24 = T23                                                              ABCD1364
      WFD = 0.0D0                                                            ABCD1365
      FAR24 = 0.0D0                                                          ABCD1366
      GO TO 17                                                               ABCD1367
8     IF (IDBURN .EQ. 2)  T24 = T23 + TDEL                                   ABCD1368
9     IF (T24 .GT. TMAX)  T24 = TMAX                                         ABCD1369
      IF (T24 .LT. T23)  T24 = T23                                           ABCD1370
C***  DUCT BURNING                                                           ABCD1371
      RHO42 = CAPSF * PS24 / (AJ * REX24 * TS24)                             ABCD1372
      PS42 = PS24                                                            ABCD1373
      V42 = V24                                                              ABCD1374
      Q(2) = 0.0D0                                                           ABCD1375
      Q(3) = 0.0D0                                                           ABCD1376
      IF (T24 .LT. T23)  T24 = T23 * 1.001D0                                 ABCD1377
C *** IF DESIRED, ENTER CALCULATIONS FOR ETAD HERE                           ABCD1378
      IF (SI)  T24 = T24 * 9.0D0 / 5.0D0                                     ABCD1379
      HV = ((((( - .4594317D-19 * T24) - .2034116D-15) * T24 +               ABCD1380
     1 .2783643D-11) * T24 + .2051501D-07) * T24 - .2453116D-03) * T24       ABCD1381
     2 - .9433296D-01) * T24 + .1845537D+05                                  ABCD1382
      IF (.NOT. SI)  GO TO 102                                               ABCD1383
```

```
      T24 = T24 * 5.0D0 / 9.0D0                                          ABCD1384
      HV = HV * 2325.4295D0                                              ABCD1385
102   CALL THERMO (P24,HA,T24,XX1,XX2,0,0.0D0,0)                         ABCD1386
      FAR24 = (HA - H23) / (HV * ETAD)                                   ABCD1387
      IF (FAR24 .LT. 0.0D0)  FAR24 = 0.0D0                               ABCD1388
      WFDX = FAR24 * WAD                                                 ABCD1389
      IF (IDBURN .NE. 2)  GO TO 12                                       ABCD1390
      ERRW = (WFD - WFDX) / WFD                                          ABCD1391
      DIR = DSQRT(WFD / WFDX)                                            ABCD1392
      CALL AFQUIR (Q(1),T24,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T24T,IGO)  ABCD1393
      ICODUC = 3                                                         ABCD1394
      GO TO (10,13,11), IGO                                              ABCD1395
10    T24 = T24T                                                         ABCD1396
      GO TO 9                                                            ABCD1397
11    CALL ERROR                                                         ABCD1398
12    WFD = WFDX                                                         ABCD1399
C***  MOMENTUM LOSS                                                      ABCD1400
13    WG24 = WFD + WAD                                                   ABCD1401
      CALL PROCOM (FAR24,T24,XX1,XX2,XX3,REX24,PHI24,H24)                ABCD1402
      RHO24 = CAPSF * P24 / (AJ * REX24 * T24)                           ABCD1403
      V24 = WG24 / (RHO24 * A24)                                         ABCD1404
      Q(2) = 0.0D0                                                       ABCD1405
      Q(3) = 0.0D0                                                       ABCD1406
      PS24 = PS42 - 0.01D0                                               ABCD1407
14    RHO24 = WG24 / (V24 * A24)                                         ABCD1408
      HS24 = H24 - V24 ** 2 / GAJ2                                       ABCD1409
      CALL THERMO (1.0D0,HS24,TS24,PHIS24,XX2,1,FAR24,1)                 ABCD1410
      IF (TS24 .GE. 301.0D0)  GO TO 15                                   ABCD1411
      CALL THERMO (1.0D0,HS24,400.0D0,PHIS24,XX2,1,FAR24,1)              ABCD1412
      V24 = DSQRT(GAJ2 * (H24 - HS24))                                   ABCD1413
      GO TO 14                                                           ABCD1414
15    PS24 = RHO24 * AJ * REX24 * TS24 / CAPSF                           ABCD1415
      PS24A = PS42 + (RHO42 * V42 ** 2 - RHO24 * V24 ** 2) / (G * CAPSF) ABCD1416
      DIR = DSQRT(DABS(PS24 / PS24A))                                    ABCD1417
      EP = (PS24 - PS24A) / PS24                                         ABCD1418
      CALL AFQUIR (Q(1),V24,EP,0.0D0,50.0D0,1.0D0*TOLALL,DIR,V24T,IGO)   ABCD1419
      V24 = V24T                                                         ABCD1420
      IF (V24 .LT. 25.0D0)  V24 = 25.0D0                                 ABCD1421
      ICODUC = 4                                                         ABCD1422
      GO TO (14,16,11), IGO                                              ABCD1423
16    P24 = PS24 * DEXP((PHI24 - PHIS24) / REX24)                        ABCD1424
      CALL PROCOM (FAR24,TS24,CS24,XX2,XX3,XX4,XX5,XX6)                  ABCD1425
      AM24 = V24 / CS24                                                  ABCD1426
17    CALL THERMO (P24,H24,T24,S24,XX1,1,FAR24,0)                        ABCD1427
      WG24 = WFD + WAD                                                   ABCD1428
      IF (VFDUCT .EQ. 0.0D0)  GO TO 31                                   ABCD1429
      Q(2) = 0.0D0                                                       ABCD1430
      Q(3) = 0.0D0                                                       ABCD1431
      WG24P = WG24                                                       ABCD1432
      H24P = H24                                                         ABCD1433
      P24DOT = DERIV(20,P24)                                             ABCD1434
25    CALL THERMO (P24,H24,T24,S24,XX2,1,FAR24,0)                        ABCD1435
      WG24 = WG24P - P24DOT * VFDUCT / T24 / (1.4D0 * RA)                ABCD1436
      U24 = H24 - AJX * RA * T24                                         ABCD1437
```

```
      U24DOT = DERIV(21,U24)                                                    ABCD1438
      H24X = (WG24P * H24P - (WG24P - WG24) * U24 - U24DOT * P24 *              ABCD1439
     1 VFDUCT / T24 / RA) / WG24                                                ABCD1440
      ERRW = (H24 - H24X) / H24                                                 ABCD1441
      DIR = DSQRT(DABS(H24 / H24X))                                             ABCD1442
      CALL AFQUIR (Q(1),T24,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T24T,IGO)         ABCD1443
      ICODUC = 5                                                                ABCD1444
      GO TO (29,31,30), IGO                                                     ABCD1445
29    T24 = T24T                                                                ABCD1446
      GO TO 28                                                                  ABCD1447
30    CALL ERROR                                                                ABCD1448
31    T25 = T24                                                                 ABCD1449
      P25 = P24                                                                 ABCD1450
      H25 = H24                                                                 ABCD1451
      S25 = S24                                                                 ABCD1452
      AM25 = AM24                                                               ABCD1453
      IF (IGASMX .GT. 0)  GO TO 21                                              ABCD1454
      WORD(1) = AWORD2(1)                                                       ABCD1455
      WORD(2) = AWORD2(2)                                                       ABCD1456
      A28SAV = A28                                                              ABCD1457
      A29SAV = A29                                                              ABCD1458
      NOZD = 0                                                                  ABCD1459
      IDNOZ = 0                                                                 ABCD1460
      IF (NOZFLT .EQ. 2 .OR. NOZFLT .EQ. 3)  NOZD = 1                           ABCD1461
      IF (IDES .EQ. 1 .OR. IDBURN .GT. 0 .OR. NOZF .EQ. 1)  IDNOZ = 1           ABCD1462
      IF (ITRAN .EQ. 1)  IDNOZ = 0                                              ABCD1463
      IF (IDCD .EQ. 1)  GO TO 18                                                ABCD1464
      CALL CONVRG (T25,H25,P25,S25,FAR24,WG24,P1,IDNOZ,A28,P25R,T24,ICON,       ABCD1465
     1 P28,S28,TS28,PS28,V28,AM28,ICON)                                         ABCD1466
      GO TO (19,19,19,11), ICON                                                 ABCD1467
18    CALL CONDIV (T25,H25,P25,S25,FAR24,WG24,P1,IDNOZ,A28,P25R,T24,            ABCD1468
     1 H28,P28,S28,T28,H29,P29,S29,TS29,PS29,P28R,P29,V28,V29,AM28,             ABCD1469
     2 AM29,ICON)                                                               ABCD1470
      IDSHOC = ICON                                                             ABCD1471
      ICODUC = 6                                                                ABCD1472
      GO TO (20,20,20,11), ICON                                                 ABCD1473
19    T29 = T28                                                                 ABCD1474
      H29 = H28                                                                 ABCD1475
      P29 = P28                                                                 ABCD1476
      S29 = S28                                                                 ABCD1477
      TS29 = TS28                                                               ABCD1478
      PS29 = PS28                                                               ABCD1479
      V29 = V28                                                                 ABCD1480
      AM29 = AM28                                                               ABCD1481
      A29 = A29                                                                 ABCD1482
      IDSHOC = ICON + 3                                                         ABCD1483
20    ERR(5) = (P25R - P25) / P25R                                              ABCD1484
      IF (IDNOZ .EQ. 1)  WRITE (6,22) A28,AM28,A29,AM29                         ABCD1485
21    ICODUC = 0                                                                ABCD1486
      CALL FASTBK                                                               ABCD1487
      RETURN                                                                    ABCD1488
C                                                                               ABCD1489
C                                                                               ABCD1490
22    FORMAT (19H0DUCT NOZZLE DESIGN, 5X, 8H    A28=, E15.8,                    ABCD1491
```

```
     1 8H    AM28=, E15.8, 8H     A29=, E15.8, 8H    AM29=, E15.8)          ABCD1492
       END                                                                  ABCD1493
```

Subroutine COFAN

```
       SUBROUTINE COFAN                                                     ABCD1494
       IMPLICIT  REAL*8  (A-H,O-Z)                                          ABCD1495
       LOGICAL  SI, FXM2CP                                                  ABCD1496
       COMMON /COMALL/  COM(1062)                                           ABCD1497
       COMMON /COMDAT/  COMD(5423)                                          ABCD1498
       DIMENSION  WORD(2), CNXF(15), PRXF(15,15), WACXF(15,15),             ABCD1499
      1 ETAXF(15,15), NPTF(15), DUMD1(15)                                   ABCD1500
       DIMENSION  Q(9), WLH(2,2), AWORD(2)                                  ABCD1501
       EQUIVALENCE (WORD(1), COM(1)), (IDES, COM(3)), (JDES, COM(4)),       ABCD1502
      1 (MODE, COM(6)), (INIT, COM(7)), (MAPEDG, COM(22)), (TOLALL,         ABCD1503
      2 COM(23)), (H22, COM(34)), (T2DS, COM(91)), (T2, COM(92)), (P2,      ABCD1504
      3 COM(93)), (H2, COM(94)), (S2, COM(95)), (S22, COM(96)), (T22DS,     ABCD1505
      4 COM(97)), (T4GU, COM(100)), (T4DS, COM(101)), (PRFCF, COM(121)),    ABCD1506
      5 (ETAFCF, COM(122)), (WAFCF, COM(123)), (PCNFDS, COM(124)),          ABCD1507
      6 (PRFDS, COM(125)), (ETAFDS, COM(126)), (WAFDS, COM(127)),           ABCD1508
      7 (PCNCGU, COM(128)), (PFF, COM(131)), (ETAF, COM(132)),              ABCD1509
      8 (ZCDS, COM(133)), (CNF, COM(134)), (WAFC, COM(135)),                ABCD1510
      9 (ZF, COM(136)), (PCNF, COM(137)), (PCBLF, COM(138)),                ABCD1511
      1 (ZI, COM(139)), (PCNI, COM(140)), (ZIDS, COM(146)),                 ABCD1512
      2 (PCNIDS, COM(147)), (PCNIGU, COM(148)), (T4, COM(156)),             ABCD1513
      3 (WFBDS, COM(185)), (WFB, COM(192)), (T21, COM(263)),                ABCD1514
      4 (T21DS, COM(267)), (T22, COM(268)), (ZC, COM(300)),                 ABCD1515
      5 (PCNC, COM(301)), (PCNCDS, COM(303)), (BLF, COM(316)),              ABCD1516
      6 (WAF, COM(319)), (P22, COM(375)), (U22, COM(376)),                  ABCD1517
      7 (VFAN, COM(395)), (DUMD1(1), COM(405)), (WAFP, COM(420)),           ABCD1518
      8 (JTRAN, COM(1050)), (SI, COM(1055)), (FXM2CP, COM(1059))            ABCD1519
       EQUIVALENCE  (CNXF(1), COMD(1)), (PRXF(1,1), COMD(16)),              ABCD1520
      1 (WACXF(1,1), COMD(241)), (ETAXF(1,1), COMD(466)),                   ABCD1521
      2 (NCNF, COMD(5296)), (NPTF(1), COMD(5297))                           ABCD1522
       DATA  AWORD, WLH /4H COF, 4HAN  , 4H (LO, 4H)   , 4H (HI, 4H)   /    ABCD1523
       WORD(1) = AWORD(1)                                                   ABCD1524
       WORD(2) = AWORD(2)                                                   ABCD1525
       IF (SI)  GO TO 100                                                   ABCD1526
       TSTD = 518.668D0                                                     ABCD1527
       PSTD = 1.0D0                                                         ABCD1528
       RA = .0252D0                                                         ABCD1529
       AJ = 2.719D0                                                         ABCD1530
       GO TO 101                                                            ABCD1531
100    TSTD = 288.149D0                                                     ABCD1532
       PSTD = 101325.0D0                                                    ABCD1533
       RA = 286.9D0                                                         ABCD1534
       AJ = 1.0D0                                                           ABCD1535
101    THETA = DSQRT(T2 / TSTD)                                             ABCD1536
       DELTA = P2 / PSTD                                                    ABCD1537
       IF (IDES .NE. 1)  GO TO 1                                            ABCD1538
       WAFDS = WAFC * DELTA / THETA                                         ABCD1539
1      CNF = PCNF / (100.0D0 * THETA)                                       ABCD1540
       IF (ZF .LT. 0.0D0)  ZF = 0.0D0                                       ABCD1541
       IF (ZF .GT. 1.0D0)  ZF = 1.0D0                                       ABCD1542
```

```
      CNFS = CNF                                                        ABCD1543
      CALL SEARCH (ZF,CNF,PRF,WAFC,ETAF,CNXF(1),NCNF,PRXF(1,1),         ABCD1544
     1 WACXF(1,1),ETAXF(1,1),NPTF(1),15,15,IGO)                         ABCD1545
      IF ((CNF - CNFS) .GT. .5D0 * TOLALL * CNF)  MAPEDG = 1            ABCD1546
      IF (IGO .EQ. 1 .OR. IGO .EQ. 2)  WRITE (8,12)  CNFS, WLH(1,IGO),  ABCD1547
     1 WLH(2,IGO)                                                       ABCD1548
      WAF = WAFC * DELTA / THETA                                        ABCD1549
      IF (IDES .NE. 1)  GO TO 2                                         ABCD1550
      PRFCF = (PRFDS - 1.0D0) / (PRF - 1.0D0)                           ABCD1551
      ETAFCF = ETAFDS / ETAF                                            ABCD1552
      WAFCF = WAFDS / WAF                                               ABCD1553
      WRITE (6,13) PRFCF,ETAFCF,WAFCF,T2DS                              ABCD1554
2     PRF = PRFCF * (PRF - 1.0D0) + 1.0D0                               ABCD1555
      ETAF = ETAFCF * ETAF                                              ABCD1556
      WAF = WAFCF * WAF                                                 ABCD1557
      WAFP = WAF                                                        ABCD1558
      WAFC = WAFC * WAFCF                                               ABCD1559
      PCNF = 100.0D0 * THETA * CNF                                      ABCD1560
      DUMD1(1) = PCNF                                                   ABCD1561
      CALL THCOMP (PRF,ETAF,T2,H2,S2,P2,T22,H22,S22,P22)                ABCD1562
      IF (VFAN .EQ. 0.0D0)  GO TO 21                                    ABCD1563
      Q(2) = 0.0D0                                                      ABCD1564
      Q(3) = 0.0D0                                                      ABCD1565
      H22P = H22                                                        ABCD1566
      P22DOT = DERIV(4,P22)                                             ABCD1567
18    CALL THERMO (P22,H22,T22,S22,XX2,0,0.0D0,0)                       ABCD1568
      WAF = WAFP - P22DOT * VFAN / T22 / 1.4D0 / RA                     ABCD1569
      U22 = H22 - AJ * RA * T22                                         ABCD1570
      U22DOT = DERIV(5,U22)                                             ABCD1571
      H22X = (WAFP * H22P - (WAFP - WAF) * U22 - U22DOT * P22 * VFAN /  ABCD1572
     1 T22 / RA) / WAF                                                  ABCD1573
      ERRW = (H22 - H22X) / H22                                         ABCD1574
      DIR = DSQRT(DABS(H22 / H22X))                                     ABCD1575
      CALL AFQUIR (Q(1),T22,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T22T,IGO) ABCD1576
      GO TO (19,21,20), IGO                                             ABCD1577
19    T22 = T22T                                                        ABCD1578
      GO TO 18                                                          ABCD1579
20    CALL ERROR                                                        ABCD1580
21    IF (PCBLF .GT. 0.0D0)  BLF = PCBLF * WAF                          ABCD1581
      IF (JDES .EQ. 1)  GO TO 9                                         ABCD1582
      JDES = 1                                                          ABCD1583
      IF (INIT .EQ. 1)  GO TO 8                                         ABCD1584
      IF (IDES .EQ. 1)  GO TO 6                                         ABCD1585
      IF (JTRAN .EQ. 1)  GO TO 8                                        ABCD1586
      IF (MODE .NE. 2)  GO TO 3                                         ABCD1587
      T4 = GUESS(3,Y1,Y2,PCNF,PCNFDS,WFB,WFBDS,Y7,Y8,T4DS)              ABCD1588
      PCNI = GUESS(8,T4,T4DS,Y3,Y4,Y5,Y6,T22,T22DS,PCNIDS)              ABCD1589
      PCNC = GUESS(4,Y1,Y2,PCNI,PCNIDS,WFB,WFBDS,Y7,Y8,PCNCDS)          ABCD1590
      GO TO 7                                                           ABCD1591
3     IF (MODE .EQ. 1)  GO TO 5                                         ABCD1592
      IF (MODE .EQ. 0)  GO TO 4                                         ABCD1593
      T4 = GUESS(7,Y1,Y2,PCNF,PCNFDS,Y5,Y6,T2,T2DS,T4DS)                ABCD1594
4     PCNC = GUESS(5,T4,T4DS,Y3,Y4,Y5,Y6,T22,T22DS,PCNCDS)              ABCD1595
      IF (FXM2CP)  PCNC = PCNCDS * .99D0 * THETA                        ABCD1596
```

```
      PCNCG1 = PCNC                                                          ABCD1597
      PCNCG2 = PCNCDS                                                        ABCD1598
      PCNI = GUESS(9,Y1,Y2,PCNCG1,PCNCG2,Y5,Y6,T22,T22DS,PCNIDS)             ABCD1599
      GO TO 7                                                                ABCD1600
5     T4 = GUESS(6,Y1,Y2,PCNC,PCNCDS,Y5,Y6,T22,T22DS,T4DS)                   ABCD1601
      PCNI = GUESS(8,T4,T4DS,Y3,Y4,Y5,Y6,T22,T22DS,PCNIDS)                   ABCD1602
      GO TO 7                                                                ABCD1603
6     T4 = T4DS                                                              ABCD1604
      WFB = WFBDS                                                            ABCD1605
      T21DS = T21                                                            ABCD1606
7     ZC = ZCDS                                                              ABCD1607
      ZI = ZIDS                                                              ABCD1608
      PCNIGU = PCNI                                                          ABCD1609
      PCNCGU = PCNC                                                          ABCD1610
      T4GU = T4                                                              ABCD1611
8     INIT = 0                                                               ABCD1612
9     IF (MODE .NE. 3)  GO TO 10                                             ABCD1613
      IF (DABS(CNF - CNFS) .LE. 1.0D0 * TOLALL * CNFS)  GO TO 11             ABCD1614
      WRITE (8,14) CNFS,CNF                                                  ABCD1615
      CALL ERROR                                                             ABCD1616
10    PCNF = 100.0D0 * THETA * CNF                                           ABCD1617
11    CALL COINTC                                                            ABCD1618
      RETURN                                                                 ABCD1619
C                                                                            ABCD1620
C                                                                            ABCD1621
12    FORMAT (19H0* * *  CNF OFF MAP,F10.4,2X,A4,A2,11H* * *$$$$$$)         ABCD1622
13    FORMAT(11H0FAN DESIGN,13X,8H  PRFCF=,E15.8,8H ETAFCF=,E15.8,           ABCD1623
     1 8H  WAFCF=,E15.8,8H   T2DS=,E15.8)                                    ABCD1624
14    FORMAT (10H0CNF WAS= ,E15.8,11H  AND NOW= ,E15.8,12H  CHECK PCNF,      ABCD1625
     1 12H INPUT$$$$$$)                                                      ABCD1626
      END                                                                    ABCD1627
```

Subroutine COHPTB

```
      SUBROUTINE COHPTB                                                      ABCD1628
      IMPLICIT  REAL*8  (A-H,O-Z)                                            ABCD1629
      LOGICAL  SI, DUMSPL, FXFN2M                                            ABCD1630
      COMMON /COMALL/  COM(1062)                                             ABCD1631
      COMMON /COMDAT/  COMD(5423)                                            ABCD1632
      DIMENSION  WORD(2), ERR(9), TFFXH(15), CNXH(15,15), DHTCXH(15,15),ABCD1633
     1 ETATXH(15,15), NPTTFH(15)                                             ABCD1634
      DIMENSION  Q(9), AWORD(2), WLO(2), WHI(2)                              ABCD1635
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (NOMAP, COM(20)), ABCD1636
     1 (TOLALL, COM(23)), (ERR(1), COM(24)), (T5, COM(102)), (H5,        ABCD1637
     2 COM(103)), (S5, COM(104)), (WG5, COM(105)), (FAR5, COM(106)),     ABCD1638
     3 (TFHPCF, COM(114)), (CNHPCF, COM(115)), (ETHPCF, COM(116)),       ABCD1639
     4 (DHHPCF, COM(117)), (TFHPDS, COM(118)), (CNHPDS, COM(119)),       ABCD1640
     5 (ETHPDS, COM(120)), (HPEXT, COM(129)), (TFFIP, COM(141)),         ABCD1641
     6 (CNIP, COM(142)), (ETATIP, COM(143)), (DHTCIP, COM(144)),         ABCD1642
     7 (DHTI, COM(145)), (H3, COM(153)), (T4, COM(156)), (H4, COM(157)),ABCD1643
     8 (S4, COM(158)), (WG4, COM(159)), (FAR4, COM(160)),                ABCD1644
     9 (T50, COM(161)), (H50, COM(162)), (S50, COM(163)),                ABCD1645
     1 (WG50, COM(164)), (FAR50, COM(165)), (CNHP, COM(171)),            ABCD1646
     2 (ETATHP, COM(172)), (DHTCHP, COM(173)), (DHTC, COM(174)),         ABCD1647
```

```
     3 (TFFHP, COM(175)), (H21, COM(264)), (PCNC, COM(301)),                ABCD1648
     4 (BLHP, COM(313)), (XNHP, COM(372)), (P4, COM(381)),                  ABCD1649
     5 (P50, COM(383)), (U50, COM(384)), (P5, COM(385)),                    ABCD1650
     6 (VHPTRB, COM(399)), (WACP, COM(422)), (XNHPDS, COM(423)),            ABCD1651
     7 (PMIHP, COM(426)), (ISPOOL, COM(1044)), (SI, COM(1055)),             ABCD1652
     8 (DUMSPL, COM(1057)), (FXFN2M, COM(1058))                             ABCD1653
      EQUIVALENCE  (TFFXH(1), COMD(3226)), (CNXH(1,1), COMD(3241)),         ABCD1654
     1 (DHTCXH(1,1), COMD(3466)), (ETATXH(1,1), COMD(3691)),                ABCD1655
     2 (NTFFSH, COMD(5376)), (NPTTFH(1), COMD(5377))                        ABCD1656
      DATA  AWORD, WLO, WHI /4HCOHP, 4HTB  , 4H (LO, 4H)    , 4H (HI,       ABCD1657
     1 4H)   /                                                              ABCD1658
      WORD(1) = AWORD(1)                                                    ABCD1659
      WORD(2) = AWORD(2)                                                    ABCD1660
      IF (SI)  GO TO 100                                                    ABCD1661
      RA = .0252D0                                                          ABCD1662
      AJ = 2.719D0                                                          ABCD1663
      CONFAC = 1.4091D-5                                                    ABCD1664
      GO TO 101                                                             ABCD1665
100   RA = 286.9D0                                                          ABCD1666
      AJ = 1.0D0                                                            ABCD1667
      CONFAC = 1.0966D-2                                                    ABCD1668
101   IF (ISPOOL .EQ. 1)  GO TO 8                                           ABCD1669
      THDE = DSQRT(T4) / PCNC                                               ABCD1670
      IF (IDES .EQ. 0)  GO TO 1                                             ABCD1671
      CNHPCF = CNHPDS * THDE                                                ABCD1672
1     CNHP = CNHPCF / THDE                                                  ABCD1673
      CNHPS = CNHP                                                          ABCD1674
      TFFHPS = TFFHP                                                        ABCD1675
      CALL SEARCH (-1.D0,TFFHP,CNHP,DHTCHP,ETATHP,TFFXH(1),NTFFSH,          ABCD1676
     1 CNXH(1,1),DHTCXH(1,1),ETATXH(1,1),NPTTFH(1),15,15,IGO)               ABCD1677
      IF (IGO .EQ. 1 .OR. IGO .EQ. 11 .OR. IGO .EQ. 21)                     ABCD1678
     1 WRITE (8,9) TFFHPS,WLO                                               ABCD1679
      IF (IGO .EQ. 2 .OR. IGO .EQ. 12 .OR. IGO .EQ. 22)                     ABCD1680
     1 WRITE (8,9) TFFHPS,WHI                                               ABCD1681
      IF (IGO .EQ. 10 .OR. IGO .EQ. 11 .OR. IGO .EQ. 12)                    ABCD1682
     1 WRITE (8,10) CNHPS,WLO                                               ABCD1683
      IF (IGO .EQ. 20 .OR. IGO .EQ. 21 .OR. IGO .EQ. 22)                    ABCD1684
     1 WRITE (8,10) CNHPS,WHI                                               ABCD1685
      IF (IGO .NE. 7)  GO TO 2                                              ABCD1686
      CALL ERROR                                                            ABCD1687
      RETURN                                                                ABCD1688
2     NOMAP = 0                                                             ABCD1689
      TFHCAL = WG4 * DSQRT(T4) / (14.696D0 * P4)                            ABCD1690
      BTUEXT = 0.706705D0 * HPEXT                                           ABCD1691
      IF (.NOT. SI)  GO TO 102                                              ABCD1692
      TFHCAL = WG4 * DSQRT(T4) / P4                                         ABCD1693
      BTUEXT = HPEXT                                                        ABCD1694
102   XNHP = XNHPDS * PCNC / 100.0D0                                        ABCD1695
      XNHDOT = DERIV(1,XNHP)                                                ABCD1696
      DHTCC = (BTUEXT + WACP * (H3 - H21) + CONFAC * PMIHP * XNHP *         ABCD1697
     1 XNHDOT) / (WG4 * T4)                                                 ABCD1698
      IF (IDES .EQ. 0)  GO TO 5                                             ABCD1699
      TFHPCF = TFHPDS / TFHCAL                                              ABCD1700
      DHHPCF = DHTCC / DHTCHP                                               ABCD1701
```

```
      ETHPCF = ETHPDS / ETATHP                                          AECD1702
      WRITE (6,11) CNHPCF,TFHPCF,ETHPCF,DHHPCF                          ABCD1703
5     TFHCAL = TFHPCF * TFHCAL                                          ABCD1704
      DHTCHP = DHHPCF * DHTCHP                                          ABCD1705
      ETATHP = ETHPCF * ETATHP                                          ABCD1706
      DHTC = DHTCC * T4                                                 ABCD1707
      ERR(1) = (TFHCAL - TFFHP) / TFHCAL                                ABCD1708
      ERR(2) = (DHTCC - DHTCHP) / DHTCC                                 ABCD1709
      CALL THTURB (DHTC,ETATHP,FAR4,H4,S4,P4,T50,H50,S50,P50)           ABCD1710
      IF (BLHP .LE. 0.0D0)  GO TO 6                                     ABCD1711
      FAR50 = FAR4 * WG4 / (WG4 + BLHP * (FAR4 + 1.0D0))                ABCD1712
      WG50 = WG4 + BLHP                                                 ABCD1713
      H50 = (BLHP * H3 + WG4 * H50) / WG50                              ABCD1714
      CALL THERMO (P50,H50,T50,S50,XX2,1,FAR50,1)                       ABCD1715
      GO TO 7                                                           ABCD1716
6     FAR50 = FAR4                                                      ABCD1717
      WG50 = WG4                                                        ABCD1718
7     IF (VHPTRB .EQ. 0.0D0)  GO TO 21                                  ABCD1719
      Q(2) = 0.0D0                                                      ABCD1720
      Q(3) = 0.0D0                                                      ABCD1721
      WG50P = WG50                                                      ABCD1722
      H50P = H50                                                        ABCD1723
      P50DOT = DERIV(12,P50)                                            ABCD1724
18    CALL THERMO (P50,H50,T50,S50,XX2,1,FAR50,0)                       ABCD1725
      WG50 = WG50P - P50DOT * VHPTRB / T50 / 1.4D0 / RA                 ABCD1726
      U50 = H50 - RA * AJ * T50                                         ABCD1727
      U50DOT = DERIV(13,U50)                                            ABCD1728
      H50X = (WG50P * H50P - (WG50P - WG50) * U50 - U50DOT * P50 *      ABCD1729
     1 VHPTRB / T50 / RA) / WG50                                        ABCD1730
      ERRW = (H50 - H50X) / H50                                         ABCD1731
      DIR = DSQRT(DABS(H50 / H50X))                                     ABCD1732
      CALL AFQUIR (Q(1),T50,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T50T,IGO) ABCD1733
      GO TO (19,21,20), IGO                                             ABCD1734
19    T50 = T50T                                                        ABCD1735
      GO TO 18                                                          ABCD1736
20    CALL ERROR                                                        ABCD1737
21    IF (FXFN2M .OR. DUMSPL)  GO TO 8                                  ABCD1738
      CALL COIPTB                                                       ABCD1739
      RETURN                                                            ABCD1740
8     P5 = P50                                                          ABCD1741
      H5 = H50                                                          ABCD1742
      T5 = T50                                                          ABCD1743
      S5 = S50                                                          ABCD1744
      FAR5 = FAR50                                                      ABCD1745
      WG5 = WG50                                                        ABCD1746
C     SET MIDDLE TURBINE PARAMETERS TO ZERO, NOT USED                   ABCD1747
      TFFIP = 0.0D0                                                     ABCD1748
      CNIP = 0.0D0                                                      ABCD1749
      DHTI = 0.0D0                                                      ABCD1750
      DHTCIP = 0.0D0                                                    ABCD1751
      ETATIP = 0.0D0                                                    ABCD1752
      CALL COLPTB                                                       ABCD1753
      RETURN                                                            ABCD1754
C                                                                       ABCD1755
```

```
C                                                                       ABCD1756
C                                                                       ABCD1757
9     FORMAT (19H0*****TFFHP OFF MAP,F10.4,2X,A4,A2,11H*****$$$$$$)    ABCD1758
10    FORMAT (19H0***** CNHP OFF MAP,F10.4,2X,A4,A2,11H*****$$$$$$)    ABCD1759
11    FORMAT(20H0H.P. TURBINE DESIGN,5X,7HCNHPCF=,E15.8,8H TFHPCF=,     ABCD1760
     1 E15.8,8H ETHPCF=,E15.8,8H DHHPCF=,E15.8)                         ABCD1761
      END                                                               ABCD1762
```

Subroutine COINLT

```
      SUBROUTINE COINLT                                                 ABCD1763
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD1764
      LOGICAL  SI                                                       ABCD1765
      COMMON /COMALL/  COM(1062)                                        ABCD1766
      DIMENSION  WORD(2)                                                ABCD1767
      DIMENSION  AWORD(2)                                               ABCD1768
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MODE, COM(6)),   ABCD1769
     1 (INIT, COM(7)), (IAMTP, COM(9)), (TOLALL, COM(23)), (P1,         ABCD1770
     2 COM(33)), (T2DS, COM(91)), (T2, COM(92)), (P2, COM(93)), (H2,    ABCD1771
     3 COM(94)), (S2, COM(95)), (T4DS, COM(101)), (PCNFDS, COM(124)),   ABCD1772
     4 (ZF, COM(136)), (PCNF, COM(137)), (T1, COM(149)), (H1, COM(150)),ABCD1773
     5 (S1, COM(151)), (T4, COM(156)), (WFBDS, COM(185)), (ZFDS,        ABCD1774
     6 COM(186)), (ETAR, COM(187)), (WFB, COM(192)), (CS, COM(194)),    ABCD1775
     7 (AM, COM(195)), (ALTP, COM(196)), (PCNFGU, COM(199)),            ABCD1776
     8 (PCNC, COM(301)), (PCNCDS, COM(303)), (DELT1, COM(429)),         ABCD1777
     9 (SI, COM(1055))                                                  ABCD1778
      DATA  AWORD /4HCOIN, 4HLT  /                                      ABCD1779
      WORD(1) = AWORD(1)                                                ABCD1780
      WORD(2) = AWORD(2)                                                ABCD1781
      IF (SI)  GO TO 10                                                 ABCD1782
      AJ = 778.26D0                                                     ABCD1783
      G = 32.174049D0                                                   ABCD1784
      REF59 = 2.0855531D07                                              ABCD1785
      R = 1.986375D0                                                    ABCD1786
      TSTD = 518.668D0                                                  ABCD1787
      GO TO 11                                                          ABCD1788
10    AJ = 1.0D0                                                        ABCD1789
      G = 1.0D0                                                         ABCD1790
      REF59 = 6.3567658D06                                              ABCD1791
      R = 8314.34D0                                                     ABCD1792
      TSTD = 288.149D0                                                  ABCD1793
11    ALT = ALTP * REF59 / (REF59 - ALTP)                               ABCD1794
      GAJ2 = 2.0D0 * G * AJ                                             ABCD1795
      CALL ATMOS (ALT,T1STD,XX1,XX2,XX3,DELTA,CS,XX4,IIER)              ABCD1796
      P1 = DELTA                                                        ABCD1797
      IF (SI)  P1 = 101325.0D0 * DELTA                                  ABCD1798
      T1 = T1STD                                                        ABCD1799
      IF (IAMTP .EQ. 2)  T1 = T1STD + DELT1                             ABCD1800
      IF (IAMTP .EQ. 5)  CALL RAM2 (AM,ETAR)                            ABCD1801
      IF (IAMTP .NE. 1 .AND. IAMTP .NE. 5)  CALL RAM (AM,ETAR)          ABCD1802
      FAR = 0.0D0                                                       ABCD1803
      CALL PROCOM (FAR,T1,CS,XX2,XX3,R1,PHI1,H1)                        ABCD1804
      S1 = PHI1 - R1 * DLOG(DELTA)                                      ABCD1805
      H2 = H1 + (AM * CS) ** 2 / GAJ2                                   ABCD1806
```

```
      P2T = 1.0D0                                                          ABCD1807
      IF (SI)  P2T = 101325.0D0                                            ABCD1808
      DO 1  I = 1,50                                                       ABCD1809
      CALL THERMO (P2T,H2,T2T,S2T,AW,0,0.0D0,1)                            ABCD1810
      IF (DABS(S2T - S1) .LE. .1D0 * TOLALL * S1)  GO TO 2                 ABCD1811
1     P2T = P1 * DEXP((AW / R) * ((S2T - S1) + (R / AW) *                  ABCD1812
     1 DLOG(P2T / P1)))                                                    ABCD1813
      CALL ERROR                                                           ABCD1814
      RETURN                                                               ABCD1815
2     IF (IAMTP .EQ. 3 .OR. IAMTP .EQ. 4)  ETAR = P2 / P2T                 ABCD1816
      P2 = ETAR * P2T                                                      ABCD1817
      ITHER = 1                                                            ABCD1818
      IF (IAMTP .EQ. 4)  ITHER = 0                                         ABCD1819
      CALL THERMO (P2,H2,T2,S2,XX5,0,0.0D0,ITHER)                          ABCD1820
      IF (INIT .EQ. 1)  RETURN                                             ABCD1821
      IF (IDES .EQ. 1)  GO TO 3                                            ABCD1822
      IF (MODE .EQ. 3)  GO TO 4                                            ABCD1823
      PCNF = GUESS(MODE,T4,T4DS,PCNC,PCNCDS,WFB,WFBDS,T2,T2DS,PCNFDS)      ABCD1824
      PCNFGU = PCNF                                                        ABCD1825
      GO TO 4                                                              ABCD1826
3     PCNF = PCNFDS * DSQRT(T2 / TSID)                                     ABCD1827
      PCNFGU = PCNF                                                        ABCD1828
      T2DS = T2                                                            ABCD1829
4     ZF = ZFDS                                                            ABCD1830
      RETURN                                                               ABCD1831
      END                                                                  ABCD1832
```

Subroutine COINTC

```
      SUBROUTINE COINTC                                                    ABCD1833
      IMPLICIT  REAL*8  (A-H,O-Z)                                          ABCD1834
      LOGICAL  SI, DUMSPL, FXFN2M, AFTFAN, FAN                             ABCD1835
      COMMON /COMALL/  COM(1062)                                           ABCD1836
      COMMON /COMDAT/  COMD(5423)                                          ABCD1837
      DIMENSION  WORD(2), CNXIN(15), PRXIN(15,15), WACXIN(15,15),          ABCD1838
     1 ETAXIN(15,15), CNXXI(15), PRXXI(15,15), WACXXI(15,15),              ABCD1839
     2 ETAXXI(15,15), NPTI(15), NPTXI(15)                                  ABCD1840
      DIMENSION  Q(9), WLH(2,2), AWORD(2)                                  ABCD1841
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MAPFDS, COM(22)),ABCD1842
     1 (TOLALL, COM(23)), (H22, COM(34)), (T2, COM(92)), (P2, COM(93)),    ABCD1843
     2 (H2, COM(94)), (S2, COM(95)), (S22, COM(96)), (T2DS, COM(97)),      ABCD1844
     3 (CNF, COM(134)), (ZI, COM(139)), (PCNI, COM(140)), (PCNIDS,         ABCD1845
     4 COM(147)), (PRICF, COM(176)), (ETAICF, COM(177)), (WAICF,           ABCD1846
     5 COM(178)), (PFIDS, COM(180)), (ETAIDS, COM(181)), (WAIDS,           ABCD1847
     6 COM(182)), (WAICDS, COM(183)), (ETAI, COM(188)), (PRI, COM(189)),ABCD1848
     7 (WAC, COM(191)), (T21, COM(263)), (H21, COM(264)),                  ABCD1849
     8 (S21, COM(265)), (WA21, COM(266)), (T22, COM(268)),                 ABCD1850
     9 (WA22, COM(269)), (WA32, COM(271)), (PCBLI, COM(304)),              ABCD1851
     1 (PCBLID, COM(305)), (CNI, COM(309)), (WACI, COM(310)),              ABCD1852
     2 (WAI, COM(311)), (BLI, COM(312)), (BLF, COM(316)),                  ABCD1853
     3 (WAF, COM(319)), (P22, COM(375)), (P21, COM(377)),                  ABCD1854
     4 (U21, COM(378)), (VINTC, COM(396)), (WAIP, COM(521)),               ABCD1855
     5 (ISPOOL, COM(1044)), (SI, COM(1055)), (DUMSPL, COM(1057)),          ABCD1856
     6 (FXFN2M, COM(1058)), (AFTFAN, COM(1060)), (FAN, COM(1061))          ABCD1857
```

```
      EQUIVALENCE  (CNXIN(1), COMD(691)), (PRXIN(1,1), COMD(706)),      ABCD1858
     1 (WACXIN(1,1), COMD(931)), (ETAXIN(1,1), COMD(1156)),             ABCD1859
     2 (CNXXI(1), COMD(1381)), (PRXXI(1,1), COMD(1396)),                ABCD1860
     3 (WACXXI(1,1), COMD(1621)), (ETAXXI(1,1), COMD(1846)),            ABCD1861
     4 (NCNI, COMD(5312)), (NPTI(1), COMD(5313)), (NCNXI, COMD(5328)),  ABCD1862
     5 (NPTXI(1), COMD(5329))                                           ABCD1863
      DATA  AWORD, WLH /4HCOIN, 4HTC  , 4H (LO, 4H)   , 4H (HI, 4H)   / ABCD1864
      WORD(1) = AWORD(1)                                                ABCD1865
      WORD(2) = AWORD(2)                                                ABCD1866
      IF (SI)  GO TO 100                                                ABCD1867
      TSTD = 518.668D0                                                  ABCD1868
      PSTD = 1.0D0                                                      ABCD1869
      RA = .0252D0                                                      ABCD1870
      AJ = 2.719D0                                                      ABCD1871
      GO TO 101                                                         ABCD1872
100   TSTD = 288.149D0                                                  ABCD1873
      PSTD = 101325.0D0                                                 ABCD1874
      RA = 286.9D0                                                      ABCD1875
      AJ = 1.0D0                                                        ABCD1876
101   IF (.NOT. AFTFAN)  GO TO 1                                        ABCD1877
      T22S = T22                                                        ABCD1878
      H22S = H22                                                        ABCD1879
      S22S = S22                                                        ABCD1880
      P22S = P22                                                        ABCD1881
      T22 = T2                                                          ABCD1882
      H22 = H2                                                          ABCD1883
      S22 = S2                                                          ABCD1884
      P22 = P2                                                          ABCD1885
1     THETA = DSQRT(T22 / TSTD)                                         ABCD1886
      DELTA = P22 / PSTD                                                ABCD1887
      IF (.NOT. FAN)  WAI = WAF - PLF                                   ABCD1888
      IF (IDES .NE. 1)  GO TO 2                                         ABCD1889
      PCNI = PCNIDS * THETA                                             ABCD1890
      PRI = PRIDS                                                       ABCD1891
      PCBLI = PCBLID                                                    ABCD1892
      IF (FAN)  GO TO 102                                               ABCD1893
      WAICDS = WAI * THETA / DELTA                                      ABCD1894
      DUMSPL = .TRUE.                                                   ABCD1895
102   WACI = WAICDS                                                     ABCD1896
      WAIDS = WACI * DELTA / THETA                                      ABCD1897
      ETAI = ETAIDS                                                     ABCD1898
2     IF (.NOT. FXFN2M)  GO TO 3                                        ABCD1899
C     FAN AND MIDDLE SPOOL ROTATE AT SAME SPEED                         ABCD1900
      SPDFAN = CNF * DSQRT(T2 / TSTD)                                   ABCD1901
      CNI = SPDFAN / THETA                                              ABCD1902
      PCNI = 100.0D0 * THETA * CNI                                      ABCD1903
      IF (IDES .EQ. 1)  PCNIDS = PCNI / THETA                           ABCD1904
3     CNI = PCNI / (100.0D0 * THETA)                                    ABCD1905
      ZI = DMAX1(ZI,0.0D0)                                              ABCD1906
      ZI = DMIN1(ZI,1.0D0)                                              ABCD1907
      CNIS = CNI                                                        ABCD1908
      IF (.NOT. DUMSPL)  GO TO 4                                        ABCD1909
      CALL INDUMY (CNI,ZI,WAICDS,IDES)                                  ABCD1910
      CALL SEARCH (ZI,CNI,PRI,WACI,ETAI,CNXXI(1),NCNXI,PRXXI(1,1),      ABCD1911
```

```
     1 WACXXI(1,1),ETAXXI(1,1),NPTXI(1),15,15,IGO)                      ABCD1912
      GO TO 5                                                           ABCD1913
4     CALL SEARCH (ZI,CNI,PRI,WACI,ETAI,CNXIN(1),NCNI,PRXIN(1,1),       ABCD1914
     1 WACXIN(1,1),ETAXIN(1,1),NPTI(1),15,15,IGO)                       ABCD1915
5     IF ((CNI - CNIS) .GT. .5D0 * TOLALL * CNI)  MAPEDG = 1            ABCD1916
      IF (IGO .EQ. 1 .OR. IGO .EQ. 2)  WRITE (8,12) CNIS,WLH(1,IGO),    ABCD1917
     1 WLH(2,IGO)                                                       ABCD1918
      IF (.NOT. FAN)  WACI = WAI * THETA / DELTA                        ABCD1919
      WAI = WACI * DELTA / THETA                                        ABCD1920
      WA22 = WAI                                                        ABCD1921
      IF (IDES .NE. 1)  GO TO 7                                         ABCD1922
      T22DS = T22                                                       ABCD1923
      IF (AFTFAN)  T22DS = T22S                                         ABCD1924
      ETAICF = ETAIDS / ETAI                                            ABCD1925
      WAICF = WAIDS / WAI                                               ABCD1926
      PRICF = (PRIDS - 1.0D0) / (PRI - 1.0D0)                           ABCD1927
      IF (.NOT. DUMSPL)  GO TO 6                                        ABCD1928
      PRICF = 1.0D0                                                     ABCD1929
      ETAICF = 1.0D0                                                    ABCD1930
      WAICF = 1.0D0                                                     ABCD1931
6     WRITE (6,13) PRICF,ETAICF,WAICF,T22DS                             ABCD1932
7     PRI = PRICF * (PRI - 1.0D0) + 1.0D0                               ABCD1933
      ETAI = ETAICF * ETAI                                              ABCD1934
      WAI = WAICF * WAI                                                 ABCD1935
      WAIP = WAI                                                        ABCD1936
      WACI = WACI * WAICF                                               ABCD1937
      WA22 = WAI                                                        ABCD1938
      CALL THCOMP (PRI,ETAI,T22,H22,S22,P22,T21,H21,S21,P21)            ABCD1939
      IF (VINTC .EQ. 0.0D0)  GO TO 21                                   ABCD1940
      Q(2) = 0.0D0                                                      ABCD1941
      Q(3) = 0.0D0                                                      ABCD1942
      H21P = H21                                                        ABCD1943
      P21DOT = DERIV(6,P21)                                             ABCD1944
18    CALL THERMO (P21,H21,T21,S21,XX2,0,0.0D0,0)                       ABCD1945
      WAI = WAIP - P21DOT * VINTC / T21 / 1.4D0 / RA                    ABCD1946
      U21 = H21 - AJ * RA * T21                                         ABCD1947
      U21DOT = DERIV(7,U21)                                             ABCD1948
      H21X = (WAIP * H21P - (WAIP - WAI) * U21 - U21DOT * P21 * VINTC / ABCD1949
     1 T21 / RA) / WAI                                                  ABCD1950
      ERRW = (H21 - H21X) / H21                                         ABCD1951
      DIR = DSQRT(DABS(H21 / H21X))                                     ABCD1952
      CALL AFQUIR (Q(1),T21,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T21T,IGO) ABCD1953
      GO TO (19,21,20), IGO                                             ABCD1954
19    T21 = T21T                                                        ABCD1955
      GO TO 18                                                          ABCD1956
20    CALL ERROR                                                        ABCD1957
21    IF (.NOT. DUMSPL)  GO TO 8                                        ABCD1958
      PRI = 1.0D0                                                       ABCD1959
      ETAI = 1.0D0                                                      ABCD1960
      T21 = T22                                                         ABCD1961
      H21 = H22                                                         ABCD1962
      S21 = S22                                                         ABCD1963
      P21 = P22                                                         ABCD1964
      IF (ISPOOL .EQ. 1)  WA21 = WAI                                    ABCD1965
```

```
8     IF (IDES .NE. 1)  GO TO 9                                          ABCD1966
      BLI = PCBLI * WAI                                                  ABCD1967
      WA21 = WA22 - BLI                                                  ABCD1968
      WA32 = BLI                                                         ABCD1969
      IF (FAN .OR. IDES .EQ. 1)  WAC = WA21                              ABCD1970
9     IF (DABS(CNI - CNIS) .LE. 1.0D0 * TOLALL * CNIS)  GO TO 10         ABCD1971
      WRITE (8,14) CNIS,CNI                                              ABCD1972
      CALL ERROR                                                         ABCD1973
      PCNI = 100.0D0 * THETA * CNI                                       ABCD1974
10    IF (.NOT. AFTFAN)  GO TO 11                                        ABCD1975
      T22 = T22S                                                         ABCD1976
      H22 = H22S                                                         ABCD1977
      S22 = S22S                                                         ABCD1978
      P22 = P22S                                                         ABCD1979
11    CALL COCOMP                                                        ABCD1980
      RETURN                                                             ABCD1981
C                                                                        ABCD1982
C                                                                        ABCD1983
C                                                                        ABCD1984
12    FORMAT (19H0* * *  CNI OFF MAP,F10.4,2X,A4,A2,11H* * *$$$$$$)     ABCD1985
13    FORMAT(20H0MIDDLE SPOOL DESIGN,4X,8H  PRICF=,E15.8,8H ETAICF=,     ABCD1986
     1 E15.8,8H  WAICF=,E15.8,8H  T22DS=,E15.8)                          ABCD1987
14    FORMAT (10H0CNI WAS= ,E15.8,11H  AND NOW= ,E15.8,12H  CHECK PCNI,  ABCD1988
     1 12H INPUT$$$$$$)                                                 ABCD1989
      END                                                                ABCD1990
```

Subroutine COIPTB

```
      SUBROUTINE COIPTB                                                  ABCD1991
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD1992
      LOGICAL  SI, FXM2CP, AFTFAN                                        ABCD1993
      COMMON /COMALL/  COM(1062)                                         ABCD1994
      COMMON /COMDAT/  COMD(5423)                                        ABCD1995
      DIMENSION  WORD(2), ERR(9), TFFXH(15), CNXH(15,15), DHTCXH(15,15),ABCD1996
     1 ETATXH(15,15), TFFXI(15), CNXI(15,15), DHTCXI(15,15),             ABCD1997
     2 ETATXI(15,15), NPTTFH(15), NPTTFI(15)                             ABCD1998
      DIMENSION  Q(9), AWORD(2), WLO(2), WHI(2)                          ABCD1999
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (NOMAP, COM(20)),  ABCD2000
     1 (TOLALL, COM(23)), (ERR(1), COM(24)), (H22, COM(34)), (H2,        ABCD2001
     2 COM(94)), (T5, COM(102)), (H5, COM(103)), (S5, COM(104)), (WG5,   ABCD2002
     3 COM(105)), (FAR5, COM(106)), (TFHPDS, COM(118)), (CNHPDS,         ABCD2003
     4 COM(119)), (ETHPDS, COM(120)), (HPEXT, COM(129)), (PCNI,          ABCD2004
     5 COM(140)), (TFFIP, COM(141)), (CNIP, COM(142)), (ETATIP,          ABCD2005
     6 COM(143)), (DHTCIP, COM(144)), (DHTI, COM(145)), (H3, COM(153)),  ABCD2006
     7 (T50, COM(161)), (H50, COM(162)), (S50, COM(163)),                ABCD2007
     8 (WG50, COM(164)), (FAR50, COM(165)), (H21, COM(264)),             ABCD2008
     9 (TFIPDS, COM(278)), (CNIPDS, COM(279)), (ETIPDS, COM(280)),       ABCD2009
     1 (TFIPCF, COM(285)), (CNIPCF, COM(286)), (ETIPCF, COM(287)),       ABCD2010
     2 (DHIPCF, COM(288)), (BLIP, COM(314)), (XNIP, COM(373)),           ABCD2011
     3 (P50, COM(383)), (P5, COM(385)), (U5, COM(386)),                  ABCD2012
     4 (VIPTRB, COM(400)), (WAIP, COM(421)), (WACP, COM(422)),           ABCD2013
     5 (XNIPDS, COM(424)), (PMIIP, COM(427)), (SI, COM(1055)),           ABCD2014
     6 (FXM2CP, COM(1059)), (AFTFAN, COM(1060))                          ABCD2015
      EQUIVALENCE  (TFFXH(1), COMD(3226)), (CNXH(1,1), COMD(3241)),      ABCD2016
```

```
     1 (DHTCXH(1,1), COMD(3466)), (ETATXH(1,1), COMD(3691)),                ABCD2017
     2 (TFFXI(1), COMD(3916)), (CNXI(1,1), COMD(3931)),                     ABCD2018
     3 (DHTCXI(1,1), COMD(4156)), (ETATXI(1,1), COMD(4381)),                ABCD2019
     4 (NTFFSH, COMD(5376)), (NPTTFH(1), COMD(5377)),                       ABCD2020
     5 (NTFFSI, COMD(5392)), (NPTTFI(1), COMD(5393))                        ABCD2021
      DATA  AWORD, WLO, WHI /4HCOIP, 4HIB  , 4H (LO, 4H)    , 4H (HI,      ABCD2022
     1 4H)   /                                                              ABCD2023
      IF (SI)  GO TO 100                                                    ABCD2024
      RA = .0252D0                                                          ABCD2025
      AJ = 2.719D0                                                          ABCD2026
      CONFAC = 1.4091D-5                                                    ABCD2027
      GO TO 101                                                             ABCD2028
100   RA = 286.9D0                                                          ABCD2029
      AJ = 1.0D0                                                            ABCD2030
      CONFAC = 1.0966D-2                                                    ABCD2031
101   H22SAV = H22                                                          ABCD2032
      IF (AFTFAN)  H22 = H2                                                 ABCD2033
      WORD(1) = AWORD(1)                                                    ABCD2034
      WORD(2) = AWORD(2)                                                    ABCD2035
      THDE = DSQRT(T50) / PCNI                                              ABCD2036
      IF (IDES .EQ. 0)  GO TO 1                                             ABCD2037
      CNIPCF = CNIPDS * THDE                                                ABCD2038
      IF (FXM2CP)  CNIPCF = CNHPDS * THDE                                   ABCD2039
1     CNIP = CNIPCF / THDE                                                  ABCD2040
      CNIPS = CNIP                                                          ABCD2041
      TFFIPS = TFFIP                                                        ABCD2042
      IF (FXM2CP)  GO TO 2                                                  ABCD2043
      CALL SEARCH (-1.0D0,TFFIP,CNIP,DHTCIP,ETATIP,TFFXI(1),NTFFSI,         ABCD2044
     1 CNXI(1,1),DHTCXI(1,1),ETATXI(1,1),NPTTFI(1),15,15,IGO)               ABCD2045
      GO TO 104                                                             ABCD2046
2     CALL SEARCH (-1.0D0,TFFIP,CNIP,DHTCIP,ETATIP,TFFXH(1),NTFFSH,         ABCD2047
     1 CNXH(1,1),DHTCXH(1,1),ETATXH(1,1),NPTTFH(1),15,15,IGO)               ABCD2048
104   IF (IGO .EQ. 1 .OR. IGO .EQ. 11 .OR. IGO .EQ. 21)                     ABCD2049
     1 WRITE (8,9) TFFIPS,WLO                                               ABCD2050
      IF (IGO .EQ. 2 .OR. IGO .EQ. 12 .OR. IGO .EQ. 22)                     ABCD2051
     1 WRITE (8,9) TFFIPS,WHI                                               ABCD2052
      IF (IGO .EQ. 30 .OR. IGO .EQ. 11 .OR. IGO .EQ. 12)                    ABCD2053
     1 WRITE (8,10) CNIPS,WLO                                               ABCD2054
      IF (IGO .EQ. 50 .OR. IGO .EQ. 21 .OR. IGO .EQ. 22)                    ABCD2055
     1 WRITE (8,10) CNIPS,WHI                                               ABCD2056
      IF (IGO .NE. 7)  GO TO 3                                              ABCD2057
      CALL ERROR                                                            ABCD2058
      RETURN                                                                ABCD2059
3     NOMAP = 0                                                             ABCD2060
      TFICAL = WG50 * DSQRT(T50) / (14.696D0 * P50)                         ABCD2061
      IF (SI)  TFICAL = WG50 * DSQRT(T50) / P50                             ABCD2062
      XNIP = XNIPDS * PCNI / 100.0D0                                        ABCD2063
      XNIDOT = DERIV(2,XNIP)                                                ABCD2064
      BTUEXT = .706705D0 * HPEXT                                            ABCD2065
      IF (SI)  BTUEXT = HPEXT                                               ABCD2066
      DHACEL = CONFAC * PMIIP * XNIP * XNIDOT                               ABCD2067
      DHTIC = (WAIP * (H21 - H22) + DHACEL) / (WG50 * T50)                  ABCD2068
      IF (FXM2CP)  DHTIC = (BTUEXT + WACP * (H3 - H21) + WAIP *             ABCD2069
     1 (H21 - H22) + DHACEL) / (WG50 * T50)                                 ABCD2070
```

```
      IF (IDES .EQ. 0)  GO TO 6                                          ABCD2071
      TFIPCF = TFIPDS / TFICAL                                           ABCD2072
      DHIPCF = DHTIC / DHTCIP                                            ABCD2073
      ETIPCF = ETIPDS / ETATIP                                           ABCD2074
      IF (.NOT. FXM2CP)  GO TO 102                                       ABCD2075
      TFIPCF = TFHPDS / TFICAL                                           ABCD2076
      ETIPCF = ETHPDS / ETATIP                                           ABCD2077
102   WRITE (6,11) CNIPCF,TFIPCF,ETIPCF,DHIPCF                           ABCD2078
6     TFICAL = TFIPCF * TFICAL                                           ABCD2079
      DHTCIP = DHIPCF * DHTCIP                                           ABCD2080
      ETATIP = ETIPCF * ETATIP                                           ABCD2081
      DHTI = DHTIC * T50                                                 ABCD2082
      N1 = 8                                                             ABCD2083
      N2 = 9                                                             ABCD2084
      IF (.NOT. FXM2CP)  GO TO 103                                       ABCD2085
      N1 = 1                                                             ABCD2086
      N2 = 2                                                             ABCD2087
103   ERR(N1) = (TFICAL - TFFIP) / TFICAL                                ABCD2088
      ERR(N2) = (DHTIC - DHTCIP) / DHTIC                                 ABCD2089
      CALL THTURB (DHTI,ETATIP,FAR50,H50,S50,P50,T5,H5,S5,P5)            ABCD2090
      IF (BLIP .LE. 0.0D0)  GO TO 7                                      ABCD2091
      FAR5 = FAR50 * WG50 / (WG50 + BLIP * (FAR50 + 1.0D0))              ABCD2092
      WG5 = WG50 + BLIP                                                  ABCD2093
      H5 = (BLIP * H3 + WG50 * H5) / WG5                                 ABCD2094
      CALL THERMO (P5,H5,T5,S5,XX2,1,FAR5,1)                             ABCD2095
      GO TO 8                                                            ABCD2096
7     FAR5 = FAR50                                                       ABCD2097
      WG5 = WG50                                                         ABCD2098
8     IF (VIPTRB .EQ. 0.0D0)  GO TO 21                                   ABCD2099
      Q(2) = 0.0D0                                                       ABCD2100
      Q(3) = 0.0D0                                                       ABCD2101
      WG5P = WG5                                                         ABCD2102
      H5P = H5                                                           ABCD2103
      P5DOT = DERIV(14,P5)                                               ABCD2104
18    CALL THERMO (P5,H5,T5,S5,XX2,1,FAR5,0)                             ABCD2105
      WG5 = WG5P - P5DOT * VIPTRB / T5 / 1.4D0 / RA                      ABCD2106
      U5 = H5 - RA * AJ * T5                                             ABCD2107
      U5DOT = DERIV(15,U5)                                               ABCD2108
      H5X = (WG5P * H5P - (WG5P - WG5) * U5 - U5DOT * P5 * VIPTRB /      ABCD2109
     1 T5 / RA) / WG5                                                    ABCD2110
      ERRW = (H5 - H5X) / H5                                             ABCD2111
      DIR = DSQRT(DABS(H5 / H5X))                                        ABCD2112
      CALL AFQUIR (Q(1),T5,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T5T,IGO)    ABCD2113
      GO TO (19,21,20), IGO                                              ABCD2114
19    T5 = T5T                                                           ABCD2115
      GO TO 18                                                           ABCD2116
20    CALL ERROR                                                         ABCD2117
21    H22 = H22SAV                                                       ABCD2118
      CALL COLPTB                                                        ABCD2119
      RETURN                                                             ABCD2120
C                                                                        ABCD2121
C                                                                        ABCD2122
C                                                                        ABCD2123
9     FORMAT (19H0*****TFFIP OFF MAP,F10.4,2X,A4,A2,11H*****$$$$$$)     ABCD2124
```

```
10    FORMAT (19H0***** CNIP OFF MAP,F10.4,2X,A4,A2,11H*****$$$$$$)      ABCD2125
11    FORMAT(20H0I.P. TURBINE DESIGN,5X,7HCNIPCF=,E15.8,8H TFIPCF=,       ABCD2126
     1 E15.8,8H ETIPCF=,E15.8,8H DHIPCF=,E15.8)                          ABCD2127
      END                                                                ABCD2128
```

Subroutine COLPTB

```
      SUBROUTINE COLPTB                                                  ABCD2129
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD2130
      LOGICAL  SI, FXFN2M, AFTFAN                                        ABCD2131
      COMMON /COMALL/  COM(1062)                                         ABCD2132
      COMMON /COMDAT/  COMD(5423)                                        ABCD2133
      DIMENSION  WORD(2), ERR(9), TFFXL(15), CNXL(15,15), DHTCXL(15,15),ABCD2134
     1 ETATXL(15,15), NPTTFL(15)                                         ABCD2135
      DIMENSION  Q(9), AWORD(2), WLO(2), WHI(2)                          ABCD2136
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (NOMAP, COM(20)),  ABCD2137
     1 (TOLALL, COM(23)), (ERR(1), COM(24)), (H22, COM(34)), (H2,        ABCD2138
     2 COM(94)), (T5, COM(102)), (H5, COM(103)), (S5, COM(104)), (WG5,   ABCD2139
     3 COM(105)), (FAR5, COM(106)), (HPEXT, COM(129)), (PCNF, COM(137)),ABCD2140
     4 (H3, COM(153)), (H21, COM(264)), (P55, COM(272)),                 ABCD2141
     5 (H55, COM(273)), (S55, COM(274)), (TFLPDS, COM(275)),             ABCD2142
     6 (CNLPDS, COM(276)), (ETLPDS, COM(277)), (TFLPCF, COM(281)),       ABCD2143
     7 (CNLPCF, COM(282)), (ETLPCF, COM(283)), (DHLPCF, COM(284)),       ABCD2144
     8 (TFFLP, COM(289)), (CNLP, COM(290)), (ETATLP, COM(291)),          ABCD2145
     9 (DHTCLP, COM(292)), (DHTF, COM(293)), (BLLP, COM(315)),           ABCD2146
     1 (WG55, COM(323)), (FAR55, COM(324)), (XNLP, COM(374)),            ABCD2147
     2 (P5, COM(385)), (P55, COM(387)), (U55, COM(388)),                 ABCD2148
     3 (VLPTRB, COM(401)), (WAFP, COM(420)), (WAIP, COM(421)),           ABCD2149
     4 (XNLPDS, COM(425)), (PMILP, COM(428)), (ISPOOL, COM(1044)),       ABCD2150
     5 (SI, COM(1055)), (FXFN2M, COM(1058)), (AFTFAN, COM(1060))         ABCD2151
      EQUIVALENCE  (TFFXL(1), COMD(4506)), (CNXL(1,1), COMD(4621)),      ABCD2152
     1 (DHTCXL(1,1), COMD(4846)), (ETATXL(1,1), COMD(5071)),             ABCD2153
     2 (NTFFSL, COMD(5408)), (NPTTFL(1), COMD(5409))                     ABCD2154
      DATA  AWORD, WLO, WHI /4HCOLP, 4HTB  , 4H (LO, 4H)   , 4H (HI,     ABCD2155
     1 4H)   /                                                           ABCD2156
      WORD(1) = AWORD(1)                                                 ABCD2157
      WORD(2) = AWORD(2)                                                 ABCD2158
      IF (SI)  GO TO 100                                                 ABCD2159
      RA = .0252D0                                                       ABCD2160
      AJ = 2.719D0                                                       ABCD2161
      CONFAC = 1.4091D-5                                                 ABCD2162
      GO TO 101                                                          ABCD2163
100   RA = 286.9D0                                                       ABCD2164
      AJ = 1.0D0                                                         ABCD2165
      CONFAC = 1.0966D-2                                                 ABCD2166
101   THDE = DSQRT(T5) / PCNF                                            ABCD2167
      IF (IDES .EQ. 0)  GO TO 1                                          ABCD2168
      CNLPCF = CNLPDS * THDE                                             ABCD2169
1     CNLP = CNLPCF / THDE                                               ABCD2170
      CNLPS = CNLP                                                       ABCD2171
      TFFLPS = TFFLP                                                     ABCD2172
      CALL SEARCH (-1.0D0,TFFLP,CNLP,DHTCLP,ETATLP,TFFXL(1),NTFFSL,      ABCD2173
     1 CNXL(1,1),DHTCXL(1,1),ETATXL(1,1),NPTTFL(1),15,15,IGO)            ABCD2174
      IF (IGO .EQ. 1 .OR. IGO .EQ. 11 .OR. IGO .EQ. 21)                  ABCD2175
```

```
     1 WRITE (8,8) TFFLPS,WLO                                                  ABCD2176
       IF (IGO .EQ. 2 .OR. IGO .EQ. 12 .OR. IGO .EQ. 22)                       ABCD2177
     1 WRITE (8,8) TFFLPS,WHI                                                  ABCD2178
       IF (IGO .EQ. 10 .OR. IGO .EQ. 11 .OR. IGO .EQ. 12)                      ABCD2179
     1 WRITE (8,9) CNLPS,WLO                                                   ABCD2180
       IF (IGO .EQ. 20 .OR. IGO .EQ. 21 .OR. IGO .EQ. 22)                      ABCD2181
     1 WRITE (8,9) CNLPS,WHI                                                   ABCD2182
       IF (IGO .NE. 7)  GO TO 2                                                ABCD2183
       CALL ERROR                                                              ABCD2184
       RETURN                                                                  ABCD2185
2      NOMAP = 0                                                              ABCD2186
       TFLCAL = WG5 * DSQRT(T5) / (14.696D0 * P5)                              ABCD2187
       IF (SI)  TFLCAL = WG5 * DSQRT(T5) / P5                                  ABCD2188
       XNLP = XNLPDS * PCNF / 100.0D0                                          ABCD2189
       XNLDOT = DERIV(3,XNLP)                                                  ABCD2190
       DHACEL = CONFAC * PMILP * XNLP * XNLDOT                                 ABCD2191
       DHTCF = (WAFP * (H22 - H2) + DHACEL) / (WG5 * T5)                       ABCD2192
       DEXT = DHTCF                                                            ABCD2193
       IF (FXFN2M .AND. .NOT. AFTFAN)  DHTCF = DEXT + WAIP * (H21 - H22)       ABCD2194
     1 / (WG5 * T5)                                                            ABCD2195
       IF (FXFN2M .AND. AFTFAN)  DHTCF = DEXT + WAIP * (H21 - H2) /            ABCD2196
     1 (WG5 * T5)                                                              ABCD2197
       IF (ISPOOL .GE. 2)  GO TO 11                                            ABCD2198
       BTUEXT = 0.706705D0 * HPEXT                                             ABCD2199
       IF (SI)  BTUEXT = HPEXT                                                 ABCD2200
       DHTCF = DEXT + BTUEXT / (WG5 * T5)                                      ABCD2201
11     IF (IDES .EQ. 0)  GO TO 5                                               ABCD2202
       TFLPCF = TFLPDS / TFLCAL                                                ABCD2203
       DHLPCF = DHTCF / DHTCLP                                                 ABCD2204
       ETLPCF = ETLPDS / ETATLP                                                ABCD2205
       WRITE (6,10) CNLPCF,TFLPCF,ETLPCF,DHLPCF                                ABCD2206
5      TFLCAL = TFLPCF * TFLCAL                                                ABCD2207
       DHTCLP = DHLPCF * DHTCLP                                                ABCD2208
       ETATLP = ETLPCF * ETATLP                                                ABCD2209
       DHTF = DHTCF * T5                                                       ABCD2210
       I1 = 3                                                                  ABCD2211
       I2 = 4                                                                  ABCD2212
       IF (ISPOOL .NE. 1)  GO TO 102                                           ABCD2213
       I1 = 1                                                                  ABCD2214
       I2 = 2                                                                  ABCD2215
102    ERR(I1) = (TFLCAL - TFFLP) / TFLCAL                                     ABCD2216
       ERR(I2) = (DHTCF - DHTCLP) / DHTCF                                      ABCD2217
       CALL THTURB (DHTF,ETATLP,FAR5,H5,S5,P5,T55,H55,S55,P55)                 ABCD2218
       IF (BLLP .LE. 0.0D0)  GO TO 6                                           ABCD2219
       FAR55 = FAR5 * WG5 / (WG5 + BLLP * (1.0D0 + FAR5))                      ABCD2220
       WG55 = WG5 + BLLP                                                       ABCD2221
       H55 = (BLLP * H3 + WG5 * H55) / WG55                                    ABCD2222
       CALL THERMO (P55,H55,T55,S55,XX2,1,FAR55,1)                             ABCD2223
       GO TO 7                                                                 ABCD2224
6      FAR55 = FAR5                                                            ABCD2225
       WG55 = WG5                                                              ABCD2226
7      IF (VLPTRB .EQ. 0.0D0)  GO TO 21                                        ABCD2227
       Q(2) = 0.0D0                                                            ABCD2228
       Q(3) = 0.0D0                                                            ABCD2229
```

```
      WG55P = WG55                                                          ABCD2230
      H55P = H55                                                            ABCD2231
      P55DOT = DERIV(16,P55)                                                ABCD2232
18    CALL THERMO (P55,H55,T55,S55,XX2,1,FAR55,0)                           ABCD2233
      WG55 = WG55P - P55DOT * VLPTRB / T55 / 1.4D0 / RA                     ABCD2234
      U55 = H55 - RA * AJ * T55                                             ABCD2235
      U55DOT = DERIV(17,U55)                                                ABCD2236
      H55X = (WG55P * H55P - (WG55P - WG55) * U55 - U55DOT * P55 *          ABCD2237
     1 VLPTRB / T55 / RA) / WG55                                            ABCD2238
      ERRW = (H55 - H55X) / H55                                             ABCD2239
      DIR = DSQRT(DABS(H55 / H55X))                                         ABCD2240
      CALL AFQUIR (Q(1),T55,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T55T,IGO)     ABCD2241
      GO TO (19,21,20), IGO                                                 ABCD2242
19    T55 = T55T                                                            ABCD2243
      GO TO 18                                                              ABCD2244
20    CALL ERROR                                                            ABCD2245
21    CALL FRTOSD                                                           ABCD2246
      RETURN                                                                ABCD2247
C                                                                           ABCD2248
C                                                                           ABCD2249
8     FORMAT (19H0*****TFFLP OFF MAP,F10.4,2X,A4,A2,11H*****$$$$$$)        ABCD2250
9     FORMAT (19H0***** CNLP OFF MAP,F10.4,2X,A4,A2,11H*****$$$$$$)        ABCD2251
10    FORMAT(20H0L.P. TURBINE DESIGN,5X,7HCNLPCF=,E15.8,8H TFLPCF=,         ABCD2252
     1 E15.8,8H ETLPCF=,E15.8,8H DHLPCF=,E15.8)                             ABCD2253
      END                                                                   ABCD2254
```

Subroutine COMIX

```
      SUBROUTINE COMIX                                                      ABCD2255
      IMPLICIT  REAL*8  (A-H,O-Z)                                           ABCD2256
      LOGICAL  SI                                                           ABCD2257
      COMMON /COMALL/  COM(1062)                                            ABCD2258
      DIMENSION  WORD(2), ERR(9), DUMD1(15)                                 ABCD2259
      DIMENSION  QQ(9), AWORD(2)                                            ABCD2260
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MODE, COM(6)),       ABCD2261
     1 (IGASMX, COM(10)), (NOMAP, COM(20)), (TOLALL, COM(23)), (ERR(1),     ABCD2262
     2 COM(24)), (AM25, COM(45)), (T25, COM(46)), (P25, COM(47)), (H25,     ABCD2263
     3 COM(48)), (S25, COM(49)), (A25, COM(98)), (V25, COM(99)), (AM55,     ABCD2264
     4 COM(107)), (V55, COM(108)), (A55, COM(109)), (PS55, COM(110)),       ABCD2265
     5 (S6, COM(111)), (PS6, COM(112)), (V6, COM(113)),                     ABCD2266
     6 (PRFDS, COM(125)), (ZF, COM(136)), (PCNF, COM(137)),                 ABCD2267
     7 (T55, COM(272)), (H55, COM(273)), (S55, COM(274)),                   ABCD2268
     8 (PRCDS, COM(297)), (WG24, COM(321)), (FAR24, COM(322)),              ABCD2269
     9 (WG55, COM(323)), (FAR55, COM(324)), (AM6, COM(327)),                ABCD2270
     1 (A6, COM(328)), (WG6, COM(330)), (T6, COM(331)), (P6, COM(332)),     ABCD2271
     2 (H6, COM(333)), (P55, COM(387)), (DUMD1(1), COM(405)),               ABCD2272
     3 (PRFNEW, COM(991)), (PRCNFW, COM(992)), (ICOMIX, COM(1047)),         ABCD2273
     4 (KKGO, COM(1048)), (SI, COM(1055))                                   ABCD2274
      DATA  AWORD /4H COM, 4HIX  /                                          ABCD2275
      WORD(1) = AWORD(1)                                                    ABCD2276
      WORD(2) = AWORD(2)                                                    ABCD2277
      IF (SI)  GO TO 100                                                    ABCD2278
      AJ = 778.26D0                                                         ABCD2279
      CAPSF = 2116.2170D0                                                   ABCD2280
```

```
      G = 32.174049D0                                                    ABCD2281
      RDEM = 1.986375D0                                                  ABCD2282
      GO TO 101                                                          ABCD2283
100   AJ = 1.0D0                                                         ABCD2284
      CAPSF = 1.0D0                                                      ABCD2285
      G = 1.0D0                                                          ABCD2286
      RDEM = 8316.41D0                                                   ABCD2287
101   GAJ2 = 2.0 * G * AJ                                                ABCD2288
      ICOMIX = 0                                                         ABCD2289
      CALL PROCOM (FAR55,T55,XX1,XX2,XX3,XX4,PHI55,XX5)                  ABCD2290
      CALL PROCOM (FAR24,T25,XX1,XX2,XX3,XX4,PHI25,XX5)                  ABCD2291
      IF (IDES .EQ. 0)  GO TO 12                                         ABCD2292
C *** CALCULATE A55 AND A25 WITH PS25=PS55                               ABCD2293
      IF (PS55 .EQ. 0.0D0)  GO TO 3                                      ABCD2294
      POP = PS55 / P55                                                   ABCD2295
      ALPOPI = DLOG(1.0D0 / POP)                                         ABCD2296
      TS55 = T55 * POP ** .286D0                                         ABCD2297
      DO 1  I = 1,50                                                     ABCD2298
      CALL PROCOM (FAR55,TS55,CS55,AK55,CP55,REX55,PHIS55,HS55)          ABCD2299
      PHIS = PHI55 - REX55 * ALPOPI                                      ABCD2300
      DELPHI = PHIS - PHIS55                                             ABCD2301
      IF (DABS(DELPHI) .LE. .1D0 * TOLALL * PHIS)  GO TO 6               ABCD2302
1     TS55 = TS55 * DEXP(4.0D0 * DELPHI)                                 ABCD2303
      ICOMIX = 1                                                         ABCD2304
2     CALL ERROR                                                         ABCD2305
      RETURN                                                             ABCD2306
3     TS55 = 0.875D0 * T55                                               ABCD2307
      DO 4  I = 1,50                                                     ABCD2308
      CALL PROCOM (FAR55,TS55,CS55,AK55,CP55,REX55,PHIS55,HS55)          ABCD2309
      V55 = AM55 * CS55                                                  ABCD2310
      HSCAL = H55 - V55 ** 2 / GAJ2                                      ABCD2311
      DELHS = HSCAL - HS55                                               ABCD2312
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 5               ABCD2313
4     TS55 = TS55 + DELHS / CP55                                         ABCD2314
      ICOMIX = 2                                                         ABCD2315
      GO TO 2                                                            ABCD2316
5     PS55 = P55 / DEXP((PHI55 - PHIS55) / REX55)                        ABCD2317
      IF (PS55 .GT. P25 .AND. IDES .EQ. 1 .AND. IGASMX .GT. 0)  GO TO 45 ABCD2318
6     IF (H55 .GT. HS55)  GO TO 7                                        ABCD2319
      WRITE (8,46) P55,PS55,T55,TS55,H55,HS55                            ABCD2320
      ICOMIX = 3                                                         ABCD2321
      CALL ERROR                                                         ABCD2322
7     V55 = DSQRT(GAJ2 * (H55 - HS55))                                   ABCD2323
      RHO = CAPSF * PS55 / (AJ * REX55 * TS55)                           ABCD2324
      A55 = WG55 / (RHO * V55)                                           ABCD2325
      AM55 = V55 / CS55                                                  ABCD2326
      IF (IGASMX .GT. 0)  GO TO 8                                        ABCD2327
      WRITE (6,47) A55,AM55                                              ABCD2328
      IF (IGASMX)  35,41,8                                               ABCD2329
8     PS25 = PS55                                                        ABCD2330
      POP = PS25 / P25                                                   ABCD2331
      ALPOPI = DLOG(1.0D0 / POP)                                         ABCD2332
      TS25 = T25 * POP ** .286D0                                         ABCD2333
      DO 9  I = 1,50                                                     ABCD2334
```

```
      CALL PROCOM (FAR24,TS25,CS25,AK25,CP25,REX25,PHIS25,HS25)          ABCD2335
      PHIS = PHI25 - REX25 * ALPOPI                                      ABCD2336
      DELPHI = PHIS - PHIS25                                             ABCD2337
      IF (DABS(DELPHI) .LE. .1D0 * TOLALL * PHIS)  GO TO 10              ABCD2338
9     TS25 = TS25 * DEXP(4.0D0 * DELPHI)                                 ABCD2339
      ICOMIX = 4                                                         ABCD2340
      GO TO 2                                                            ABCD2341
10    IF (H25 .GT. HS25)  GO TO 11                                       ABCD2342
      WRITE (8,48) P25,PS25,T25,TS25,H25,HS25                            ABCD2343
      ICOMIX = 5                                                         ABCD2344
      CALL ERROR                                                         ABCD2345
11    V25 = DSQRT(GAJ2 * (H25 - HS25))                                   ABCD2346
      RHO = CAPSF * PS25 / (AJ * REX25 * TS25)                           ABCD2347
      A25 = WG24 / (RHO * V25)                                           ABCD2348
      AM25 = V25 / CS25                                                  ABCD2349
      WRITE (6,49) A55,AM55,A25,AM25                                     ABCD2350
      GO TO 27                                                           ABCD2351
C *** CALCULATE PS55 AND PS25                                            ABCD2352
12    WQA = WG55 / A55                                                   ABCD2353
      C1 = P55 * DSQRT(G / (T55 * AJ)) * CAPSF                           ABCD2354
      MCON = 0                                                           ABCD2355
      QQ(2) = 0.0D0                                                      ABCD2356
      QQ(3) = 0.0D0                                                      ABCD2357
      AM55 = 0.50D0                                                      ABCD2358
      TS55 = 0.875D0 * T55                                               ABCD2359
13    DO 14  I = 1,50                                                    ABCD2360
      CALL PROCOM (FAR55,TS55,CS55,AK55,CP55,REX55,PHIS55,HS55)          ABCD2361
      V55 = AM55 * CS55                                                  ABCD2362
      HSCAL = H55 - V55 ** 2 / GAJ2                                      ABCD2363
      DELHS = HSCAL - HS55                                               ABCD2364
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 15              ABCD2365
14    TS55 = TS55 + DELHS / CP55                                         ABCD2366
      ICOMIX = 6                                                         ABCD2367
      GO TO 2                                                            ABCD2368
15    WQAT = C1 * DSQRT(AK55 / REX55) * AM55 / (1.0D0 + (AK55 - 1.0D0) *ABCD2369
     1 AM55 ** 2 / 2.0D0) ** ((AK55 + 1.0D0) / (2.0D0 * (AK55 - 1.0D0)))ABCD2370
      AMX = AM55                                                         ABCD2371
      IGOGO = 0                                                          ABCD2372
16    DIR = WQA / WQAT                                                   ABCD2373
      EW = (WQA - WQAT) / WQA                                            ABCD2374
      CALL AFQUIR (QQ(1),AMX,EW,0.0D0,30.0D0,.5D0*TOLALL,DIR,AMXT,ICON)  ABCD2375
      ICOMIX = 7                                                         ABCD2376
      GO TO (17,22,2), ICON                                              ABCD2377
17    IF (AMXT .LE. 1.0D0)  GO TO 20                                     ABCD2378
      AMXT = 0.7D0                                                       ABCD2379
      MCON = MCON + 1                                                    ABCD2380
      IF (MCON .LE. 1)  GO TO 20                                         ABCD2381
      IF (MODE .EQ. 3)  GO TO 19                                         ABCD2382
      PCNF = DUMD1(1)                                                    ABCD2383
      WRITE (8,50) PCNF,AMX,P55,PS55,P25,PS25                            ABCD2384
      PCNF = 1.01D0 * PCNF                                               ABCD2385
      DUMD1(1) = PCNF                                                    ABCD2386
18    NOMAP = 7                                                          ABCD2387
      ICOMIX = 0                                                         ABCD2388
```

```
      RETURN                                                                ABCD2389
19    WRITE (8,51) ZF,AMX,P55,PS55,P25,PS25                                 ABCD2390
      ZF = 0.99D0 * ZF                                                      ABCD2391
      GO TO 18                                                              ABCD2392
20    IF (IGOGO .EQ. 1)  GO TO 21                                           ABCD2393
      AM55 = AMXT                                                           ABCD2394
      GO TO 13                                                              ABCD2395
21    AM25 = AMXT                                                           ABCD2396
      GO TO 23                                                              ABCD2397
22    IF (IGOGO .EQ. 1)  GO TO 26                                           ABCD2398
      PS55 = P55 / DEXP((PHI55 - PHIS55) / REX55)                           ABCD2399
      IF (IGASMX)  35,41,103                                                ABCD2400
103   WQA = WG24 / A25                                                      ABCD2401
      C1 = P25 * DSQRT(G / (T25 * AJ)) * CAPSP                              ABCD2402
      MCON = 0                                                              ABCD2403
      QQ(2) = 0.0D0                                                         ABCD2404
      QQ(3) = 0.0D0                                                         ABCD2405
      AM25 = 0.25D0                                                         ABCD2406
      TS25 = 0.875D0 * T25                                                  ABCD2407
23    DO 24  I = 1,50                                                       ABCD2408
      CALL PROCOM (FAR24,TS25,CS25,AK25,CP25,REX25,PHIS25,HS25)             ABCD2409
      V25 = AM25 * CS25                                                     ABCD2410
      HSCAL = H25 - V25 ** 2 / GAJ2                                         ABCD2411
      DELHS = HSCAL - HS25                                                  ABCD2412
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 25                 ABCD2413
24    TS25 = TS25 + DELHS / CP25                                            ABCD2414
      ICOMIX = 8                                                            ABCD2415
      GO TO 2                                                               ABCD2416
25    WQAT = C1 * DSQRT(AK25 / REX25) * AM25 / (1.0D0 + (AK25 - 1.0D0) *ABCD2417
     1 AM25 ** 2 / 2.0D0) ** ((AK25 + 1.0D0) / (2.0D0 * (AK25 - 1.0D0)))ABCD2418
      AMX = AM25                                                            ABCD2419
      IGOGO = 1                                                             ABCD2420
      GO TO 16                                                              ABCD2421
26    PS25 = P25 / DEXP((PHI25 - PHIS25) / REX25)                           ABCD2422
27    WG6 = WG24 + WG55                                                     ABCD2423
      ERR(5) = (PS25 - PS55) / PS25                                         ABCD2424
      WF55 = FAR55 * WG55 / (FAR55 + 1.0D0)                                 ABCD2425
      WA55 = WG55 / (FAR55 + 1.0D0)                                         ABCD2426
      WF24 = FAR24 * WG24 / (FAR24 + 1.0D0)                                 ABCD2427
      WA24 = WG24 / (FAR24 + 1.0D0)                                         ABCD2428
      FAR6 = (WF55 + WF24) / (WA55 + WA24)                                  ABCD2429
      H6 = (WG24 * H25 + WG55 * H55) / WG6                                  ABCD2430
      CALL THERMO (1.0D0,H6,T6,PHI6,AMX,1,FAR6,1)                           ABCD2431
      C1 = PS55 * A55 * (1.0D0 + AK55 * AM55 ** 2) + PS25 * A25 *           ABCD2432
     1 (1.0D0 + AK25 * AM25 ** 2)                                           ABCD2433
      TS6 = 0.833D0 * T6                                                    ABCD2434
      DO 32  I = 1,50                                                       ABCD2435
      CALL PROCOM (FAR6,TS6,CS6,AK6,CP6,REX6,PHIS6,HS6)                     ABCD2436
      C2 = WG6 * DSQRT(AJ * REX6 * T6 / (AK6 * G))                          ABCD2437
      C3 = C2 / (CAPSP * C1)                                                ABCD2438
      C4 = (AK6 - 1.0D0) / 2.0D0 - (C3 * AK6) ** 2                          ABCD2439
      C5 = 1.0D0 - 2.0D0 * AK6 * C3 ** 2                                    ABCD2440
      C6 = C5 ** 2 + 4.0D0 * C4 * C3 ** 2                                   ABCD2441
      ICOMIX = 9                                                            ABCD2442
```

```
      IF (C6)  28,29,30                                                  ABCD2443
28    CALL ERROR                                                         ABCD2444
      RETURN                                                             ABCD2445
29    AM62G = - C5 / (2.0D0 * C4)                                        ABCD2446
      GO TO 31                                                           ABCD2447
30    AM62G = (DSQRT(C6) - C5) / (2.0D0 * C4)                            ABCD2448
31    IF (AM62G .LE. 0.0D0)  GO TO 28                                    ABCD2449
      AM6G = DSQRT(AM62G)                                                ABCD2450
      V6 = AM6G * CS6                                                    ABCD2451
      HSCAL = H6 - V6 ** 2 / GAJ2                                        ABCD2452
      DELHS = HSCAL - HS6                                                ABCD2453
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 33              ABCD2454
32    TS6 = TS6 + DELHS / CP6                                            ABCD2455
      ICOMIX = 10                                                        ABCD2456
      CALL ERROR                                                         ABCD2457
33    A6G = A25 + A55                                                    ABCD2458
      C7 = DSQRT(1.0D0 + (AK6 - 1.0D0) * AM62G / 2.0D0)                  ABCD2459
      PS6 = C2 / (CAPSF * A6G * AM6G * C7)                               ABCD2460
      P6 = PS6 * DEXP((PHI6 - PHIS6) / REX6)                             ABCD2461
      CALL THERMO (P6,H6,T6,S6,XX1,1,FAR6,0)                             ABCD2462
      S6AVE = (WG24 * S25 + WG55 * S55) / WG6                            ABCD2463
      IF (S6 .GE. S6AVE)  GO TO 35                                       ABCD2464
      S6 = S6AVE                                                         ABCD2465
      P6 = DEXP(AMX * (PHI6 - S6) / RDEM)                                ABCD2466
35    IF (IGASMX .LE. 0)  GO TO 36                                       ABCD2467
      GO TO (43,37), IGASMX                                              ABCD2468
36    T6 = T55                                                           ABCD2469
      P6 = P55                                                           ABCD2470
      H6 = H55                                                           ABCD2471
      S6 = S55                                                           ABCD2472
      WG6 = WG55                                                         ABCD2473
      PS6 = PS55                                                         ABCD2474
      FAR6 = FAR55                                                       ABCD2475
      AK6 = AK55                                                         ABCD2476
37    IF (IDES .EQ. 0)  GO TO 38                                         ABCD2477
C***  CALCULATES A6 AS A FUNCTION OF INPUT AM6                           ABCD2478
      TS6 = T6 / (1.0D0 + (((AK6 - 1.0D0) / 2.0D0) * AM6 ** 2))          ABCD2479
      DO 34  JJ = 1,50                                                   ABCD2480
      AK6P = AK6                                                         ABCD2481
      CALL PROCOM (FAR6,TS6,CS6,AK6,CP6,REX6,PHIS6,HS6)                  ABCD2482
      V6 = AM6 * CS6                                                     ABCD2483
      DELAK6 = AK6P - AK6                                                ABCD2484
      IF (DABS(DELAK6) .LE. .5D0 * TOLALL * AK6)  GO TO 54               ABCD2485
34    TS6 = T6 / (1.0D0 + (((AK6 - 1.0D0) / 2.0D0) * AM6 ** 2))          ABCD2486
      ICOMIX = 11                                                        ABCD2487
      CALL ERROR                                                         ABCD2488
54    PS6 = P6 / ((1.0D0 + (((AK6 - 1.0D0) / 2.0D0) * AM6 ** 2)) **      ABCD2489
     1 (AK6 / (AK6 - 1.0D0)))                                            ABCD2490
      AM6ABD = AM6                                                       ABCD2491
      RHO = CAPSF * PS6 / (AJ * REX6 * TS6)                              ABCD2492
      A6 = WG6 / (RHO * V6)                                              ABCD2493
      WRITE (6,52) A6                                                    ABCD2494
      GO TO 44                                                           ABCD2495
C     CALCULATES M6=F(A6DESIGN)                                          ABCD2496
```

```
38    TS6P = T6 / (1.0D0 + (((AK6 - 1.0D0) / 2.0D0) * AM6ABD ** 2))      ABCD2497
      DO 39  I = 1,50                                                    ABCD2498
      CALL PROCOM (FAR6,TS6P,CS6,AK6,CP6,REX6,PHIS6,HS6)                 ABCD2499
      PS6P = PS6 * (TS6P / TS6) ** (AK6 / (AK6 - 1.0D0))                 ABCD2500
      RHO6 = CAPSF * PS6P / (AJ * REX6 * TS6P)                           ABCD2501
      V6 = DSQRT(GAJ2 * (H6 - HS6))                                      ABCD2502
      IF ((H6 - HS6) .LT. 0.0D0)  GO TO 42                               ABCD2503
      A6P = WG6 / (RHO6 * V6)                                            ABCD2504
      DELA6 = A6P - A6                                                   ABCD2505
      V6 = WG6 / (RHO6 * A6)                                             ABCD2506
      AM6 = V6 / CS6                                                     ABCD2507
      AM62 = AM6 ** 2                                                    ABCD2508
      IF (DABS(DELA6) .LE. .002D0 * A6)  GO TO 40                        ABCD2509
39    TS6P = T6 / (1.0D0 + (((AK6 - 1.0D0) / 2.0D0) * AM62))             ABCD2510
      ICOMIX = 12                                                        ABCD2511
      CALL ERROR                                                         ABCD2512
40    TS6 = TS6P                                                         ABCD2513
      PS6 = PS6P                                                         ABCD2514
      GO TO 44                                                           ABCD2515
41    T6 = T55                                                           ABCD2516
      P6 = P55                                                           ABCD2517
      H6 = H55                                                           ABCD2518
      S6 = S55                                                           ABCD2519
      WG6 = WG55                                                         ABCD2520
      PS6 = PS55                                                         ABCD2521
      V6 = V55                                                           ABCD2522
      AM6 = AM55                                                         ABCD2523
      IF (IGASMX .EQ. 0)  A6 = A55                                       ABCD2524
      GO TO 44                                                           ABCD2525
42    WRITE (6,53) H6,HS6                                                ABCD2526
      ICOMIX = 13                                                        ABCD2527
      CALL ERROR                                                         ABCD2528
43    AM62 = AM62G                                                       ABCD2529
      AM6 = AM6G                                                         ABCD2530
      A6 = A25 + A55                                                     ABCD2531
      ICOMIX = 0                                                         ABCD2532
44    CALL COAFBN                                                        ABCD2533
      RETURN                                                             ABCD2534
45    KKGO = 1                                                           ABCD2535
      OPRDS = PRFDS * PRCDS                                              ABCD2536
      PRFNEW = PRFDS * PS55 / P25 * 1.02D0                               ABCD2537
      PRCNEW = OPRDS / PRFNEW                                            ABCD2538
      ICOMIX = 0                                                         ABCD2539
      CALL ENGBAL                                                        ABCD2540
      RETURN                                                             ABCD2541
C                                                                        ABCD2542
C                                                                        ABCD2543
46    FORMAT (22H0SQRT OF H55-HS55 NEG ,6E15.6,6H$$$$$$)                 ABCD2544
47    FORMAT (20H0TURBINE AREA DESIGN, 6X, 6H  A55=, E15.8, 8H   AM55=,  ABCD2545
     1 E15.8)                                                            ABCD2546
48    FORMAT (22H0SQRT OF H25-HS25 NEG ,6E15.6,6H$$$$$$)                 ABCD2547
49    FORMAT (25H0TURBINE/DUCT AREA DESIGN,7H   A55=,E15.8,8H   AM55=,   ABCD2548
     1 E15.8,8H    A25=,E15.8,8H   AM25=,E15.8)                          ABCD2549
50    FORMAT (12H0COMIX PCNF=,F7.4,4H AM=,F8.6,5H P55=,F9.5,6H PS55=,    ABCD2550
```

```
     1 F9.5,5H P25=,F9.5,6H PS25=,F9.5,6H$$$$$$)                                ABCD2551
51    FORMAT (10HOCOMIX ZF=,F8.5,4H AM=,F8.6,5H P55=,F9.5,6H PS55=,F9.5,ABCD2552
     1 5H P25=,F9.5,6H PS25=,F9.5,6H$$$$$$)                                     ABCD2553
52    FORMAT (3X,27HAFTERBURNER DESIGN AREA A6 ,F8.3)                           ABCD2554
53    FORMAT (3X,18HNEG.HS6 FACTOR H6 ,F9.4,3X,4HHS6 ,F9.4)                     ABCD2555
      END                                                                       ABCD2556
```

Subroutine COMNOZ

```
      SUBROUTINE COMNOZ                                                         ABCD2557
      IMPLICIT  REAL*8  (A-H,O-Z)                                               ABCD2558
      COMMON /COMALL/  COM(1062)                                                ABCD2559
      DIMENSION  WORD(2), ERR(9)                                                ABCD2560
      DIMENSION  AWORD(2)                                                       ABCD2561
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (IAFTBN, COM(12)),ABCD2562
     1 (IMCD, COM(14)), (IMSHOC, COM(16)), (NOZFLT, COM(17)),                   ABCD2563
     2 (ERR(1), COM(24)), (P1, COM(33)), (WG7, COM(334)),                       ABCD2564
     3 (FAR7, COM(335)), (T7, COM(343)), (H7, COM(344)), (S7, COM(345)),ABCD2565
     4 (A8, COM(346)), (A8SAV, COM(347)), (TS8, COM(348)),                      ABCD2566
     5 (PS8, COM(349)), (V8, COM(350)), (AM8, COM(351)), (P8, COM(352)),ABCD2567
     6 (P8, COM(353)), (H8, COM(354)), (S8, COM(355)), (A9, COM(356)),          ABCD2568
     7 (A9SAV, COM(357)), (TS9, COM(358)), (PS9, COM(359)),                     ABCD2569
     8 (V9, COM(360)), (AM9, COM(361)), (T9, COM(362)), (P9, COM(363)),ABCD2570
     9 (H9, COM(364)), (S9, COM(365)), (P7, COM(389)),                          ABCD2571
     1 (ISPOOL, COM(1044)), (ITRAN, COM(1049))                                  ABCD2572
      DATA  AWORD /4HMNOZ, 4HZL  /                                              ABCD2573
      WORD(1) = AWORD(1)                                                        ABCD2574
      WORD(2) = AWORD(2)                                                        ABCD2575
      A8SAV = A8                                                                ABCD2576
      A9SAV = A9                                                                ABCD2577
      NOZM = 0                                                                  ABCD2578
      IMNOZ = 0                                                                 ABCD2579
      IF (ITRAN .EQ. 1)  CALL NOZCTR                                            ABCD2580
      IF (NOZFLT .EQ. 1 .OR. NOZFLT .EQ. 3)  NOZM = 1                           ABCD2581
      IF (IDES .EQ. 1 .OR. IAFTBN .GT. 0 .OR. NOZM .EQ. 1)  IMNOZ = 1           ABCD2582
      IF (ITRAN .EQ. 1)  IMNOZ = 0                                              ABCD2583
      IF (IMCD .EQ. 1)  GO TO 1                                                 ABCD2584
      CALL CONVRG (T7,H7,P7,S7,FAR7,WG7,P1,IMNOZ,A8,P7R,T8,H8,P8,S8,TS8,ABCD2585
     1 PS8,V8,AM8,ICON)                                                         ABCD2586
      GO TO (3,3,3,2), ICON                                                     ABCD2587
1     CALL CONDIV (T7,H7,P7,S7,FAR7,WG7,P1,IMNOZ,A8,A9,P7R,T8,H8,P8,S8, ABCD2588
     1 T9,H9,P9,S9,TS8,TS9,PS8,PS9,V8,V9,AM8,AM9,ICON)                          ABCD2589
      IMSHOC = ICON                                                             ABCD2590
      GO TO (4,4,4,2), ICON                                                     ABCD2591
2     CALL ERROR                                                                ABCD2592
3     T9 = T8                                                                   ABCD2593
      H9 = H8                                                                   ABCD2594
      P9 = P8                                                                   ABCD2595
      S9 = S8                                                                   ABCD2596
      TS9 = TS8                                                                 ABCD2597
      PS9 = PS8                                                                 ABCD2598
      V9 = V8                                                                   ABCD2599
      AM9 = AM8                                                                 ABCD2600
      A9 = A8                                                                   ABCD2601
```

```
      IMSHOC = ICON + 3                                                 ABCD2602
4     ERR(6) = (P7R - P7) / P7R                                         ABCD2603
      IF (ISPOOL .EQ. 1)  ERR(3) = ERR(6)                               ABCD2604
      IF (IMNOZ .EQ. 1)  WRITE (6,5) A8,AM8,A9,AM9                      ABCD2605
      RETURN                                                            ABCD2606
C                                                                       ABCD2607
C                                                                       ABCD2608
5     FORMAT (14H0NOZZLE DESIGN,10X,3H     A8=,E15.8,8H     AM8=,E15.8,  ABCD2609
     1 8H     A9=,E15.8,8H     AM9=,E15.8)                             ABCD2610
      END                                                               ABCD2611
```

Subroutine CONDIV

```
      SUBROUTINE CONDIV (TI,HI,PI,SI,FAR,WG,PA,IDES,AT,AO,PIR,TT,HT,PT, ABCD2612
     1 ST,TO,HO,PO,SO,TST,TSO,PST,PSO,VT,VO,AMT,AMO,ICON)               ABCD2613
C     ICON=1  SUBSONIC, COMPARE PIR WITH PI                             ABCD2614
C     ICON=2  SONIC, SHOCK INSIDE NOZZLE, COMPARE PIR WITH PI           ABCD2615
C     ICON=3  SONIC, SHOCK OUTSIDE NOZZLE, COMPARE PIP WITH PI          ABCD2616
C     ICON=4  ERROR                                                     ABCD2617
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD2618
      LOGICAL  ZI                                                       ABCD2619
      COMMON /COMALL/  COM(1062)                                        ABCD2620
      DIMENSION  Q(9)                                                   ABCD2621
      EQUIVALENCE  (TOLALL, COM(23)), (ZI, COM(1055))                   ABCD2622
      Q(2) = 0.0D0                                                      ABCD2623
      Q(3) = 0.0D0                                                      ABCD2624
      IF (ZI)  GO TO 100                                                ABCD2625
      AJ = 778.26D0                                                     ABCD2626
      CAPSF = 2116.2170D0                                               ABCD2627
      G = 32.174049D0                                                   ABCD2628
      GO TO 101                                                         ABCD2629
100   AJ = 1.0D0                                                        ABCD2630
      CAPSF = 101325.0D0                                                ABCD2631
      G = 1.0D0                                                         ABCD2632
101   GAJ2 = 2.0 * G * AJ                                               ABCD2633
      CALL PROCOM (FAR,TI,XX1,XX2,XX3,XX4,PHII,XX6)                     ABCD2634
C *** SONIC CALCULATIONS                                                ABCD2635
      TSS = 0.833D0 * TI                                                ABCD2636
      DO 2  J = 1,50                                                    ABCD2637
      CALL PROCOM (FAR,TSS,CSS,AK,CP,REXS,PHISS,HSS)                    ABCD2638
      HSCAL = HI - CSS ** 2 / GAJ2                                      ABCD2639
      DELHS = HSCAL - HSS                                               ABCD2640
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 4              ABCD2641
2     TSS = TSS + DELHS / CP                                            ABCD2642
3     ICON = 4                                                          ABCD2643
      RETURN                                                            ABCD2644
4     IF (IDES .LE. 0)  GO TO 11                                        ABCD2645
C *** SONIC DESIGN, CALCULATE AT                                        ABCD2646
      VT = CSS                                                          ABCD2647
      TST = TSS                                                         ABCD2648
      PST = PI * (TST / TI) ** (AK / (AK - 1.0D0))                      ABCD2649
      RHO = CAPSF * PST / (AJ * REXS * TST)                             ABCD2650
      AT = WG / (RHO * VT)                                              ABCD2651
      AMT = 1.0D0                                                       ABCD2652
```

```
C *** IDEAL EXPANSION DESIGN, CALCULATE AO                                    ABCD2653
      PSO = PA                                                                ABCD2654
      TSO = TI * (PSO / PI) ** .286D0                                         ABCD2655
      DO 7  J = 1,50                                                          ABCD2656
      CALL PROCOM (FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)                           ABCD2657
      PHICAL = PHII - REX * DLOG(PI / PSO)                                    ABCD2658
      DELPHI = PHICAL - PHISO                                                 ABCD2659
      IF (DABS(DELPHI) .LE. .1D0 * TOLALL * PHICAL)  GO TO 8                  ABCD2660
7     TSO = TSO * DEXP(4.0D0 * DELPHI)                                        ABCD2661
      GO TO 3                                                                 ABCD2662
8     VO = DSQRT(GAJ2 * (HI - HSO))                                           ABCD2663
      AMO = VO / CSO                                                          ABCD2664
      AKP1 = AK + 1.0D0                                                       ABCD2665
      AKM1 = AK - 1.0D0                                                       ABCD2666
      AO = (AT / AMO) * (2.0D0 * (1.0D0 + AKM1 * AMO ** 2 / 2.0D0) /          ABCD2667
     1 AKP1) ** (AKP1 / (2.0D0 * AKM1))                                       ABCD2668
      PIR = PI                                                                ABCD2669
      ICON = 3                                                                ABCD2670
9     TO = TI                                                                 ABCD2671
      HO = HI                                                                 ABCD2672
      PO = PI                                                                 ABCD2673
      SO = SI                                                                 ABCD2674
10    TT = TI                                                                 ABCD2675
      HT = HI                                                                 ABCD2676
      PT = PI                                                                 ABCD2677
      ST = SI                                                                 ABCD2678
      RETURN                                                                  ABCD2679
C *** ASSUME SONIC THROAT AND ISENTROPIC EXPANSION TO AO                      ABCD2680
11    VT = CSS                                                                ABCD2681
      AMT = 1.0D0                                                             ABCD2682
      TST = TSS                                                               ABCD2683
      RHO = WG / (AT * VT)                                                    ABCD2684
      PST = RHO * AJ * REXS * TST / CAPSF                                     ABCD2685
      PIR = PST * (TI / TST) ** (AK / (AK - 1.0D0))                           ABCD2686
      IF (PST .GE. PA)  GO TO 27                                              ABCD2687
      TSO = 0.95D0 * TI                                                       ABCD2688
      MAM = 0                                                                 ABCD2689
13    CALL PROCOM (FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)                           ABCD2690
      AKP1 = AK + 1.0D0                                                       ABCD2691
      AKM1 = AK - 1.0D0                                                       ABCD2692
      AMO = DSQRT(2.0D0 * ((TI / TSO) - 1.0D0) / AKM1)                        ABCD2693
      AOCAL = (AT / AMO) * (2.0D0 * (1.0D0 + AKM1 * AMO ** 2 / 2.0D0) /       ABCD2694
     1 AKP1) ** (AKP1 / (2.0D0 * AKM1))                                       ABCD2695
      EA = (AO - AOCAL) / AO                                                  ABCD2696
      DIR = DSQRT(AO / AOCAL)                                                 ABCD2697
      CALL AFQUIR (Q(1),TSO,EA,0.0D0,100.0D0,.1D0*TOLALL,DIR,TSOT,JCON)       ABCD2698
      GO TO (14,18,3), JCON                                                   ABCD2699
14    TSO = TSOT                                                              ABCD2700
      IF (TSO - TI)  15,13,16                                                 ABCD2701
15    TSC = 2.0D0 * TI / (AK + 1.0D0)                                         ABCD2702
      IF (TSO .GT. TSC)  GO TO 17                                             ABCD2703
16    TSO = 0.98D0 * TI                                                       ABCD2704
      GO TO 13                                                                ABCD2705
17    IF (Q(2) .LT. 30.0D0 .OR. AMO .LT. 0.95D0 .OR. MAM .EQ. 1)              ABCD2706
```

```
     1 GO TO 13                                                          ABCD2707
       TSO = 2.0D0 * TI / (2.0D0 + 0.98D0 * (AK - 1.0D0))                ABCD2708
       MAM = 1                                                           ABCD2709
       GO TO 13                                                          ABCD2710
18     PSO = PIR * (TSO / TI) ** (AK / (AK - 1.0D0))                     ABCD2711
       IF (PSO - PA)  20,19,27                                           ABCD2712
C *** CRITICAL FLOW, ISENTROPIC EXPANSION TO PA                          ABCD2713
19     VO = AMO * CSO                                                    ABCD2714
       ICON = 1                                                          ABCD2715
       GO TO 9                                                           ABCD2716
C *** SUBSONIC FLOW                                                      ABCD2717
20     PSO = PA                                                          ABCD2718
       Q(2) = 0.0D0                                                      ABCD2719
       Q(3) = 0.0D0                                                      ABCD2720
       TSO = 0.833D0 * TI                                                ABCD2721
       DO 22  J = 1,50                                                   ABCD2722
       CALL PROCOM (FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)                     ABCD2723
       RHO = CAPSF * PSO / (AJ * REX * TSO)                              ABCD2724
       VO = WG / (RHO * AO)                                              ABCD2725
       HSCAL = HI - VO ** 2 / GAJ2                                       ABCD2726
       DELHS = HSCAL - HSO                                               ABCD2727
       IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 23             ABCD2728
22     TSO = TSO + DELHS / CP                                            ABCD2729
       GO TO 3                                                           ABCD2730
23     AMO = VO / CSO                                                    ABCD2731
       PIR = PSO * (TI / TSO) ** (AK / (AK - 1.0D0))                     ABCD2732
       TST = TSO                                                         ABCD2733
24     CALL PROCOM (FAR,TST,CST,AK,CP,REX,PHIST,HST)                     ABCD2734
       PST = PIR * (TST / TI) ** (AK / (AK - 1.0D0))                     ABCD2735
       RHO = PST * CAPSF / (AJ * REX * TST)                              ABCD2736
       VT = WG / (RHO * AT)                                              ABCD2737
       HSCAL = HI - VT ** 2 / GAJ2                                       ABCD2738
       EH = (HSCAL - HST) / HSCAL                                        ABCD2739
       DIR = 1.0D0 + (HSCAL - HST) / (CP * TST)                          ABCD2740
       CALL AFQUIR (Q(1),TST,EH,0.0D0,20.0D0,.5D0*TOLALL,DIR,TSTT,JCON)  ABCD2741
       GO TO (25,26,3), JCON                                             ABCD2742
25     TST = TSTT                                                        ABCD2743
       GO TO 24                                                          ABCD2744
26     AMT = VT / CST                                                    ABCD2745
       ICON = 1                                                          ABCD2746
       GO TO 9                                                           ABCD2747
C *** SUPERCRITICAL FLOW, ISENTROPIC EXPANSION TO PA                     ABCD2748
27     PSO = PA                                                          ABCD2749
       TSO = TI * (PSO / PIR) ** .286D0                                  ABCD2750
       DO 29  J = 1,50                                                   ABCD2751
       CALL PROCOM (FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)                     ABCD2752
       PHICAL = PHII - REX * DLOG(PIR / PSO)                             ABCD2753
       DELPHI = PHICAL - PHISO                                           ABCD2754
       IF (DABS(DELPHI) .LE. .1D0 * TOLALL * PHICAL)  GO TO 30           ABCD2755
29     TSO = TSO * DEXP(4.0D0 * DELPHI)                                  ABCD2756
       GO TO 3                                                           ABCD2757
30     VO = DSQRT(GAJ2 * (HI - HSO))                                     ABCD2758
       AMO = VO / CSO                                                    ABCD2759
       AKP1 = AK + 1.0D0                                                 ABCD2760
```

```
      AKM1 = AK - 1.0D0                                                 ABCD2761
      AOID = (AT / AMO) * (2.0D0 * (1.0D0 + AKM1 * AMO ** 2 / 2.0D0) /  ABCD2762
     1 AKP1) ** (AKP1 / (2.0D0 * AKM1))                                 ABCD2763
      ICON = 3                                                          ABCD2764
      N = 0                                                             ABCD2765
      IF (AO - AOID)  31,9,32                                           ABCD2766
C *** SUPERCRITICAL FLOW, ISENTROPIC EXPANSION TO AO                    ABCD2767
31    N = 1                                                             ABCD2768
32    TSO = 0.833D0 * TI                                                ABCD2769
      DO 34  J = 1,100                                                  ABCD2770
      CALL PROCOM (FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)                     ABCD2771
      AKP1 = AK + 1.0D0                                                 ABCD2772
      AKM1 = AK - 1.0D0                                                 ABCD2773
      AMO = DSQRT(2.0D0 * ((TI / TSO) - 1.0D0) / AKM1)                  ABCD2774
      AOCAL = (AT / AMO) * (2.0D0 * (1.0D0 + AKM1 * AMO ** 2 / 2.0D0) / ABCD2775
     1 AKP1) ** (AKP1 / (2.0D0 * AKM1))                                 ABCD2776
      DELA = AO - AOCAL                                                 ABCD2777
      IF (DABS(DELA) .LE. .1D0 * TOLALL * AO)  GO TO 35                 ABCD2778
34    TSO = TSO * DSQRT(AOCAL / AO)                                     ABCD2779
      GO TO 3                                                           ABCD2780
35    IF (N .LE. 0)  GO TO 37                                           ABCD2781
C *** UNDEREXPANDED, SHOCK OUTSIDE NOZZLE                               ABCD2782
      PSO = PIR * (TSO / TI) ** (AK / (AK - 1.0D0))                     ABCD2783
      VO = AMO * CSO                                                    ABCD2784
      GO TO 9                                                           ABCD2785
C *** OVEREXPANDED, FIND SHOCK POSITION                                 ABCD2786
37    PSX = PIR * (TSO / TI) ** (AK / (AK - 1.0D0))                     ABCD2787
      PSY = PSX * (2.0D0 * AK * AMO ** 2 / (AK + 1.0D0) - (AK - 1.0D0) /ABCD2788
     1 (AK + 1.0D0))                                                    ABCD2789
      IF (PA .GE. PSY)  GO TO 39                                        ABCD2790
C *** OVEREXPANDED, SHOCK OUTSIDE NOZZLE                                ABCD2791
      PSO = PSX                                                         ABCD2792
      VO = AMO * CSO                                                    ABCD2793
      GO TO 9                                                           ABCD2794
C *** OVEREXPANDED, SHOCK INSIDE NOZZLE                                 ABCD2795
39    PSO = PA                                                          ABCD2796
      TSO = 0.833D0 * TI                                                ABCD2797
      DO 41  J = 1,50                                                   ABCD2798
      CALL PROCOM (FAR,TSO,CSO,AK,CP,REX,PHISO,HSO)                     ABCD2799
      RHO = CAPSF * PSO / (AJ * REX * TSO)                              ABCD2800
      VO = WG / (RHO * AO)                                              ABCD2801
      HSCAL = HI - VO ** 2 / GAJ2                                       ABCD2802
      DELHS = HSCAL - HSO                                               ABCD2803
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 42             ABCD2804
41    TSO = TSO + DELHS / CP                                            ABCD2805
      GO TO 3                                                           ABCD2806
42    AMO = VO / CSO                                                    ABCD2807
      TO = TI                                                           ABCD2808
      HO = HI                                                           ABCD2809
      PO = PSO * (TO / TSO) ** (AK / (AK - 1.0D0))                      ABCD2810
      SO = PHII - REX * DLOG(PO)                                        ABCD2811
      ICON = 2                                                          ABCD2812
      GO TO 10                                                          ABCD2813
      END                                                               ABCD2814
```

Subroutine CONOUT

```
      SUBROUTINE CONOUT (ICON)                                                ABCD2815
      IMPLICIT  REAL*8  (A-H,O-Z)                                             ABCD2816
      LOGICAL  SI                                                             ABCD2817
      COMMON /COMALL/  COM(1062)                                              ABCD2818
      DIMENSION  WORDY(368,2), IOUT(150), AOUT(6), WOUT(6,2),                 ABCD2819
     1 WOR11(152), WOR21(152), WOR31(64), WOR12(152), WOR22(152),             ABCD2820
     2 WOR32(64), THEEND(2), BLANK(2), AIN(2), CHANGE(2)                      ABCD2821
      EQUIVALENCE  (TIME,COM(993)), (ITRAN,COM(1049)), (SI,COM(1055))         ABCD2822
      EQUIVALENCE  (WOR11(1),WORDY(1,1)), (WOR21(1), WORDY(153,1)),           ABCD2823
     1 (WOR31(1),WORDY(305,1)), (WOR12(1),WORDY(1,2)),                        ABCD2824
     2 (WOR22(1),WORDY(153,2)), (WOR32(1),WORDY(305,2))                       ABCD2825
      DATA WOR11 /                                                            ABCD2826
     1 4HP1  , 4HH22 , 4HAM23, 4HWA23, 4HT23 , 4HP23 , 4HH23 , 4HS23 ,       ABCD2827
     2 4HA24 , 4HT24 , 4HH24 , 4HS24 , 4HAM25, 4HT25 , 4HP25 , 4HH25 ,       ABCD2828
     3 4HS25 , 4HA28 , 4HA28S, 4HAM28, 4HV28 , 4HTS28, 4HPS28, 4HT28 ,       ABCD2829
     4 4HP28 , 4HH28 , 4HS28 , 4HA29 , 4HA29S, 4HAM29, 4HV29 , 4HTS29,       ABCD2830
     5 4HPS29, 4HT29 , 4HP29 , 4HH29 , 4HS29 , 4HBYPA, 4HWAD , 4HWFD ,       ABCD2831
     6 4HETAD, 4HDPDU, 4HDPDU, 4HXXP1, 4HXWG2, 4HXFAR, 4HXT25, 4HXP25,       ABCD2832
     7 4HXH25, 4HXS25, 4HXWG5, 4HXFAR, 4HXT55, 4HXP55, 4HXH55, 4HXS55,       ABCD2833
     8 4HXWFB, 4HXWFD, 4HT2DS, 4HT2  , 4HP2  , 4HH2  , 4HS2  , 4HS22 ,       ABCD2834
     9 4HT22D, 4HA25 , 4HV25 , 4HT4GU, 4HT4DS, 4HT5  , 4HH5  , 4HS5  ,       ABCD2835
     1 4HWG5 , 4HFAR5, 4HAM55, 4HV55 , 4HA55 , 4HPS55, 4HS6  , 4HPS6 ,       ABCD2836
     2 4HV6  , 4HTFHP, 4HCNHP, 4HETHP, 4HDHHP, 4HTFHP, 4HCNHP, 4HETHP,       ABCD2837
     3 4HPRFC, 4HETAF, 4HWAFC, 4HPCNF, 4HPRFD, 4HETAF, 4HWAFD, 4HPCNC,       ABCD2838
     4 4HHPEX, 4HDUMD, 4HPRF , 4HETAF, 4HZCDS, 4HCNF , 4HWAFC, 4HZF  ,       ABCD2839
     5 4HPCNF, 4HPCBL, 4HZI  , 4HPCNI, 4HTFFI, 4HCNIP, 4HETAT, 4HDHTC,       ABCD2840
     6 4HDHTI, 4HZIDS, 4HPCNI, 4HPCNI, 4HT1  , 4HH1  , 4HS1  , 4HT3  ,       ABCD2841
     7 4HH3  , 4HWA3 , 4HWA3C, 4HT4  , 4HH4  , 4HS4  , 4HWG4 , 4HFAR4,       ABCD2842
     8 4HT50 , 4HH50 , 4HS50 , 4HWG50, 4HFAR5, 4HPCBL, 4HPCBL, 4HPCBL,       ABCD2843
     9 4HPCBL, 4HPCBL, 4HCNHP, 4HETAT, 4HDHTC, 4HDHTC, 4HTFFH, 4HPFIC,       ABCD2844
     1 4HETAI, 4HWAIC, 4HETAB, 4HPRID, 4HETAI, 4HWAID, 4HWAIC, 4HETAB /      ABCD2845
      DATA WOR21 /                                                            ABCD2846
     1 4HWFBD, 4HZFDS, 4HETAR, 4HETAI, 4HPRI , 4HETAB, 4HWAC , 4HWFB ,       ABCD2847
     2 4HBLC , 4HCS  , 4HAM  , 4HALTP, 4HDPCO, 4HDPCO, 4HPCNF, 4HXP1 ,       ABCD2848
     3 4HXT21, 4HXP21, 4HXH21, 4HXS21, 4HXH3 , 4HXWAF, 4HXWAC, 4HXBLF,       ABCD2849
     4 4HXBLD, 4HWG37, 4HA38 , 4HAM38, 4HV38 , 4HT38 , 4HH38 , 4HP38 ,       ABCD2850
     5 4HTS38, 4HPS38, 4HA39 , 4HAM39, 4HV39 , 4HT39 , 4HH39 , 4HP39 ,       ABCD2851
     6 4HTS39, 4HPS39, 4HWA32, 4HDPWI, 4HBPRI, 4HFGMW, 4HFGPW, 4HFNWI,       ABCD2852
     7 4HFNMA, 4HFWOV, 4HDPWG, 4HDELF, 4HDELF, 4HDELS, 4HCVDW, 4HCVDN,       ABCD2853
     8 4HCVMN, 4HVA  , 4HVJD , 4HVJW , 4HVJM , 4HWFT , 4HWGT , 4HSFC ,       ABCD2854
     9 4HTFAB, 4HFRD , 4HFGMD, 4HFGMM, 4HFGPD, 4HFGPM, 4HFGM , 4HFGP ,       ABCD2855
     1 4HFG  , 4HFN  , 4HFFOV, 4HFCOV, 4HFMNO, 4HFNOV, 4HT21 , 4HH21 ,       ABCD2856
     2 4HS21 , 4HWA21, 4HT21D, 4HT22 , 4HWA22, 4HS3  , 4HWA32, 4HT55 ,       ABCD2857
     3 4HH55 , 4HS55 , 4HTFLP, 4HCNLP, 4HETLP, 4HTFIP, 4HCNIP, 4HETIP,       ABCD2858
     4 4HTFLP, 4HCNLP, 4HETLP, 4HDHLP, 4HTFIP, 4HCNIP, 4HETIP, 4HDHIP,       ABCD2859
     5 4HTFFL, 4HCNLP, 4HETAT, 4HDHTC, 4HDHTF, 4HPRCC, 4HETAC, 4HWACC,       ABCD2860
     6 4HPRCD, 4HETAC, 4HWACD, 4HZC  , 4HPCNC, 4HPCBL, 4HPCNC, 4HPCBL,       ABCD2861
     7 4HPCBL, 4HCNC , 4HPRC , 4HETAC, 4HCNI , 4HWACI, 4HWAI , 4HBLI ,       ABCD2862
     8 4HBLHP, 4HBLIP, 4HBLLP, 4HBLF , 4HBLDU, 4HBLOB, 4HWAF , 4HWACC,       ABCD2863
     9 4HWG24, 4HFAR2, 4HWG55, 4HFAR5, 4HP6DS, 4HAM6D, 4HAM6 , 4HA6  ,       ABCD2864
     1 4HWG6C, 4HWG6 , 4HT6  , 4HP6  , 4HH6  , 4HWG7 , 4HFAR7, 4HFAR7 /      ABCD2865
```

```
      DATA WOR31 /                                                          ABCD2866
     1 4HTS7 , 4HPS7 , 4HV7   , 4HAM7 , 4HA7  , 4HT7DS, 4HT7  , 4HH7  ,    ABCD2867
     2 4HS7  , 4HA8  , 4HA8SA, 4HTS8 , 4HPS8 , 4HV8  , 4HAM8 , 4HT8  ,    ABCD2868
     3 4HP8  , 4HH8  , 4HS8  , 4HA9  , 4HA9SA, 4HTS9 , 4HPS9 , 4HV9  ,    ABCD2869
     4 4HAM9 , 4HT9  , 4HP9  , 4HH9  , 4HS9  , 4HETAA, 4HDPAF, 4HWFA ,    ABCD2870
     5 4HETAA, 4HETAA, 4HDPAF, 4HXNHP, 4HXNIP, 4HXNLP, 4HP22 , 4HU22 ,    ABCD2871
     6 4HP21 , 4HU21 , 4HP3  , 4HU3  , 4HP4  , 4HU4  , 4HP50 , 4HU50 ,    ABCD2872
     7 4HP5  , 4HU5  , 4HP55 , 4HU55 , 4HP7  , 4HU7  , 4HP24 , 4HU24 ,    ABCD2873
     8 4HP37 , 4HU37 , 4HDUMS, 4HFXFN, 4HFXM2, 4HAFTF, 4HFAN , 4HISPO /   ABCD2874
      DATA WOR12 /                                                          ABCD2875
     1 3 * 4H    , 4HDS  , 14 * 4H    , 4HAV  , 9 * 4H    , 4HAV  ,       ABCD2876
     2 8 * 4H    , 4HSS  , 3 * 4H     , 4HC   , 4HDS  , 4H    , 4H4   ,    ABCD2877
     3 4H24  , 4 * 4H    , 4H5   , 4H55  , 12 * 4H    , 4HS   ,           ABCD2878
     4 16 * 4H    , 4 * 4HCF  , 3 * 4HDS  , 4HF   , 4HCF  , 4HF   ,       ABCD2879
     5 4HDS  , 4HS   , 4HDS  , 4HS   , 4HGU  , 4HT   , 4H1   ,           ABCD2880
     6 7 * 4H    , 4HF   , 2 * 4H    , 4HP   , 4H    , 2 * 4HIP  ,        ABCD2881
     7 2 * 4H    , 4HDS  , 4HGU  , 6 * 4H    , 4HDS  , 9 * 4H    ,       ABCD2882
     8 4H0   , 4HHP  , 4HIP  , 4HLP  , 4HDU  , 4HOB  , 4H    ,           ABCD2883
     9 2 * 4HHP  , 4H    , 4HP   , 4HF   , 4HCF  , 4HF   , 4HCF  ,       ABCD2884
     1 4HS   , 4HDS  , 4HS   , 2 * 4HDS   /                               ABCD2885
      DATA WOR22 /                                                          ABCD2885
     1 4HS   , 11 * 4H    , 4HDS  , 4HM   , 4HGU  , 9 * 4H    , 4HU   ,    ABCD2887
     2 17 * 4H    , 4HDS  , 4HNG  , 4HNT  , 3 * 4HNG  , 4HIN  , 4HFN  ,    ABCD2888
     3 4HDS  , 4HG   , 4HN   , 4HFC  , 4HNG  , 2 * 4HOZ  , 17 * 4H    ,    ABCD2889
     4 3 * 4HFN  , 4HFD  , 4 * 4H    , 4HS   , 7 * 4H    , 6 * 4HDS  ,       ABCD2890
     5 8 * 4HCF  , 4HP   , 4H    , 2 * 4HLP  , 4H    , 4HF   , 4HCF  ,       ABCD2891
     6 4HF   , 4HS   , 4HDS  , 4HS   , 2 * 4H    , 4HC   , 4HDS  ,       ABCD2892
     7 4HI   , 4HID  , 16 * 4H    , 4H4   , 4H    , 4H5   , 4HAV  ,       ABCD2893
     8 4HSV  , 2 * 4H    , 4HDS  , 6 * 4H    , 4HSV  /                    ABCD2894
      DATA WOR32 /                                                          ABCD2895
     1 10 * 4H    , 4HV   , 9 * 4H    , 4HV   , 8 * 4H    , 2 * 4HDS  ,    ABCD2896
     2 2 * 4H    , 4HSV  , 4HT   , 23 * 4H    , 4HPL  , 4H2M  , 4HCP  ,    ABCD2897
     3 4HAN  , 4H    , 4HOL  /                                             ABCD2898
      DATA  THEEND, BLANK, LIMIT /4HTHEE, 4HND  , 4H    , 4H    , 368 /   ABCD2899
      IF (ICON .EQ. 1)  GO TO 24                                            ABCD2900
      IF (SI)  GO TO 22                                                     ABCD2901
      WRITE (6,21)                                                          ABCD2902
      GO TO 24                                                              ABCD2903
22    WRITE (6,23)                                                          ABCD2904
24    GO TO (1,6), ICON                                                     ABCD2905
C *** INPUT SECTION                                                         ABCD2906
1     DO 4  N = 1,150                                                       ABCD2907
      NUM = N                                                               ABCD2908
      READ (5,11) AIN,CHANGE                                                ABCD2909
      IF (AIN(1) .EQ. THEEND(1) .AND. AIN(2) .EQ. THEEND(2))  GO TO 5       ABCD2910
      DO 2  J = 1,LIMIT                                                     ABCD2911
      JJ = J                                                                ABCD2912
      IF (AIN(1) .EQ. WORDY(J,1) .AND. AIN(2) .EQ. WORDY(J,2))  GO TO 3     ABCD2913
2     CONTINUE                                                              ABCD2914
      WRITE (6,12) AIN                                                      ABCD2915
      GO TO 4                                                               ABCD2916
3     IOUT(NUM) = JJ                                                        ABCD2917
      IF (CHANGE(1) .EQ. BLANK(1) .AND. CHANGE(2) .EQ. BLANK(2))            ABCD2918
     1 GO TO 4                                                              ABCD2919
```

```
      WORDY(JJ,1) = CHANGE(1)                                                ABCD2920
      WORDY(JJ,2) = CHANGE(2)                                                ABCD2921
4     CONTINUE                                                               ABCD2922
      WRITE (6,13)                                                           ABCD2923
5     NUM = NUM - 1                                                          ABCD2924
      RETURN                                                                 ABCD2925
C *** OUTPUT SECTION                                                         ABCD2926
6     IF (NUM .EQ. 1)  RETURN                                                ABCD2927
C                                                                            ABCD2928
C     THE FOLLOWING THREE STATEMENTS ARE USED AT LEWIS ONLY                  ABCD2929
      IF (TIME .EQ. 0.0D0 .AND. ITRAN .NE. 1)  GO TO 16                      ABCD2930
      WRITE (50,19)  TIME, COM(248), COM(257), COM(258)                      ABCD2931
      WRITE (50,20)  (COM(I),  I = 372,394)                                  ABCD2932
C                                                                            ABCD2933
16    N = NUM                                                                ABCD2934
      J = 6                                                                  ABCD2935
      DO 9  I = 1,NUM,6                                                      ABCD2936
      IF (N .GT. 6)  GO TO 7                                                 ABCD2937
      J = N                                                                  ABCD2938
7     N = N - 6                                                              ABCD2939
      DO 8  K = 1,J                                                          ABCD2940
      L = I + K - 1                                                          ABCD2941
      M = IOUT(L)                                                            ABCD2942
      WOUT(K,1) = WORDY(M,1)                                                 ABCD2943
      WOUT(K,2) = WORDY(M,2)                                                 ABCD2944
      IF (M .LE. 362)  AOUT(K) = COM(M+32)                                   ABCD2945
      IF (M .GT. 362 .AND. M .LE. 367)  AOUT(K) = COM(M+694)                 ABCD2946
      IF (M .GT. 367)  AOUT(K) = COM(1044)                                   ABCD2947
8     CONTINUE                                                               ABCD2948
      WRITE (6,14) (WOUT(K,1), WOUT(K,2),  K = 1,J)                          ABCD2949
      WRITE (6,15) (AOUT(K),K=1,J)                                           ABCD2950
      IF (N .LE. 0)  RETURN                                                  ABCD2951
9     CONTINUE                                                               ABCD2952
      RETURN                                                                 ABCD2953
C                                                                            ABCD2954
C                                                                            ABCD2955
C                                                                            ABCD2956
11    FORMAT  (A4, A2, 6X, A4, A2)                                           ABCD2957
12    FORMAT  (1H0, 9HTHE WORD , A4, A2, 26H NOT FOUND IN COMMON ARRAY)      ABCD2958
13    FORMAT (22H0ERROR IN CONOUT INPUT)                                     ABCD2959
14    FORMAT  (1H0, 25X, A4, A2, 5(9X, A4, A2))                              ABCD2960
15    FORMAT  (21X, 6E15.6)                                                  ABCD2961
C                                                                            ABCD2962
C     THE FOLLOWING TWO FORMATS ARE USED AT LEWIS ONLY                       ABCD2963
19    FORMAT (1X, 1PE20.6)                                                   ABCD2964
20    FORMAT (1X, 1P6E20.6)                                                  ABCD2965
C                                                                            ABCD2966
21    FORMAT  (1X, 30HTHE OUTPUT IS IN ENGLISH UNITS)                        ABCD2967
23    FORMAT  (1X, 25HTHE OUTPUT IS IN SI UNITS)                             ABCD2968
      END                                                                    ABCD2969
```

Subroutine CONVRG

```
      SUBROUTINE CONVRG (TI,HI,PI,SI,FAR,WG,PA,IDES,AO,PR,TO,HO,PO,SO,   ABCD2970

     1 TSO,PSO,VO,AMO,ICON)                                              ABCD2971
C     ICON=1      SUBSONIC, COMPARE PI WITH PR                          ABCD2972
C     ICON=2      SONIC, COMPARE PI WITH PR                             ABCD2973
C     ICON=4      ERROR                                                 ABCD2974
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD2975
      LOGICAL  ZI                                                       ABCD2976
      COMMON /COMALL/  COM(1062)                                        ABCD2977
      EQUIVALENCE  (TOLALL, COM(23)), (ZI, COM(1055))                   ABCD2978
      IF (ZI)  GO TO 100                                                ABCD2979
      AJ = 778.26D0                                                     ABCD2980
      CAPSF = 2116.217D0                                                ABCD2981
      G = 32.174049D0                                                   ABCD2982
      CPG = .250D0                                                      ABCD2983
      GO TO 101                                                         ABCD2984
100   AJ = 1.0D0                                                        ABCD2985
      CAPSF = 1.0D0                                                     ABCD2986
      G = 1.0D0                                                         ABCD2987
      CPG = 1048.D0                                                     ABCD2988
101   GAJ2 = 2.0D0 * G * AJ                                             ABCD2989
      CALL PROCOM (FAR,TI,XX1,XX2,XX3,XX4,PHII,XX6)                     ABCD2990
C *** SONIC CALCULATIONS                                                ABCD2991
      TSS = 0.833D0 * TI                                                ABCD2992
      DO 2  J = 1,50                                                    ABCD2993
      CALL PROCOM (FAR,TSS,CSS,AKS,CP,REXS,PHISS,HSS)                   ABCD2994
      HSCAL = HI - CSS ** 2 / GAJ2                                      ABCD2995
      DELHS = HSCAL - HSS                                               ABCD2996
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 4              ABCD2997
2     TSS = TSS + DELHS / CP                                            ABCD2998
3     ICON = 4                                                          ABCD2999
      RETURN                                                            ABCD3000
4     IF (IDES .LE. 0)  GO TO 12                                        ABCD3001
C *** ISENTROPIC EXPANSION CALCULATIONS                                 ABCD3002
      TSI = TI * (PA / PI) ** 0.286D0                                   ABCD3003
      DO 7  J = 1,70                                                    ABCD3004
      CALL THERMO (PA,HSI,TSI,SSI,XX1,1,FAR,0)                          ABCD3005
      IF (DABS(SSI - SI) .LE. .1D0 * TOLALL * SI)  GO TO 8              ABCD3006
7     TSI = TSI / DEXP((SSI - SI) / CPG)                                ABCD3007
      GO TO 3                                                           ABCD3008
8     VIS = DSQRT(GAJ2 * (HI - HSI))                                    ABCD3009
      IF (VIS .GE. CSS)  GO TO 11                                       ABCD3010
C *** SUBSONIC DESIGN, CALCULATE AO                                     ABCD3011
      VO = VIS                                                          ABCD3012
      TSO = TSI                                                         ABCD3013
      PSO = PA                                                          ABCD3014
      CALL PROCOM (FAR,TSO,CSO,XX2,XX3,REX,PHISO,HSO)                   ABCD3015
      RHO = CAPSF * PSO / (AJ * REX * TSO)                              ABCD3016
      AO = WG / (RHO * VO)                                              ABCD3017
      AMO = VO / CSO                                                    ABCD3018
      PR = PI                                                           ABCD3019
      ICON = 1                                                          ABCD3020
10    TO = TI                                                           ABCD3021
      HO = HI                                                           ABCD3022
      PO = PI                                                           ABCD3023
      SO = SI                                                           ABCD3024
```

```
      RETURN                                                            ABCD3025
C *** SONIC DESIGN, CALCULATE AO                                        ABCD3026
11    VO = CSS                                                          ABCD3027
      TSO = TSS                                                         ABCD3028
      PSO = PI * (TSO / TI) ** (AKS / (AKS - 1.0D0))                    ABCD3029
      RHO = CAPSF * PSO / (AJ * REXS * TSO)                             ABCD3030
      AO = WG / (RHO * VO)                                              ABCD3031
      AMO = 1.0D0                                                       ABCD3032
      PR = PI                                                           ABCD3033
      ICON = 2                                                          ABCD3034
      GO TO 10                                                          ABCD3035
C *** NON-DESIGN, CALCULATE CRITICAL CONDITIONS                         ABCD3036
12    VO = CSS                                                          ABCD3037
      TSO = TSS                                                         ABCD3038
      PSO = PA                                                          ABCD3039
      RHO = CAPSF * PSO / (AJ * REXS * TSO)                             ABCD3040
      AOCRIT = WG / (RHO * VO)                                          ABCD3041
      AMO = 1.0D0                                                       ABCD3042
      PR = PSO * (TI / TSO) ** (AKS / (AKS - 1.0D0))                    ABCD3043
      IF (AO .GT. AOCRIT)  GO TO 14                                     ABCD3044
C *** NON-DESIGN, CRITICAL AND SUPERCRITICAL CONDITIONS                 ABCD3045
      PSO = PSO * AOCRIT / AO                                           ABCD3046
      PR = PR * AOCRIT / AO                                             ABCD3047
      ICON = 2                                                          ABCD3048
      GO TO 10                                                          ABCD3049
C *** NON-DESIGN, SUBSONIC CALCULATIONS                                 ABCD3050
14    PSO = PA                                                          ABCD3051
      TSO = 0.833D0 * TSO                                               ABCD3052
      DO 16  J = 1,50                                                   ABCD3053
      CALL PROCOM (FAR,TSO,CSO,AKO,CP,REX,PHISO,HSO)                    ABCD3054
      RHO = CAPSF * PSO / (AJ * REX * TSO)                              ABCD3055
      VO = WG / (RHO * AO)                                              ABCD3056
      HSCAL = HI - VO ** 2 / GAJ2                                       ABCD3057
      DELHS = HSCAL - HSO                                               ABCD3058
      IF (DABS(DELHS) .LE. .5D0 * TOLALL * HSCAL)  GO TO 17             ABCD3059
16    TSO = TSO + DELHS / CP                                            ABCD3060
      GO TO 3                                                           ABCD3061
17    AMO = VO / CSO                                                    ABCD3062
      PR = PSO * (TI / TSO) ** (AKO / (AKO - 1.0D0))                    ABCD3063
      ICON = 1                                                          ABCD3064
      GO TO 10                                                          ABCD3065
      END                                                               ABCD3066
```

Function DERIV

```
      FUNCTION DERIV (I,X)                                              ABCD3067
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD3068
      COMMON /COMALL/  COM(1062)                                        ABCD3069
      DIMENSION  FO(50,4), PVRDOT(23), XS(23)                           ABCD3070
      EQUIVALENCE  (FO(1,1), COM(430)), (DT, COM(994)),                 ABCD3071
     1 (PVRDOT(1), COM(998)), (XS(1), COM(1021)), (JTRAN, COM(1050)),   ABCD3072
     2 (IVRDOT, COM(1052)), (IAMTRX, COM(1054))                         ABCD3073
      IF (IAMTRX .EQ. 1)  GO TO 2                                       ABCD3074
      IF (JTRAN .EQ. 1)  GO TO 1                                        ABCD3075
```

```
      DERIV = 0.0D0                                                    ABCD3076
      FO(I,1) = X                                                      ABCD3077
      FO(I,2) = X                                                      ABCD3078
      FO(I,3) = DERIV                                                  ABCD3079
      FO(I,4) = DERIV                                                  ABCD3080
      RETURN                                                           ABCD3081
1     X0 = FO(I,2)                                                     ABCD3082
      DERIV = (X - X0) / DT                                            ABCD3083
      IF (DABS(DERIV) .LT. 1.0D-10)  DERIV = 0.0D0                     ABCD3084
      FO(I,1) = X                                                      ABCD3085
      FO(I,3) = DERIV                                                  ABCD3086
      RETURN                                                           ABCD3087
2     IF (I .EQ. IVRDOT)  GO TO 3                                      ABCD3088
      DERIV = 0.0D0                                                    ABCD3089
      RETURN                                                           ABCD3090
3     DERIV = XS(I) * PVRDOT(I)                                        ABCD3091
      RETURN                                                           ABCD3092
      END                                                              ABCD3093
```

Subroutine DISTRB

```
      SUBROUTINE DISTRB                                                ABCD3094
      IMPLICIT  REAL*8  (A-H,O-Z)                                      ABCD3095
      COMMON /COMALL/  COM(1062)                                       ABCD3096
      DOUBLE PRECISION  LWV, MWV                                       ABCD3097
      DIMENSION  XVAR(23), PVRDOT(23), XS(23)                          ABCD3098
      DIMENSION  XSAVE(1062), AINV(529), ARINV(529), A(23,23),         ABCD3099
     1 A21INV(23,23), A22INV(23,23), AR(23,23), CINT(23,23), X(23,23), ABCD3100
     2 A12INV(23,23), DINT(23,23), BINT(50,23), CA(50,23), C(50,23),   ABCD3101
     3 CR(50,23), UVAR(50,23), AIBM(23,5), B(23,5), BR(23,5), Y(50,5),  ABCD3102
     4 D(50,5), DR(50,5), XSS(23), YI(23), USS(50), DU(5), BS(5),      ABCD3103
     5 PVRDOS(23)                                                      ABCD3104
      DIMENSION  LWV(23), MWV(23), IVARB(23), NUCOM(50)                ABCD3105
      EQUIVALENCE  (IDES, COM(3)), (JDES, COM(4)), (MODE, COM(6)),     ABCD3106
     1 (INIT, COM(7)), (PS55, COM(110)), (WAFC, COM(135)),             ABCD3107
     2 (WFB, COM(192)), (A8, COM(346)), (XVAR(1), COM(372)),           ABCD3108
     3 (PVRDOT(1), COM(998)), (XS(1), COM(1021)), (ITRAN, COM(1049)),  ABCD3109
     4 (JTRAN, COM(1050)), (IVRDOT, COM(1052)), (IDOT, COM(1053)),     ABCD3110
     5 (IAMTRX, COM(1054)), (WAFCDS, COM(1062))                        ABCD3111
C                                                                      ABCD3112
C     INV        THE TOTAL NUMBER OF STATES POSSIBLE FOR THE PARTICULAR ABCD3113
C                ENGINE BEING MODELLED, LESS THAN OR EQUAL TO 23       ABCD3114
C                                                                      ABCD3115
      DATA  INV / 16 /                                                 ABCD3116
C                                                                      ABCD3117
C     INVRED     THE TOTAL NUMBER OF STATES ACTUALLY USED IN THE MODEL ABCD3118
C                (FULL OR REDUCED), LESS THAN OR EQUAL TO INV          ABCD3119
C                                                                      ABCD3120
      DATA  INVRED / 9 /                                               ABCD3121
C                                                                      ABCD3122
C     IVARB      THE ORDER IN WHICH THE STATES ARE TO BE DONE.  IF THIS ABCD3123
C                IS A REDUCED MODEL, THE STATES TO BE INCLUDED ARE     ABCD3124
C                LISTED BEFORE THOSE NOT INCLUDED.                     ABCD3125
C                                                                      ABCD3126
```

```
      DATA  IVARB / 1, 3, 21, 18, 19, 9, 8, 11, 20, 5, 13, 17, 12, 10,   ABCD3127
     1 4, 16, 7 * 0 /                                                    ABCD3128
C                                                                        ABCD3129
C     INB        THE NUMBER OF CONTROL INPUTS, LESS THAN OR EQUAL TO 5   ABCD3130
C                                                                        ABCD3131
      DATA  INB / 2 /                                                    ABCD3132
C                                                                        ABCD3133
C     BPER       A PERCENT. THE STEADY STATE VALUE OF EACH INPUT WILL BE ABCD3134
C                MULTIPLIED BY THIS PERCENT TO FORM THE DELTA U OF EACH  ABCD3135
C                INPUT IN ORDER TO CALCULATE THE B AND D MATRICES        ABCD3136
C                                                                        ABCD3137
      DATA  BPER / .01D0 /                                               ABCD3138
C                                                                        ABCD3139
C     INC        THE NUMBER OF OUTPUTS, LESS THAN OR EQUAL TO 50         ABCD3140
C                                                                        ABCD3141
      DATA  INC / 3 /                                                    ABCD3142
C                                                                        ABCD3143
C     NUCOM      A LIST OF THE EQUIVALENCED "COM" NUMBERS OF THE OUTPUTS ABCD3144
C                DESIRED.  ALL OF THE EQUIVALENCES WILL BE FOUND IN THE  ABCD3145
C                MAIN ROUTINE, DYGABCD                                   ABCD3146
C                                                                        ABCD3147
      DATA  NUCOM / 248, 257, 258, 47 * 0 /                              ABCD3148
C                                                                        ABCD3149
C                                                                        ABCD3150
C                                                                        ABCD3151
C                                                                        ABCD3152
C     IF THE A AND C MATRIX CALCS ARE FINISHED, GO ON TO THE B AND D     ABCD3153
C     MATRIX CALCS                                                       ABCD3154
C                                                                        ABCD3155
3     IF (IB .EQ. 977)  GO TO 4                                          ABCD3156
C                                                                        ABCD3157
C     IF THIS IS NOT THE FIRST TIME THROUGH, GO TO 7                     ABCD3158
C                                                                        ABCD3159
      IF (IDOT .GT. 0)  GO TO 7                                          ABCD3160
      IGIN = 0                                                           ABCD3161
      IBCK = 0                                                           ABCD3162
      ABCK = 1.0D-5                                                      ABCD3163
C                                                                        ABCD3164
C     IF THE PROGRAM IS TO CALCULATE THE PVRDOT'S, ICHOIC=1              ABCD3165
C     IF THE PROGRAM IS TO SET THE PVRDOT'S, ICHOIC=0                    ABCD3166
C                                                                        ABCD3167
      ICHOIC = 1                                                         ABCD3168
C                                                                        ABCD3169
C     PVRDOT     A VECTOR OF (INITIAL CONDITION) PERCENTS.  THE STEADY   ABCD3170
C                STATE VALUE OF EACH STATE WILL BE MULTIPLIED BY THE     ABCD3171
C                PERCENT ASSOCIATED WITH THAT STATE TO FORM THE DELTA    ABCD3172
C                XDOT FOR THAT PARTICULAR STATE IN ORDER TO CALCULATE    ABCD3173
C                THE A MATRIX                                            ABCD3174
C                                                                        ABCD3175
C     CORE ROTOR SPEED                                                   ABCD3176
C                                                                        ABCD3177
      PVRDOT(1) = .02D0                                                  ABCD3178
C                                                                        ABCD3179
C     INTERMEDIATE ROTOR SPEED                                           ABCD3180
```

```
C                                                                       ABCD3181
      PVRDOT(2) = 0.0D0                                                 ABCD3182
C                                                                       ABCD3183
C     FAN ROTOR SPEED                                                   ABCD3184
C                                                                       ABCD3185
      PVRDOT(3) = .02D0                                                 ABCD3186
C                                                                       ABCD3187
C     FAN EXIT PRESSURE                                                 ABCD3188
C                                                                       ABCD3189
      PVRDOT(4) = .02D0                                                 ABCD3190
C                                                                       ABCD3191
C     FAN EXIT INTERNAL ENERGY                                          ABCD3192
C                                                                       ABCD3193
      PVRDOT(5) = .02D0                                                 ABCD3194
C                                                                       ABCD3195
C     INTERMEDIATE COMPRESSOR EXIT PRESSURE                             ABCD3196
C                                                                       ABCD3197
      PVRDOT(6) = 0.0D0                                                 ABCD3198
C                                                                       ABCD3199
C     INTERMEDIATE COMPRESSOR EXIT INTERNAL ENERGY                      ABCD3200
C                                                                       ABCD3201
      PVRDOT(7) = 0.0D0                                                 ABCD3202
C                                                                       ABCD3203
C     COMPRESSOR EXIT PRESSURE                                          ABCD3204
C                                                                       ABCD3205
      PVRDOT(8) = .02D0                                                 ABCD3206
C                                                                       ABCD3207
C     COMPRESSOR EXIT INTERNAL ENERGY                                   ABCD3208
C                                                                       ABCD3209
      PVRDOT(9) = .02D0                                                 ABCD3210
C                                                                       ABCD3211
C     COMBUSTOR EXIT PRESSURE                                           ABCD3212
C                                                                       ABCD3213
      PVRDOT(10) = .02D0                                                ABCD3214
C                                                                       ABCD3215
C     COMBUSTOR EXIT INTERNAL ENERGY                                    ABCD3216
C                                                                       ABCD3217
      PVRDOT(11) = .02D0                                                ABCD3218
C                                                                       ABCD3219
C     HIGH PRESSURE TURBINE EXIT PRESSURE                               ABCD3220
C                                                                       ABCD3221
      PVRDOT(12) = .02D0                                                ABCD3222
C                                                                       ABCD3223
C     HIGH PRESSURE TURBINE EXIT INTERNAL ENERGY                        ABCD3224
C                                                                       ABCD3225
      PVRDOT(13) = .02D0                                                ABCD3226
C                                                                       ABCD3227
C     INTERMEDIATE PRESSURE TURBINE EXIT PRESSURE                       ABCD3228
C                                                                       ABCD3229
      PVRDOT(14) = 0.0D0                                                ABCD3230
C                                                                       ABCD3231
C     INTERMEDIATE PRESSURE TURBINE EXIT INTERNAL ENERGY                ABCD3232
C                                                                       ABCD3233
      PVRDOT(15) = 0.0D0                                                ABCD3234
```

```
C                                                                       ABCD3235
C     LOW PRESSURE TURBINE EXIT PRESSURE                                ABCD3236
C                                                                       ABCD3237
      PVRDOT(16) = .02D0                                                ABCD3238
C                                                                       ABCD3239
C     LOW PRESSURE TURBINE EXIT INTERNAL ENERGY                         ABCD3240
C                                                                       ABCD3241
      PVRDOT(17) = .02D0                                                ABCD3242
C                                                                       ABCD3243
C     AFTERBURNER EXIT PRESSURE                                         ABCD3244
C                                                                       ABCD3245
      PVRDOT(18) = .02D0                                                ABCD3246
C                                                                       ABCD3247
C     AFTERBURNER EXIT INTERNAL ENERGY                                  ABCD3248
C                                                                       ABCD3249
      PVRDOT(19) = .02D0                                                ABCD3250
C                                                                       ABCD3251
C     DUCT BURNER EXIT PRESSURE                                         ABCD3252
C                                                                       ABCD3253
      PVRDOT(20) = .02D0                                                ABCD3254
C                                                                       ABCD3255
C     DUCT BURNER EXIT INTERNAL ENERGY                                  ABCD3256
C                                                                       ABCD3257
      PVRDOT(21) = .02D0                                                ABCD3258
C                                                                       ABCD3259
C     THIRD STREAM EXIT PRESSURE                                        ABCD3260
C                                                                       ABCD3261
      PVRDOT(22) = 0.0D0                                                ABCD3262
C                                                                       ABCD3263
C     THIRD STREAM EXIT INTERNAL ENERGY                                 ABCD3264
C                                                                       ABCD3265
      PVRDOT(23) = 0.0D0                                                ABCD3266
7     CONTINUE                                                          ABCD3267
      IDOT = IDOT + 1                                                   ABCD3268
      IF (IDOT .GT. INV)  GO TO 10                                      ABCD3269
      IVRDOT = IVARB(IDOT)                                              ABCD3270
      IF (IDOT .NE. 1)  GO TO 10                                        ABCD3271
C                                                                       ABCD3272
C     SAVE STEADY STATE VALUES AND INITIALIZE                           ABCD3273
C     FOR A AND C MATRIX CALCS                                          ABCD3274
C                                                                       ABCD3275
      DO 5  I = 1,INV                                                   ABCD3276
      J = IVARB(I)                                                      ABCD3277
      XS(J) = XVAR(J)                                                   ABCD3278
      XSS(I) = XVAR(J)                                                  ABCD3279
      IF (XSS(I) .EQ. 0.0D0)  WRITE (6,50)                              ABCD3280
5     CONTINUE                                                          ABCD3281
      DO 96  I = 1,INC                                                  ABCD3282
      IU = NUCOM(I)                                                     ABCD3283
96    USS(I) = COM(IU)                                                  ABCD3284
      DO 1  I = 1,1062                                                  ABCD3285
1     XSAVE(I) = COM(I)                                                 ABCD3286
      WRITE (6,65)                                                      ABCD3287
      WRITE (6,44)  (XSS(I),  I = 1,INV)                                ABCD3288
```

```
      RETURN                                                                ABCD3289
10    IDOT1 = IDOT - 1                                                      ABCD3290
C                                                                           ABCD3291
C     GENERATE MATRIX FOR A MATRIX CALCS                                    ABCD3292
C                                                                           ABCD3293
      DO 15  I = 1,INV                                                      ABCD3294
      J = IVARB(I)                                                          ABCD3295
15    X(I,IDOT1) = XVAR(J) - XSS(I)                                         ABCD3296
      J = IVARB(IDOT1)                                                      ABCD3297
      IF (ICHOIC .EQ. 0)  GO TO 23                                          ABCD3298
      ERRX = DABS(X(IDOT1,IDOT1) / XSS(IDOT1)) - .002D0                     ABCD3299
      IF (DABS(ERRX) .LE. .00001D0)  GO TO 17                               ABCD3300
      IDOT = IDOT1                                                          ABCD3301
      IVRDOT = IVARB(IDOT)                                                  ABCD3302
      IGIN = IGIN + 1                                                       ABCD3303
      IF (DABS(X(1,IDOT1)) .NE. 0.0D0)  GO TO 18                            ABCD3304
      PVRP = PVRDOT(J)                                                      ABCD3305
      PVRDOT(J) = 2.0D0 * PVRDOT(J)                                         ABCD3306
      IF (ERRX .EQ. ERRXP .OR. IBCK .NE. 0)  GO TO 9                        ABCD3307
      PVRDOT(J) = 1.5D0 * PVRP + ABCK                                       ABCD3308
      ABCK = 2.0D0 * ABCK                                                   ABCD3309
9     ERRXP = ERRX                                                          ABCD3310
      IBCK = 1                                                              ABCD3311
      GO TO 23                                                              ABCD3312
18    IF (IGIN .NE. 1)  GO TO 16                                            ABCD3313
      IBCK = 0                                                              ABCD3314
      PVRP = PVRDOT(J)                                                      ABCD3315
      ERRXP = ERRX                                                          ABCD3316
      PVRDOT(J) = 1.05D0 * PVRDOT(J)                                        ABCD3317
      GO TO 23                                                              ABCD3318
16    PVRNEW = (PVRDOT(J) * ERRXP - PVRP * ERRX) / (ERRXP - ERRX)           ABCD3319
      IBCK = 0                                                              ABCD3320
      IF (PVRNEW .LE. 0.0D0)  PVRNEW = .95D0 * PVRDOT(J)                    ABCD3321
      IF (IGIN .EQ. 50)  PVRNEW = ((PVRDOT(J) + PVRP) / 2.0D0) + .1D-3      ABCD3322
      IF (IGIN .EQ. 100)  PVRNEW = ((PVRDOT(J) + PVRP) / 2.0D0) + .1D-2     ABCD3323
      PVRP = PVRDOT(J)                                                      ABCD3324
      ERRXP = ERRX                                                          ABCD3325
      PVRDOT(J) = PVRNEW                                                    ABCD3326
      GO TO 23                                                              ABCD3327
C                                                                           ABCD3328
C     GENERATE MATRIX FOR C MATRIX CALCS                                    ABCD3329
C                                                                           ABCD3330
17    IGIN = 0                                                              ABCD3331
      IBCK = 0                                                              ABCD3332
      ABCK = 1.0D-5                                                         ABCD3333
      IF (PVRDOT(J) .LE. 0.0D0)  PVRDOT(J) = ABCK                           ABCD3334
23    DO 97  I = 1,INC                                                      ABCD3335
      IU = NUCOM(I)                                                         ABCD3336
97    UVAR(I,IDOT1) = COM(IU) - USS(I)                                      ABCD3337
      WRITE (6,45)  IDOT1,IVARB(IDOT1)                                      ABCD3338
      WRITE (6,60)                                                          ABCD3339
      WRITE (6,44)  (X(I,IDOT1),  I = 1,INV)                                ABCD3340
      WRITE (6,282)  IGIN                                                   ABCD3341
      WRITE (6,296)  PVRDOT(J)                                              ABCD3342
```

```
      IAMTSV = IAMTRX                                                   ABCD3343
      IVRSV = IVRDOT                                                    ABCD3344
      IDOTSV = IDOT                                                     ABCD3345
      DO 19  I = 1,23                                                   ABCD3346
19    PVRDOS(I) = PVRDOT(I)                                             ABCD3347
C                                                                       ABCD3348
C     RESET STEADY STATE VALUES                                         ABCD3349
C                                                                       ABCD3350
      DO 21  I = 1,1062                                                 ABCD3351
21    COM(I) = XSAVE(I)                                                 ABCD3352
      IAMTRX = IAMTSV                                                   ABCD3353
      IVRDOT = IVRSV                                                    ABCD3354
      IDOT = IDOTSV                                                     ABCD3355
      DO 22  I = 1,23                                                   ABCD3356
22    PVRDOT(I) = PVRDOS(I)                                             ABCD3357
      IF (IGIN .NE. 0)  RETURN                                          ABCD3358
      IF (IDOT .LE. INV)  RETURN                                        ABCD3359
C                                                                       ABCD3360
C     FINAL CALCS FOR A AND C MATRICES                                  ABCD3361
C                                                                       ABCD3362
25    DO 24  I = 1,INV                                                  ABCD3363
      J = IVARB(I)                                                      ABCD3364
24    YI(I) = 1.0D0 / XSS(I) / PVRDOT(J)                                ABCD3365
      WRITE (6,55)                                                      ABCD3366
      WRITE (6,44)  (YI(I),  I = 1,INV)                                 ABCD3367
      WRITE (6,75)                                                      ABCD3368
      DO 31  J = 1,INV                                                  ABCD3369
      DO 30  I = 1,INV                                                  ABCD3370
      K = INV * (J-1) + I                                               ABCD3371
30    AINV(K) = YI(J) * X(I,J)                                          ABCD3372
      LL = K - INV + 1                                                  ABCD3373
      WRITE (6,44)  (AINV(I),  I = LL,K)                                ABCD3374
31    CONTINUE                                                          ABCD3375
      DO 46  J = 1,INVRED                                               ABCD3376
      DO 46  I = 1,INVRED                                               ABCD3377
      K = INVRED * (J - 1) + I                                          ABCD3378
46    ARINV(K) = YI(J) * X(I,J)                                         ABCD3379
      L = INVRED + 1                                                    ABCD3380
      IF (L .GT. INV)  GO TO 49                                         ABCD3381
C                                                                       ABCD3382
C     A12INV, A21INV, AND A22INV ARE NECESSARY FOR C AND D MATRIX       ABCD3383
C     CALCS IN REDUCED FORM                                             ABCD3384
C                                                                       ABCD3385
      DO 47  J = 1,INVRED                                               ABCD3386
      DO 47  I = L,INV                                                  ABCD3387
      A12INV(J,I) = YI(I) * X(J,I)                                      ABCD3388
47    A21INV(I,J) = YI(J) * X(I,J)                                      ABCD3389
      DO 48  J = L,INV                                                  ABCD3390
      DO 48  I = L,INV                                                  ABCD3391
48    A22INV(I,J) = YI(J) * X(I,J)                                      ABCD3392
49    CONTINUE                                                          ABCD3393
      DO 98  J = 1,INV                                                  ABCD3394
      DO 98  I = 1,INC                                                  ABCD3395
98    CA(I,J) = YI(J) * UVAR(I,J)                                       ABCD3396
```

```
      CALL DMINV (AINV, INV, DET, LWV, MWV)                          ABCD3397
      IF (DET .EQ. 0.0D0)  WRITE (6,70)                              ABCD3398
      DO 32  J = 1,INV                                               ABCD3399
      DO 32  I = 1,INV                                               ABCD3400
      K = INV * (J-1) + I                                            ABCD3401
32    A(I,J) = AINV(K)                                               ABCD3402
      WRITE (6,80)                                                   ABCD3403
C                                                                    ABCD3404
C     THE FOLLOWING STATEMENT IS USED AT LEWIS ONLY                  ABCD3405
C     WRITE (10)  A                                                  ABCD3406
C                                                                    ABCD3407
      DO 100  K = 1,INV                                              ABCD3408
      DO 100  I = 1,INC                                              ABCD3409
      SUM = 0.0D0                                                    ABCD3410
      DO 99  J = 1,INV                                               ABCD3411
99    SUM = SUM + CA(I,J) * A(J,K)                                   ABCD3412
100   C(I,K) = SUM                                                   ABCD3413
C                                                                    ABCD3414
C     THE FOLLOWING STATEMENT IS USED AT LEWIS ONLY                  ABCD3415
C     WRITE (20)  C                                                  ABCD3416
C                                                                    ABCD3417
      DO 33  J = 1,INV                                               ABCD3418
      WRITE (6,44)  (A(J,I),  I = 1,INV)                             ABCD3419
33    CONTINUE                                                       ABCD3420
      WRITE (6,81)                                                   ABCD3421
      DO 101  J = 1,INC                                              ABCD3422
      WRITE (6,44)  (C(J,I),  I = 1,INV)                             ABCD3423
101   CONTINUE                                                       ABCD3424
C                                                                    ABCD3425
C     RESET STEADY STATE VALUES AND INITIALIZE                       ABCD3426
C     FOR B AND D MATRIX CALCS                                       ABCD3427
C                                                                    ABCD3428
      IB = 977                                                       ABCD3429
      JDOT = 0                                                       ABCD3430
      DO 26  I = 1,1062                                              ABCD3431
26    COM(I) = XSAVE(I)                                              ABCD3432
      DO 76  I = 1,INC                                               ABCD3433
      IU = NUCOM(I)                                                  ABCD3434
76    USS(I) = COM(IU)                                               ABCD3435
      DO 34  I = 1,INV                                               ABCD3436
34    PVRDOT(I) = 0.0D0                                              ABCD3437
      BS(1) = WFB                                                    ABCD3438
      BS(2) = A8                                                     ABCD3439
      BS(3) = X3                                                     ABCD3440
      BS(4) = X4                                                     ABCD3441
      BS(5) = X5                                                     ABCD3442
      DO 27  I = 1,5                                                 ABCD3443
27    DU(I) = BPER * BS(I)                                           ABCD3444
4     IF (JDOT .EQ. 1)  GO TO 35                                     ABCD3445
      DO 2  I = 1,INV                                                ABCD3446
      J = IVARB(I)                                                   ABCD3447
2     XSS(I) = XVAR(J)                                               ABCD3448
      DO 6  I = 1,1062                                               ABCD3449
6     XSAVE(I) = COM(I)                                              ABCD3450
```

```
      IAMTRX = 1                                                        ABCD3451
      ITRAN = 0                                                         ABCD3452
      IDOT = 0                                                          ABCD3453
      IDES = 0                                                          ABCD3454
      MODE = 2                                                          ABCD3455
      JDOT = 1                                                          ABCD3456
      WFB = WFB + DU(1)                                                 ABCD3457
      RETURN                                                            ABCD3458
35    IDOT = IDOT + 1                                                   ABCD3459
C                                                                       ABCD3460
C     GENERATE MATRIX FOR B MATRIX CALCS                                ABCD3461
C                                                                       ABCD3462
      DO 36  I = 1,INV                                                  ABCD3463
      J = IVARB(I)                                                      ABCD3464
36    AIBM(I,IDOT) = (XVAR(J) - XSS(I)) / DU(IDOT)                      ABCD3465
C                                                                       ABCD3466
C     GENERATE MATRIX FOR D MATRIX CALCS                                ABCD3467
C                                                                       ABCD3468
      DO 77  I = 1,INC                                                  ABCD3469
      IU = NUCOM(I)                                                     ABCD3470
77    Y(I,IDOT) = (COM(IU) - USS(I)) / DU(IDOT)                         ABCD3471
C                                                                       ABCD3472
C     RESET STEADY STATE VALUES                                         ABCD3473
C                                                                       ABCD3474
      IAMTSV = IAMTRX                                                   ABCD3475
      MODESV = MODE                                                     ABCD3476
      IDESV = IDES                                                      ABCD3477
      ITRASV = ITRAN                                                    ABCD3478
      IDOTSV = IDOT                                                     ABCD3479
      DO 29  I = 1,1062                                                 ABCD3480
29    COM(I) = XSAVE(I)                                                 ABCD3481
      IAMTRX = IAMTSV                                                   ABCD3482
      MODE = MODESV                                                     ABCD3483
      IDES = IDESV                                                      ABCD3484
      ITRAN = ITRASV                                                    ABCD3485
      IDOT = IDOTSV                                                     ABCD3486
8     IF (IDOT .GE. INB)  GO TO 28                                      ABCD3487
      GO TO (11,12,13,14), IDOT                                         ABCD3488
11    A8 = A8 + DU(2)                                                   ABCD3489
      RETURN                                                            ABCD3490
12    X3 = X3 + DU(3)                                                   ABCD3491
      RETURN                                                            ABCD3492
13    X4 = X4 + DU(4)                                                   ABCD3493
      RETURN                                                            ABCD3494
14    X5 = X5 + DU(5)                                                   ABCD3495
      RETURN                                                            ABCD3496
C                                                                       ABCD3497
C     FINAL CALCS FOR B MATRIX                                          ABCD3498
C                                                                       ABCD3499
28    DO 38  K = 1,INB                                                  ABCD3500
      DO 38  I = 1,INV                                                  ABCD3501
      SUM = 0.0D0                                                       ABCD3502
      DO 37  J = 1,INV                                                  ABCD3503
37    SUM = SUM - (A(I,J) * AIBM(J,K))                                  ABCD3504
```

```
38    B(I,K) = SUM                                                          ABCD3505
C                                                                           ABCD3506
C     THE FOLLOWING STATEMENT IS USED AT LEWIS ONLY                         ABCD3507
C     WRITE (15)  B                                                         ABCD3508
C                                                                           ABCD3509
C                                                                           ABCD3510
C     FINAL CALCS FOR D MATRIX                                              ABCD3511
C                                                                           ABCD3512
      DO 87  K = 1,INB                                                      ABCD3513
      DO 87  I = 1,INC                                                      ABCD3514
      SUM = 0.0D0                                                           ABCD3515
      DO 86  J = 1,INV                                                      ABCD3516
86    SUM = SUM + C(I,J) * AIBM(J,K)                                        ABCD3517
87    D(I,K) = Y(I,K) - SUM                                                 ABCD3518
C                                                                           ABCD3519
C     THE FOLLOWING STATEMENT IS USED AT LEWIS ONLY                         ABCD3520
C     WRITE (25)  D                                                         ABCD3521
C                                                                           ABCD3522
      WRITE (6,85)                                                          ABCD3523
      GO TO (39,40,41,42,43), INB                                           ABCD3524
39    WRITE (6,91)  ((B(I,J),  J = 1,INB),  I = 1,INV)                      ABCD3525
      GO TO 110                                                             ABCD3526
40    WRITE (6,92)  ((B(I,J),  J = 1,INB),  I = 1,INV)                      ABCD3527
      GO TO 110                                                             ABCD3528
41    WRITE (6,93)  ((B(I,J),  J = 1,INB),  I = 1,INV)                      ABCD3529
      GO TO 110                                                             ABCD3530
42    WRITE (6,94)  ((B(I,J),  J = 1,INB),  I = 1,INV)                      ABCD3531
      GO TO 110                                                             ABCD3532
43    WRITE (6,95)  ((B(I,J),  J = 1,INB),  I = 1,INV)                      ABCD3533
110   WRITE (6,82)                                                          ABCD3534
      GO TO (111,112,113,114,115),  INB                                     ABCD3535
111   WRITE (6,91)  ((D(I,J),  J = 1,INB),  I = 1,INC)                      ABCD3536
      GO TO 200                                                             ABCD3537
112   WRITE (6,92)  ((D(I,J),  J = 1,INB),  I = 1,INC)                      ABCD3538
      GO TO 200                                                             ABCD3539
113   WRITE (6,93)  ((D(I,J),  J = 1,INB),  I = 1,INC)                      ABCD3540
      GO TO 200                                                             ABCD3541
114   WRITE (6,94)  ((D(I,J),  J = 1,INB),  I = 1,INC)                      ABCD3542
      GO TO 200                                                             ABCD3543
115   WRITE (6,95)  ((D(I,J),  J = 1,INB),  I = 1,INC)                      ABCD3544
200   IF (L .GT. INV)  STOP                                                 ABCD3545
C                                                                           ABCD3546
C     CALCULATIONS FOR ALL MATRICES OF REDUCED ORDER MODEL                  ABCD3547
C                                                                           ABCD3548
C     FINAL CALCS FOR REDUCED A MATRIX                                      ABCD3549
C                                                                           ABCD3550
      CALL DMINV (ARINV, INVRED, DET, LWV, MWV)                             ABCD3551
      DO 201  J = 1,INVRED                                                  ABCD3552
      DO 201  I = 1,INVRED                                                  ABCD3553
      K = INVRED * (J - 1) + I                                              ABCD3554
201   AR(I,J) = ARINV(K)                                                    ABCD3555
C                                                                           ABCD3556
C     FINAL CALCS FOR REDUCED B MATRIX                                      ABCD3557
C                                                                           ABCD3558
```

```
      DO 203  K = 1,INB                                                 ABCD3559
      DO 203  I = 1,INVRED                                              ABCD3560
      SUM = 0.0D0                                                       ABCD3561
      DO 202  J = 1,INVRED                                              ABCD3562
202   SUM = SUM - AR(I,J) * AIBM(J,K)                                   ABCD3563
203   BR(I,K) = SUM                                                     ABCD3564
C                                                                       ABCD3565
C     FINAL CALCS FOR REDUCED C MATRIX                                  ABCD3566
C                                                                       ABCD3567
      DO 205  K = 1,INVRED                                              ABCD3568
      DO 205  I = L,INV                                                 ABCD3569
      SUM = 0.0D0                                                       ABCD3570
      DO 204  J = 1,INVRED                                              ABCD3571
204   SUM = SUM + A21INV(I,J) * AR(J,K)                                 ABCD3572
205   CINT(I,K) = SUM                                                   ABCD3573
      DO 207  K = 1,INVRED                                              ABCD3574
      DO 207  I = 1,INC                                                 ABCD3575
      SUM = 0.0D0                                                       ABCD3576
      DO 206  J = L,INV                                                 ABCD3577
206   SUM = SUM + C(I,J) * CINT(J,K)                                    ABCD3578
207   CR(I,K) = C(I,K) + SUM                                            ABCD3579
C                                                                       ABCD3580
C     FINAL CALCS FOR REDUCED D MATRIX                                  ABCD3581
C                                                                       ABCD3582
      DO 209  K = L,INV                                                 ABCD3583
      DO 209  I = 1,INVRED                                              ABCD3584
      SUM = 0.0D0                                                       ABCD3585
      DO 208  J = 1,INVRED                                              ABCD3586
208   SUM = SUM + AR(I,J) * A12INV(J,K)                                 ABCD3587
209   CINT(I,K) = SUM                                                   ABCD3588
      DO 211  K = L,INV                                                 ABCD3589
      DO 211  I = L,INV                                                 ABCD3590
      SUM = 0.0D0                                                       ABCD3591
      DO 210  J = 1,INVRED                                              ABCD3592
210   SUM = SUM + A21INV(I,J) * CINT(J,K)                               ABCD3593
211   DINT(I,K) = SUM - A22INV(I,K)                                     ABCD3594
      DO 213  K = L,INV                                                 ABCD3595
      DO 213  I = 1,INC                                                 ABCD3596
      SUM = 0.0D0                                                       ABCD3597
      DO 212  J = L,INV                                                 ABCD3598
212   SUM = SUM + C(I,J) * DINT(J,K)                                    ABCD3599
213   BINT(I,K) = SUM                                                   ABCD3600
      DO 103  K = 1,INB                                                 ABCD3601
      DO 103  I = 1,INC                                                 ABCD3602
      SUM = 0.0D0                                                       ABCD3603
      DO 102  J = L,INV                                                 ABCD3604
102   SUM = SUM + BINT(I,J) * B(J,K)                                    ABCD3605
103   DR(I,K) = SUM + D(I,K)                                            ABCD3606
      WRITE (6,214)                                                     ABCD3607
      DO 218  I = 1,INVRED                                              ABCD3608
      WRITE (6,44)  (AR(I,J),  J = 1,INVRED)                            ABCD3609
218   CONTINUE                                                          ABCD3610
      WRITE (6,215)                                                     ABCD3611
      GO TO (219,220,221,222,223),  INB                                 ABCD3612
```

```
219   WRITE (6,91)   ((BR(I,J),  J = 1,INB),  I = 1,INVRED)                ABCD3613
      GO TO 224                                                            ABCD3614
220   WRITE (6,92)   ((BR(I,J),  J = 1,INB),  I = 1,INVRED)                ABCD3615
      GO TO 224                                                            ABCD3616
221   WRITE (6,93)   ((BR(I,J),  J = 1,INB),  I = 1,INVRED)                ABCD3617
      GO TO 224                                                            ABCD3618
222   WRITE (6,94)   ((BR(I,J),  J = 1,INB),  I = 1,INVRED)                ABCD3619
      GO TO 224                                                            ABCD3620
223   WRITE (6,95)   ((BR(I,J),  J = 1,INB),  I = 1,INVRED)                ABCD3621
224   WRITE (6,216)                                                        ABCD3622
      DO 225  J = 1,INC                                                    ABCD3623
      WRITE (6,44)   (CR(J,I),  I = 1,INVRED)                              ABCD3624
225   CONTINUE                                                             ABCD3625
      WRITE (6,217)                                                        ABCD3626
      GO TO (226,227,228,229,230),  INB                                    ABCD3627
226   WRITE (6,91)   ((DR(I,J),  J = 1,INB),  I = 1,INC)                   ABCD3628
      STOP                                                                 ABCD3629
227   WRITE (6,92)   ((DR(I,J),  J = 1,INB),  I = 1,INC)                   ABCD3630
      STOP                                                                 ABCD3631
228   WRITE (6,93)   ((DR(I,J),  J = 1,INB),  I = 1,INC)                   ABCD3632
      STOP                                                                 ABCD3633
229   WRITE (6,94)   ((DR(I,J),  J = 1,INB),  I = 1,INC)                   ABCD3634
      STOP                                                                 ABCD3635
230   WRITE (6,95)   ((DR(I,J),  J = 1,INB),  I = 1,INC)                   ABCD3636
      STOP                                                                 ABCD3637
44    FORMAT (1P10E12.4)                                                   ABCD3638
45    FORMAT (1X, I5, 2X, ' THE VARIABLE IS NO. ', I3)                     ABCD3639
50    FORMAT (1X, ' THE S.S. VARIABLE IS 0.0 ' )                           ABCD3640
55    FORMAT (1X, ' Y INVERSE = ' )                                        ABCD3641
60    FORMAT (1X, ' THIS COLUMN OF X = ' )                                 ABCD3642
65    FORMAT (1X, ' THIS IS THE S.S. SOLUTION ' )                          ABCD3643
70    FORMAT (1X, ' AINV IS SINGULAR ' )                                   ABCD3644
75    FORMAT (1X, 7HAINV = )                                               ABCD3645
80    FORMAT (1X, 4HA = )                                                  ABCD3646
81    FORMAT (1X, 4HC = )                                                  ABCD3647
82    FORMAT (1X, 4HD = )                                                  ABCD3648
85    FORMAT (1X, 4HB = )                                                  ABCD3649
91    FORMAT (1X, 1PE12.4)                                                 ABCD3650
92    FORMAT (1X, 1P2E12.4)                                                ABCD3651
93    FORMAT (1X, 1P3E12.4)                                                ABCD3652
94    FORMAT (1X, 1P4E12.4)                                                ABCD3653
95    FORMAT (1X, 1P5E12.4)                                                ABCD3654
214   FORMAT (1X, 5HAR = )                                                 ABCD3655
215   FORMAT (1X, 5HBR = )                                                 ABCD3656
216   FORMAT (1X, 5HCR = )                                                 ABCD3657
217   FORMAT (1X, 5HDR = )                                                 ABCD3658
280   FORMAT (1H1)                                                         ABCD3659
282   FORMAT (1X, 'IGIN = ' , I5)                                          ABCD3660
296   FORMAT (1X, 'THE NEXT TRY FOR PVRDOT = ', 1PE12.4)                   ABCD3661
      END                                                                  ABCD3662
```

Subroutine ENGBAL

```
      SUBROUTINE ENGBAL                                                 ABCD3663

      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD3664
      LOGICAL  ERRER, DUMSPL, FXFN2M, FXM2CP, FAN                       ABCD3665
      COMMON /COMALL/  COM(1062)                                        ABCD3666
      DIMENSION  WORD(2), ERR(9)                                        ABCD3667
      DIMENSION  VAR(9), DEL(9), ERRB(9), DELVAR(9), EMAT(9,9), VMAT(9),ABCD3668
     1 AMAT(9), DELSAV(9), AWORD(2)                                     ABCD3669
      EQUIVALENCE  (WORD(1), COM(1)), (MODE, COM(6)), (INIT, COM(7)),   ABCD3670
     1 (ITRYS, COM(18)), (LOOPER, COM(19)), (NOMAP, COM(20)),           ABCD3671
     2 (NUMMAP, COM(21)), (MAPEDG, COM(22)), (TOLALL, COM(23)),         ABCD3672
     3 (ERR(1), COM(24)), (TFHPDS, COM(118)), (ZF, COM(136)),           ABCD3673
     4 (PCNF, COM(137)), (ZI, COM(139)), (PCNI, COM(140)),              ABCD3674
     5 (TFFIP, COM(141)), (T4, COM(156)), (TFFHP, COM(175)),            ABCD3675
     6 (TFLPDS, COM(275)), (TFIPDS, COM(278)), (TFFLP, COM(289)),       ABCD3676
     7 (ZC, COM(300)), (PCNC, COM(301)), (TIME, COM(993)),              ABCD3677
     8 (DT, COM(994)), (TF, COM(995)), (ISPOOL, COM(1044)),             ABCD3678
     9 (ITRAN, COM(1049)), (JTRAN, COM(1050)), (NSTEP, COM(1051)),      ABCD3679
     1 (IAMTRX, COM(1054)), (ERRER, COM(1056)), (DUMSPL, COM(1057)),    ABCD3680
     2 (FXFN2M, COM(1058)), (FXM2CP, COM(1059)), (FAN, COM(1061))       ABCD3681
      DATA  AWORD /4HENGB, 4HAL  /                                      ABCD3682
      DATA  VDELTA, VLIM, VCHNGE, NOMISX /1.0D-4, 0.1D0, 0.85D0, 4/     ABCD3683
      DATA  DEL /9*0.0D0/                                               ABCD3684
      DATA  DELSAV /9*1.0D-4/                                           ABCD3685
C                                                                       ABCD3686
      IF (ITRAN .NE. 1)  GO TO 100                                      ABCD3687
      CALL SYG(1)                                                       ABCD3688
      JTRAN = 1                                                         ABCD3689
      INIT = 1                                                          ABCD3690
      NSTEP = NSTEP + 1                                                 ABCD3691
      IF (IAMTRX .EQ. 1)  NSTEP = NSTEP - 1                             ABCD3692
      TIME = DT * DFLOAT(NSTEP)                                         ABCD3693
      IF (TIME .GT. TF)  GO TO 100                                      ABCD3694
      CALL DISTRB                                                       ABCD3695
      CALL COINLT                                                       ABCD3696
      GO TO 101                                                         ABCD3697
  100 IF (IAMTRX .EQ. 1)  CALL DISTRB                                   ABCD3698
      IF (IAMTRX .EQ. 1)  CALL COINLT                                   ABCD3699
      IF (IAMTRX .NE. 1)  CALL PUTIN                                    ABCD3700
  101 IF (INIT .EQ. 1)  GO TO 1                                         ABCD3701
      TFFHP = TFHPDS                                                    ABCD3702
      TFFIP = TFIPDS                                                    ABCD3703
      IF (FXM2CP)  TFFIP = TFHPDS                                       ABCD3704
      TFFLP = TFLPDS                                                    ABCD3705
      LOOPER = 0                                                        ABCD3706
      NUMMAP = 0                                                        ABCD3707
      NOMISS = 0                                                        ABCD3708
      LOOP = 0                                                          ABCD3709
      MISMAT = 0                                                        ABCD3710
      NOMAP = 0                                                         ABCD3711
      IGO = 2                                                           ABCD3712
      DO 3  I = 1,9                                                     ABCD3713
      VMAT(I) = 0.0D0                                                   ABCD3714
      AMAT(I) = 0.0D0                                                   ABCD3715
      DELVAR(I) = 0.0D0                                                 ABCD3716
      DO 3  L = 1,9                                                     ABCD3717
```

```
3       EMAT(I,L) = 0.0D0                                                  ABCD3718
4       LOOPER = LOOPER + 1                                                ABCD3719
        CALL COFAN                                                         ABCD3720
        WORD(1) = AWORD(1)                                                 ABCD3721
        WORD(2) = AWORD(2)                                                 ABCD3722
        IF (.NOT. FAN)  DUMSPL = .TRUE.                                    ABCD3723
        IF (LOOPER .LE. ITRYS)  GO TO 45                                   ABCD3724
        ERRER = .TRUE.                                                     ABCD3725
        GO TO 26                                                           ABCD3726
45      IF (NOMAP .GT. 0)  GO TO 2                                         ABCD3727
        NUMMAP = 0                                                         ABCD3728
5       VAR(1) = ZF * 100.0D0                                              ABCD3729
        VAR(2) = PCNF                                                      ABCD3730
        IF (MODE .EQ. 3)  VAR(2) = T4 / 10.0D0                             ABCD3731
        VAR(3) = ZC * 100.0D0                                              ABCD3732
        VAR(4) = PCNC                                                      ABCD3733
        IF (MODE .EQ. 1)  VAR(4) = T4 / 10.0D0                             ABCD3734
        VAR(5) = TFFHP                                                     ABCD3735
        VAR(6) = TFFLP                                                     ABCD3736
        VAR(7) = ZI * 100.0D0                                              ABCD3737
        VAR(8) = PCNI                                                      ABCD3738
        VAR(9) = TFFIP                                                     ABCD3739
        NMAX = 9                                                           ABCD3740
        IF (FAN)  GO TO 39                                                 ABCD3741
        NMAX = 6                                                           ABCD3742
        IF (ISPOOL .EQ. 2)  GO TO 7                                        ABCD3743
        NMAX = 3                                                           ABCD3744
        VAR(3) = TFFLP                                                     ABCD3745
        GO TO 7                                                            ABCD3746
39      IF (.NOT. FXFN2M .AND. (.NOT. DUMSPL))  GO TO 6                    ABCD3747
        NMAX = 7                                                           ABCD3748
        IF (DUMSPL)  NMAX = 6                                              ABCD3749
6       IF (.NOT. FXM2CP)  GO TO 7                                         ABCD3750
        NMAX = 7                                                           ABCD3751
        VAR(4) = PCNI                                                      ABCD3752
        VAR(5) = TFFIP                                                     ABCD3753
7       DO 8  I = 1,NMAX                                                   ABCD3754
        IF (DABS(ERR(I)) .GT. TOLALL)  GO TO 9                             ABCD3755
8       CONTINUE                                                           ABCD3756
        IF (ITRAN .EQ. 1)  CALL FOLL                                       ABCD3757
        CALL PERF                                                          ABCD3758
        CALL ERROR                                                         ABCD3759
9       IF (LOOP .GT. 0)  GO TO 11                                         ABCD3760
        MAPEDG = 0                                                         ABCD3761
        MAPSET = 0                                                         ABCD3762
        DO 10  I = 1,NMAX                                                  ABCD3763
        ERRB(I) = ERR(I)                                                   ABCD3764
10      DEL(I) = VDELTA * VAR(I)                                           ABCD3765
        GO TO 14                                                           ABCD3766
11      IF (MISMAT .GT. 0)  GO TO 29                                       ABCD3767
        IF (MAPEDG .EQ. 0)  GO TO 12                                       ABCD3768
        MAPEDG = 0                                                         ABCD3769
        MAPSET = 1                                                         ABCD3770
        VAR(LOOP) = VAR(LOOP) + 2.0D0 * DEL(LOOP)                          ABCD3771
```

```
      GO TO 15                                                          ABCD3772
12    IF (MAPSET .EQ. 0)  VAR(LOOP) = VAR(LOOP) + DEL(LOOP)             ABCD3773
      IF (MAPSET .EQ. 1)  VAR(LOOP) = VAR(LOOP) - DEL(LOOP)             ABCD3774
      MAPSET = 0                                                        ABCD3775
      DO 13  I = 1,NMAX                                                 ABCD3776
      IF (DEL(LOOP) .NE. 0.0D0)  DELSAV(LOOP) = DEL(LOOP)               ABCD3777
      IF (DEL(LOOP) .EQ. 0.0D0)  DEL(LOOP) = DELSAV(LOOP)               ABCD3778
13    EMAT(I,LOOP) = (ERRB(I) - ERR(I)) / DEL(LOOP)                     ABCD3779
14    LOOP = LOOP + 1                                                   ABCD3780
      IF (LOOP .GT. NMAX)  GO TO 17                                     ABCD3781
      VAR(LOOP) = VAR(LOOP) - DEL(LOOP)                                 ABCD3782
15    ZF = VAR(1) / 100.0D0                                             ABCD3783
      IF (MODE .NE. 3)  PCNF = VAR(2)                                   ABCD3784
      IF (MODE .EQ. 3)  T4 = VAR(2) * 10.0D0                            ABCD3785
      ZC = VAR(3) / 100.0D0                                             ABCD3786
      IF (MODE .NE. 1)  PCNC = VAR(4)                                   ABCD3787
      IF (MODE .EQ. 1)  T4 = VAR(4) * 10.0D0                            ABCD3788
      TFFHP = VAR(5)                                                    ABCD3789
      TFFLP = VAR(6)                                                    ABCD3790
      ZI = VAR(7) / 100.0D0                                             ABCD3791
      PCNI = VAR(8)                                                     ABCD3792
      TFFIP = VAR(9)                                                    ABCD3793
      IF (.NOT. FXM2CP)  GO TO 16                                       ABCD3794
      PCNI = VAR(4)                                                     ABCD3795
      TFFIP = VAR(5)                                                    ABCD3796
16    IF (ISPOOL .EQ. 1)  TFFLP = VAR(3)                                ABCD3797
      IF (ZI .LT. 0.0D0)  ZI = 0.05D0                                   ABCD3798
      IF (ZF .LT. 0.0D0)  ZF = 0.05D0                                   ABCD3799
      IF (ZC .LT. 0.0D0)  ZC = 0.05D0                                   ABCD3800
      GO TO (2,4), IGO                                                  ABCD3801
17    DO 18  I = 1,NMAX                                                 ABCD3802
18    AMAT(I) = - ERRB(I)                                               ABCD3803
      DO 20  I = 1,NMAX                                                 ABCD3804
      IZERO = 0                                                         ABCD3805
      DO 19  LOOP = 1,NMAX                                              ABCD3806
      IF (EMAT(I,LOOP) .EQ. 0.0D0)  IZERO = IZERO + 1                   ABCD3807
19    CONTINUE                                                          ABCD3808
      IF (IZERO .LT. NMAX)  GO TO 20                                    ABCD3809
      WRITE (6,32) I                                                    ABCD3810
      LOOPER = ITRYS + 100                                              ABCD3811
      GO TO 26                                                          ABCD3812
20    CONTINUE                                                          ABCD3813
      DO 22  LOOP = 1,NMAX                                              ABCD3814
      IZERO = 0                                                         ABCD3815
      DO 21  I = 1,NMAX                                                 ABCD3816
21    IF (EMAT(I,LOOP) .EQ. 0.0D0)  IZERO = IZERO + 1                   ABCD3817
      IF (IZERO .LT. NMAX)  GO TO 22                                    ABCD3818
      WRITE (6,33) LOOP                                                 ABCD3819
      LOOPER = ITRYS + 100                                              ABCD3820
      GO TO 26                                                          ABCD3821
22    CONTINUE                                                          ABCD3822
23    CALL MATRIX (EMAT,VMAT,AMAT,NMAX)                                 ABCD3823
      LBIG = 0                                                          ABCD3824
      VARBIG = 0.0D0                                                    ABCD3825
```

```
      DO 24  L = 1,NMAX                                                  ABCD3826
      ABSVAR = DABS(VMAT(L))                                             ABCD3827
      IF (ABSVAR .LE. VLIM * VAR(L))  GO TO 24                           ABCD3828
      IF (ABSVAR .LE. VARBIG)  GO TO 24                                  ABCD3829
      LBIG = L                                                           ABCD3830
      VARBIG = ABSVAR                                                    ABCD3831
24    CONTINUE                                                           ABCD3832
      VRATIO = 1.0D0                                                     ABCD3833
      IF (LBIG .GT. 0)  VRATIO = VLIM * VAR(LBIG) / VARBIG               ABCD3834
      ERRAVE = 0.0D0                                                     ABCD3835
      VMTAVE = 0.0D0                                                     ABCD3836
      DELAVE = 0.0D0                                                     ABCD3837
      FNMAX = NMAX                                                       ABCD3838
      DO 25  L = 1,NMAX                                                  ABCD3839
      DELVAR(L) = VRATIO * VMAT(L)                                       ABCD3840
      ERRAVE = ERRAVE + DABS(AMAT(L)) / FNMAX                            ABCD3841
      VAR(L) = VAR(L) + DELVAR(L)                                        ABCD3842
      VMTAVE = VMTAVE + DABS(VMAT(L)) / FNMAX                            ABCD3843
25    DELAVE = DELAVE + DABS(DELVAR(L)) / FNMAX                          ABCD3844
      IF (MISMAT .GT. 0)  GO TO 31                                       ABCD3845
      IF (NOMISS .EQ. 0)  MISMAT = 1                                     ABCD3846
      IF (MISMAT .EQ. 0)  IGO = 1                                        ABCD3847
26    WRITE (8,34) LOOPER                                                ABCD3848
      DO 27  I = 1,NMAX                                                  ABCD3849
27    WRITE (8,35) AMAT(I),(EMAT(I,L),L=1,9),VMAT(I),DELVAR(I),VAR(I)    ABCD3850
      WRITE (8,36) ERRAVE,VMTAVE,DELAVE                                  ABCD3851
28    IF (LOOPER .LT. ITRYS)  GO TO 15                                   ABCD3852
      CALL ERROR                                                         ABCD3853
      RETURN                                                             ABCD3854
29    VMTAVX = VMTAVE                                                    ABCD3855
      DO 30  I = 1,NMAX                                                  ABCD3856
30    AMAT(I) = - ERR(I)                                                 ABCD3857
      GO TO 23                                                           ABCD3858
31    WRITE (8,37) AMAT,ERRAVE,DELVAR,DELAVE,VMAT,VMTAVE,VAR             ABCD3859
      MISMAT = MISMAT + 1                                                ABCD3860
      IF (VMTAVE .LT. VCHNGE * VMTAVX)  GO TO 28                         ABCD3861
      WRITE (8,38)                                                       ABCD3862
      IF (MISMAT .LT. NOMISX)  NOMISS = 1                                ABCD3863
      MISMAT = 0                                                         ABCD3864
      LOOP = 0                                                           ABCD3865
      IGO = 2                                                            ABCD3866
      GO TO 5                                                            ABCD3867
C                                                                        ABCD3868
C                                                                        ABCD3869
32    FORMAT (4H0ROW,I2,16H IS ZERO IN EMAT)                             ABCD3870
33    FORMAT (7H0COLUMN,I2,16H IS ZERO IN EMAT)                          ABCD3871
34    FORMAT (8HB   ERRB,28X,23HERROR MATRIX AFTER LOOP,I4,29X,4HVMAT,   ABCD3872
     1 6X,6HDELVAR,7X,14HVARIABLE$$$$$$)                                 ABCD3873
35    FORMAT (1H0,F8.4,10F9.3,2F11.4,6H$$$$$$)                           ABCD3874
36    FORMAT (1H0,F8.4,32X,14HAVERAGE VALUES,31X,2F11.4,6H$$$$$$)        ABCD3875
37    FORMAT (12H0-----  AMAT,10F11.5,6H$$$$$$,/,12H -----DELVAR,       ABCD3876
     1 10F11.6,6H$$$$$$,/,12H -----  VMAT,10F11.6,6H$$$$$$,/,           ABCD3877
     2 12H -----   VAR,9F11.6,6H$$$$$$)                                 ABCD3878
38    FORMAT (1H0,50X,22HCHANGE TOO SMALL$$$$$$)                         ABCD3879
      END                                                                ABCD3880
```

Subroutine ERROR

```
      SUBROUTINE ERROR                                                  ABCD3881
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD3882
      LOGICAL  ERRER, DUMSPL, FXFN2M, FXM2CP, AFTFAN, FAN               ABCD3883
      COMMON /COMALL/  COM(1062)                                        ABCD3884
      DIMENSION  WORD(2)                                                ABCD3885
      DIMENSION  AWORD(2)                                               ABCD3886
      EQUIVALENCE  (WORD(1), COM(1)), (MODE, COM(6)), (IDUMP, COM(8)),  ABCD3887
     1 (LOOPER, COM(19)), (ZF, COM(136)), (PCNF, COM(137)),             ABCD3888
     2 (ZI, COM(139)), (PCNI, COM(140)), (T4, COM(156)), (ZC, COM(300)),ABCD3889
     3 (PCNC, COM(301)), (ISPOOL, COM(1044)), (ICOAFB, COM(1045)),      ABCD3890
     4 (ICODUC, COM(1046)), (ICOMIX, COM(1047)), (ERREP, COM(1056)),    ABCD3891
     5 (DUMSPL, COM(1057)), (FXFN2M, COM(1058)), (FXM2CP, COM(1059)),   ABCD3892
     6 (AFTFAN, COM(1060)), (FAN, COM(1061))                            ABCD3893
      DATA  AWORD /4HCOMM, 4HON  /                                      ABCD3894
      IF (ICOAFB .LT. 1)  ICOAFB = 0                                    ABCD3895
      IF (ICOMIX .LT. 1)  ICOMIX = 0                                    ABCD3896
      IF (ICODUC .LT. 1)  ICODUC = 0                                    ABCD3897
      IF (ICOAFB .NE. 0)  WRITE (6,10) ICOAFB                           ABCD3898
      IF (ICODUC .NE. 0)  WRITE (6,11) ICODUC                           ABCD3899
      IF (ICOMIX .NE. 0)  WRITE (6,12) ICOMIX                           ABCD3900
      ERRER = .TRUE.                                                    ABCD3901
      WRITE (6,2) WORD                                                  ABCD3902
      WORD(1) = AWORD(1)                                                ABCD3903
      WORD(2) = AWORD(2)                                                ABCD3904
      WRITE (6,3) WORD,ZF,PCNF,ZI,PCNI,ZC,PCNC,T4,MODE                  ABCD3905
      WRITE (6,4)                                                       ABCD3906
      WRITE (6,5)   (COM(I),  I = 33,394)                               ABCD3907
      WRITE (6,4)                                                       ABCD3908
      WRITE (6,8)  DUMSPL, FXFN2M, FXM2CP, AFTFAN, FAN, ISPOOL          ABCD3909
      WRITE (6,4)                                                       ABCD3910
      WRITE (6,7) LOOPER                                                ABCD3911
      IF (IDUMP .EQ. 0)  GO TO 1                                        ABCD3912
      WRITE (6,6)                                                       ABCD3913
      CALL SYG (2)                                                      ABCD3914
1     CALL ENGBAL                                                       ABCD3915
      RETURN                                                            ABCD3916
C                                                                       ABCD3917
C                                                                       ABCD3918
2     FORMAT (28H0AN ERROR HAS BEEN FOUND IN ,A4,A2)                    ABCD3919
3     FORMAT (1H0,A4,A2,9X,7E15.6,I4)                                   ABCD3920
4     FORMAT (2H0 )                                                     ABCD3921
5     FORMAT (1H ,8E15.6)                                               ABCD3922
6     FORMAT (1H1)                                                      ABCD3923
7     FORMAT (25H0FAILED TO CONVERGE AFTER,I4,6H LOOPS)                 ABCD3924
8     FORMAT (1H ,5E15.6,I12)                                           ABCD3925
10    FORMAT (27H THE ERROR IN COAFBN IS AT ,I3)                        ABCD3926
11    FORMAT (27H THE ERROR IN CODUCT IS AT ,I3)                        ABCD3927
12    FORMAT (27H THE ERROR IN COMIX  IS AT ,I3)                        ABCD3928
      END                                                               ABCD3929
```

Subroutine ETAAB

```
      SUBROUTINE ETAAB (FAR,EM6,P6,ETA,ETAADS,ETAASV,P6DS,P6DSAV,AM6DS,  ABCD3930
     1 AM6DSV,IDES,FAR7DS,FAR7SV)                                        ABCD3931
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD3932
      DIMENSION  TFAR(25), ETABRT(25), EM6T(7), DELM6(7), P6T(14),       ABCD3933
     1 DELP6(14), X(3), Y(3)                                             ABCD3934
      DATA  TFAR/.039D0, .0585D0, .0732D0, .0878D0, .0976D0, .1171D0,    ABCD3935
     1 .1268D0, .1463D0, .1619D0, .1834D0, .1951D0, .2195D0, .2439D0,    ABCD3936
     2 .2927D0, .3415D0, .4146D0, .4634D0, .5366D0, .6341D0, .7317D0,    ABCD3937
     3 .8293D0, .9268D0, 1.0D0, 1.0634D0, 1.7D0/                         ABCD3938
      DATA  ETABRT/.94D0, .9887D0, 1.0193D0, 1.0306D0, 1.0227D0,         ABCD3939
     1 .9672D0, .9377D0, .9207D0, .9354D0, .9626D0, .9773D0, 1.0193D0,   ABCD3940
     2 1.0532D0, 1.077D0, 1.0781D0, 1.077D0, 1.0747D0, 1.0668D0,         ABCD3941
     3 1.0578D0, 1.051D0, 1.0374D0, 1.0192D0, 1.0D0, .9626D0, .9151D0/   ABCD3942
      DATA  EM6T/1.0D0, 1.071D0, 1.19D0, 1.309D0, 1.428D0, 1.547D0,      ABCD3943
     1 1.666D0 /                                                         ABCD3944
      DATA  DELM6/0.0D0, .013D0, .041D0, .073D0, .11D0, .147D0, .187D0/  ABCD3945
      DATA  P6T/.22D0, .2267D0, .25D0, .3D0, .3333D0, .3767D0, .4167D0,  ABCD3946
     1 .5D0, .5833D0, .5556D0, .75D0, .8333D0, .9167D0, 1.0D0/           ABCD3947
      DATA  DELP6/-.142D0, -.125D0, -.1D0, -.075D0, -.062D0, -.05D0,     ABCD3948
     1 -.041D0, -.027D0, -.019D0, -.013D0, -.008D0, -.004D0, -.0021D0,   ABCD3949
     2 0.0D0/                                                            ABCD3950
      IF (IDES .NE. 1)  GO TO 5                                          ABCD3951
      EMULT = ETAADS / ETAASV                                            ABCD3952
      FMULT = FAR7DS / FAR7SV                                            ABCD3953
      AMULT = AM6DS / AM6DSV                                             ABCD3954
      PMULT = P6DS / P6DSAV                                              ABCD3955
      DO 1  K = 1,25                                                     ABCD3956
      ETABRT(K) = ETABRT(K) * EMULT                                      ABCD3957
1     TFAR(K) = TFAR(K) * FMULT                                          ABCD3958
      DO 3  K = 1,7                                                      ABCD3959
3     EM6T(K) = EM6T(K) * AMULT                                          ABCD3960
      DO 4  M = 1,14                                                     ABCD3961
4     P6T(M) = P6T(M) * PMULT                                            ABCD3962
      ETAASV = ETAADS                                                    ABCD3963
      P6DSAV = P6DS                                                      ABCD3964
      FAR7SV = FAR7DS                                                    ABCD3965
      AM6DSV = AM6DS                                                     ABCD3966
      RETURN                                                             ABCD3967
5     N = 0                                                              ABCD3968
      IF (FAR .GT. 0.067D0)  GO TO 8                                     ABCD3969
      DO 6  J = 1,25                                                     ABCD3970
      IF (FAR .GE. TFAR(J))  N = J - 1                                   ABCD3971
6     CONTINUE                                                           ABCD3972
      IF (N .EQ. 0)  N = 1                                               ABCD3973
      IF (N .GE. 24)  N = 23                                             ABCD3974
      DO 7  I = 1,3                                                      ABCD3975
      NN = N - 1 + I                                                     ABCD3976
      X(I) = TFAR(NN)                                                    ABCD3977
7     Y(I) = ETABRT(NN)                                                  ABCD3978
      CALL PARABO (X,Y,FAR,ETA1)                                         ABCD3979
      GO TO 9                                                            ABCD3980
8     ETA1 = - 2.0D0 * FAR + .1948D0                                     ABCD3981
```

```
9     M = 0                                                              ABCD3982
      DO 10  J = 1,7                                                     ABCD3983
      IF (EM6 .GE. EM6T(J))  M = J - 1                                   ABCD3984
10    CONTINUE                                                           ABCD3985
      IF (M .EQ. 0)  M = 1                                               ABCD3986
      IF (M .GE. 6)  M = 5                                               ABCD3987
      DO 11  I = 1,3                                                     ABCD3988
      MM = M - 1 + I                                                     ABCD3989
      X(I) = EM6T(MM)                                                    ABCD3990
11    Y(I) = DELM6(MM)                                                   ABCD3991
      CALL PARABO (X,Y,EM6,COR1)                                         ABCD3992
      L = 0                                                              ABCD3993
      DO 12  J = 1,14                                                    ABCD3994
      IF (P6 .GE. P6T(J))  L = J - 1                                     ABCD3995
12    CONTINUE                                                           ABCD3996
      IF (L .EQ. 0)  L = 1                                               ABCD3997
      IF (L .GE. 13)  L = 12                                             ABCD3998
      DO 13  I = 1,3                                                     ABCD3999
      LL = L - 1 + I                                                     ABCD4000
      X(I) = P6T(LL)                                                     ABCD4001
13    Y(I) = DELP6(LL)                                                   ABCD4002
      CALL PARABO (X,Y,P6,COR2)                                          ABCD4003
      ETA = ETA1 * (1.0D0 - COR1) * (1.0D0 + COR2)                       ABCD4004
      RETURN                                                             ABCD4005
      END                                                                ABCD4006
```

Subroutine FASTBK

```
      SUBROUTINE FASTBK                                                  ABCD4007
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD4008
      LOGICAL  FAN                                                       ABCD4009
      COMMON /COMALL/  COM(1062)                                         ABCD4010
      EQUIVALENCE  (P1, COM(33)), (T25, COM(46)), (P25, COM(47)),        ABCD4011
     1 (H25, COM(48)), (S25, COM(49)), (WFD, COM(72)), (XXP1, COM(76)),  ABCD4012
     2 (XWG24, COM(77)), (XFAR24, COM(78)), (XT25, COM(79)),             ABCD4013
     3 (XP25, COM(80)), (XH25, COM(81)), (XS25, COM(82)),                ABCD4014
     4 (XWG55, COM(83)), (XFAR55, COM(84)), (XT55, COM(85)),             ABCD4015
     5 (XP55, COM(86)), (XH55, COM(87)), (XS55, COM(88)),                ABCD4016
     6 (XWFB, COM(89)), (XWFD, COM(90)), (WFB, COM(192)),                ABCD4017
     7 (T21, COM(263)), (H21, COM(264)), (S21, COM(265)),                ABCD4018
     8 (T55, COM(272)), (H55, COM(273)), (S55, COM(274)),                ABCD4019
     9 (BLF, COM(316)), (WAF, COM(319)), (WG24, COM(321)),               ABCD4020
     1 (FAR24, COM(322)), (WG55, COM(323)), (FAR55, COM(324)),           ABCD4021
     2 (P21, COM(377)), (P55, COM(387)), (FAN, COM(1061))                ABCD4022
      XT55 = T55                                                         ABCD4023
      XP55 = P55                                                         ABCD4024
      XH55 = H55                                                         ABCD4025
      XS55 = S55                                                         ABCD4026
      IF (FAN)  GO TO 1                                                  ABCD4027
      T25 = T21                                                          ABCD4028
      P25 = P21                                                          ABCD4029
      H25 = H21                                                          ABCD4030
      S25 = S21                                                          ABCD4031
      WG24 = WAF - BLF                                                   ABCD4032
```

```
1       XT25 = T25                                                      ABCD4033
        XP25 = P25                                                      ABCD4034
        XH25 = H25                                                      ABCD4035
        XS25 = S25                                                      ABCD4036
        XWFB = WFB                                                      ABCD4037
        XWG55 = WG55                                                    ABCD4038
        XFAR55 = FAR55                                                  ABCD4039
        XWFD = WFD                                                      ABCD4040
        XWG24 = WG24                                                    ABCD4041
        XFAR24 = FAR24                                                  ABCD4042
        XXP1 = P1                                                       ABCD4043
        CALL COMIX                                                      ABCD4044
        RETURN                                                          ABCD4045
        END                                                             ABCD4046
```

Subroutine FCNTRL

```
        SUBROUTINE FCNTRL                                               ABCD4047
        IMPLICIT  REAL*8  (A-H,O-Z)                                     ABCD4048
        RETURN                                                          ABCD4049
        END                                                             ABCD4050
```

Subroutine FRTOSD

```
        SUBROUTINE FRTOSD                                               ABCD4051
        IMPLICIT  REAL*8  (A-H,O-Z)                                     ABCD4052
        LOGICAL  FAN                                                    ABCD4053
        COMMON /COMALL/  COM(1062)                                      ABCD4054
        EQUIVALENCE  (P1, COM(33)), (H3, COM(153)), (WAC, COM(191)),    ABCD4055
       1 (XP1, COM(200)), (XT21, COM(201)), (XP21, COM(202)),           ABCD4056
       2 (XH21, COM(203)), (XS21, COM(204)), (XH3, COM(205)),           ABCD4057
       3 (XWAF, COM(206)), (XWAC, COM(207)), (XBLF, COM(208)),          ABCD4058
       4 (XBLDU, COM(209)), (T21, COM(263)), (H21, COM(264)),           ABCD4059
       5 (S21, COM(265)), (BLF, COM(316)), (BLDU, COM(317)),            ABCD4060
       6 (WAF, COM(319)), (P21, COM(377)), (FAN, COM(1061))             ABCD4061
        XP1 = P1                                                        ABCD4062
        XWAF = WAF                                                      ABCD4063
        XWAC = WAC                                                      ABCD4064
        XBLF = BLF                                                      ABCD4065
        XBLDU = BLDU                                                    ABCD4066
        XH3 = H3                                                        ABCD4067
        XT21 = T21                                                      ABCD4068
        XP21 = P21                                                      ABCD4069
        XH21 = H21                                                      ABCD4070
        XS21 = S21                                                      ABCD4071
        IF (FAN)  CALL CODUCT                                           ABCD4072
        IF (.NOT. FAN)  CALL FASTBK                                     ABCD4073
        RETURN                                                          ABCD4074
        END                                                             ABCD4075
```

Function GUESS

```
        FUNCTION GUESS (M, T, TD, P, PD, W, WD, D, DD, VD)              ABCD4076
        IMPLICIT  REAL*8  (A-H,O-Z)                                     ABCD4077
```

```
      IF (M .EQ. 0)  GUESS = VD * ((T / TD) ** 1.6D0) * ((DD / D) **     ABCD4078
     1 0.5D0)                                                            ABCD4079
      IF (M .LE. 0 .OR. M .GT. 9)  RETURN                                ABCD4080
      GO TO (1,2,3,3,5,6,7,8,9), M                                       ABCD4081
1     GUESS = VD * ((P / PD) ** 1.8D0) * ((DD / D) ** 0.33D0)            ABCD4082
      RETURN                                                             ABCD4083
2     GUESS = VD * ((W / WD) ** 0.33D0) * DD / D                         ABCD4084
      RETURN                                                             ABCD4085
3     GUESS = VD * ((P / PD) ** .5D0)                                    ABCD4086
      RETURN                                                             ABCD4087
5     GUESS = VD * ((T / TD) ** 1.1D0) * ((DD / D) ** .7D0)              ABCD4088
      RETURN                                                             ABCD4089
6     GUESS = VD * P / PD * ((D / DD) ** 0.25D0)                         ABCD4090
      RETURN                                                             ABCD4091
7     GUESS = VD * ((P / PD) ** 0.62D0) * ((D / DD) ** 0.31D0)           ABCD4092
      RETURN                                                             ABCD4093
8     GUESS = VD * ((T / TD) ** 1.2D0) * DD / D                          ABCD4094
      RETURN                                                             ABCD4095
9     GUESS = VD * P / PD * ((D / DD) ** 1.5D0)                          ABCD4096
      RETURN                                                             ABCD4097
      END                                                                ABCD4098
```

Subroutine INDUMY

```
      SUBROUTINE INDUMY (CNI,ZI,WACI,IDES)                               ABCD4099
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD4100
      COMMON /COMDAT/  COMD(5423)                                        ABCD4101
      DIMENSION  CNXXI(15), PRXXI(15,15), WACXXI(15,15), ETAXXI(15,15),  ABCD4102
     1 NPTXI(15)                                                         ABCD4103
      DIMENSION  XCNXX(15), WACAR(15)                                    ABCD4104
      EQUIVALENCE  (CNXXI(1), COMD(1381)), (PRXXI(1,1), COMD(1396)),     ABCD4105
     1 (WACXXI(1,1), COMD(1621)), (ETAXXI(1,1), COMD(1846)),             ABCD4106
     2 (NCNXI, COMD(5328)), (NPTXI(1), COMD(5329))                       ABCD4107
      DATA  XCNXX /.001D0, .1D0, .2D0, .3D0, .5D0, .8D0, 1.0D0, 1.5D0,   ABCD4108
     1 2.0D0, 3.0D0, 4.0D0, 5.0D0, 6.0D0, 7.0D0, 9.0D0/                  ABCD4109
      DATA  WACAR /5.0D0, 4.5D0, 4.0D0, 3.5D0, 3.0D0, 2.5D0, 2.0D0,     ABCD4110
     1 1.5D0, 1.0D0, .8D0, .6D0, .4D0, .25D0, .1D0, .05D0/               ABCD4111
      IF (IDES .NE. 1)  GO TO 1                                          ABCD4112
      WAIDS = WACI                                                       ABCD4113
      CNIDS = CNI                                                        ABCD4114
      ZI = 2.0D0 / 3.5D0                                                 ABCD4115
1     NCNXI = 15                                                         ABCD4116
      DO 2  I = 1,15                                                     ABCD4117
      NPTXI(I) = 15                                                      ABCD4118
      CNXXI(I) = XCNXX(I) * CNIDS                                        ABCD4119
      DO 2  J = 1,15                                                     ABCD4120
      PRXXI(I,J) = FLOAT(J + 3) / 4.0D0                                  ABCD4121
      ETAXXI(I,J) = 1.0D0                                                ABCD4122
2     WACXXI(J,I) = WACAR(I) * (.993D0 + .001D0 * FLOAT(J)) * WAIDS      ABCD4123
      RETURN                                                             ABCD4124
      END                                                                ABCD4125
```

```
      SUBROUTINE MATRIX (E,V,A,N)                                       ABCD4126
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD4127
      DIMENSION  E(9,9), V(9), A(9), PIV(10), T(9,10)                   ABCD4128
      NN = N + 1                                                        ABCD4129
      NM = N - 1                                                        ABCD4130
      DO 1  I = 1,N                                                     ABCD4131
      T(I,NN) = A(I)                                                    ABCD4132
      DO 1  J = 1,N                                                     ABCD4133
1     T(I,J) = E(I,J)                                                   ABCD4134
      DO 7  I = 1,N                                                     ABCD4135
      TEMP = 0.0D0                                                      ABCD4136
      DO 2  J = I,N                                                     ABCD4137
      IF (TEMP .GT. DABS(T(J,I)))  GO TO 2                              ABCD4138
      TEMP = DABS(T(J,I))                                               ABCD4139
      IPIV = J                                                          ABCD4140
2     CONTINUE                                                          ABCD4141
      IP1 = I + 1                                                       ABCD4142
      DO 3  J = IP1,NN                                                  ABCD4143
3     PIV(J) = T(IPIV,J) / T(IPIV,I)                                    ABCD4144
      IFROM = N                                                         ABCD4145
      ITO = N                                                           ABCD4146
4     IF (IFROM .EQ. IPIV)  GO TO 6                                     ABCD4147
      RM = - T(IFROM,I)                                                 ABCD4148
      DO 5  J = IP1,NN                                                  ABCD4149
5     T(ITO,J) = T(IFROM,J) + RM * PIV(J)                               ABCD4150
      ITO = ITO - 1                                                     ABCD4151
6     IFROM = IFROM - 1                                                 ABCD4152
      IF (IFROM .GE. I)  GO TO 4                                        ABCD4153
      DO 7  J = IP1,NN                                                  ABCD4154
7     T(I,J) = PIV(J)                                                   ABCD4155
      DO 8  I = 1,NM                                                    ABCD4156
      J = NN - I                                                        ABCD4157
      K = N - I                                                         ABCD4158
      DO 8  L = J,N                                                     ABCD4159
8     T(K,NN) = T(K,NN) - T(K,L) * T(L,NN)                              ABCD4160
      DO 9  I = 1,N                                                     ABCD4161
9     V(I) = T(I,NN)                                                    ABCD4162
      RETURN                                                            ABCD4163
      END                                                               ABCD4164
```

Subroutine NOZCTR

```
      SUBROUTINE NOZCTR                                                 ABCD4165
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD4166
      RETURN                                                            ABCD4167
      END                                                               ABCD4168
```

Subroutine OUTPUT

```
      SUBROUTINE OUTPUT                                                 ABCD4169
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD4170
      LOGICAL  DUMSPL, FXFN2M, FXM2CP, AFTFAN, FAN                      ABCD4171
      COMMON /COMALL/  COM(1062)                                        ABCD4172
      DIMENSION  WORD(2)                                                ABCD4173
```

```
      DIMENSION  W(6,5), AWORD1(2), AWORD2(2)                                ABCD4174
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MODE, COM(6)),       ABCD4175
     1 (IDUMP, COM(8)), (IGASMX, COM(10)), (IDBURN, COM(11)),               ABCD4176
     2 (IAFTBN, COM(12)), (IDSHOC, COM(15)), (IMSHOC, COM(16)),             ABCD4177
     3 (LOOPER, COM(19)), (T24, COM(42)), (A28, COM(50)),                   ABCD4178
     4 (A28SAV, COM(51)), (A29, COM(60)), (A29SAV, COM(61)),                ABCD4179
     5 (ZF, COM(136)), (PCNF, COM(137)), (ZI, COM(139)),                    ABCD4180
     6 (PCNI, COM(140)), (T4, COM(155)), (ETAR, COM(187)),                  ABCD4181
     7 (AM, COM(195)), (ALTP, COM(196)), (SFC, COM(248)),                   ABCD4182
     8 (FG, COM(257)), (FN, COM(258)), (ZC, COM(300)), (PCNC, COM(301)),    ABCD4183
     9 (PCBLID, COM(305)), (T7, COM(343)), (A8, COM(346)),                  ABCD4184
     1 (A8SAV, COM(347)), (A9, COM(356)), (A9SAV, COM(357)),                ABCD4185
     2 (TIME, COM(993)), (TPRINT, COM(996)), (DTPRNT, COM(997)),            ABCD4186
     3 (ISPOOL, COM(1044)), (ITRAN, COM(1049)), (IAMTRX, COM(1054)),        ABCD4187
     4 (DUMSPL, COM(1057)), (FXFN2M, COM(1058)), (FXM2CP, COM(1059)),       ABCD4188
     5 (AFTFAN, COM(1060)), (FAN, COM(1061))                                ABCD4189
      DATA  AWORD1, AWORD2 /4HOUTP, 4HUT  , 4HCOMM, 4HON  /                 ABCD4190
      DATA  W / 4HSUBS, 4HONIC, 4H C-D, 4H NOZ, 4HZLE , 4H    ,             ABCD4191
     1          4HSHOC, 4HK IN, 4HSIDE, 4H C-D, 4H NOZ, 4HZLE ,             ABCD4192
     2          4HSHOC, 4HK OU, 4HTSID, 4HE C-, 4HD NO, 4HZZLE,             ABCD4193
     3          4HSUBS, 4HONIC, 4H CON, 4HVERG, 4H. NO, 4HZZLE,             ABCD4194
     4          4HSONI, 4HC CO, 4HNVER, 4HGENT, 4H NOZ, 4HZLE   /           ABCD4195
      IF (IAMTRX .EQ. 1 .AND. ITRAN .EQ. 1)  GO TO 24                       ABCD4196
      TPRINT = TPRINT + DTPRNT                                              ABCD4197
      IF (ITRAN .EQ. 1)  WRITE(6,29) TIME                                   ABCD4198
      WORD(1) = AWORD1(1)                                                   ABCD4199
      WORD(2) = AWORD1(2)                                                   ABCD4200
      IF (IDBURN .GT. 0)  GO TO 2                                           ABCD4201
      IF (IAFTBN .GT. 0)  GO TO 1                                           ABCD4202
      WRITE (6,7) WORD,AM,ALTP,T4,ETAR                                      ABCD4203
      GO TO 3                                                               ABCD4204
1     WRITE (6,8) WORD,AM,ALTP,T4,T7,ETAR                                   ABCD4205
      GO TO 3                                                               ABCD4206
2     WRITE (6,9) WORD,AM,ALTP,T4,T24,ETAR                                  ABCD4207
3     IF (FXFN2M)  WRITE (6,17)                                             ABCD4208
      IF (FXM2CP)  WRITE (6,18)                                             ABCD4209
      IF (FAN)  GO TO 25                                                    ABCD4210
      WRITE(6,26) ISPOOL                                                    ABCD4211
      GO TO 27                                                              ABCD4212
25    IF (.NOT. FXFN2M .AND. (.NOT. FXM2CP) .AND. (.NOT. DUMSPL))           ABCD4213
     1 WRITE (6,19)                                                         ABCD4214
      IF (DUMSPL)  WRITE (6,23)                                             ABCD4215
      IF (PCBLID .EQ. 0.0D0)  WRITE (6,20)                                  ABCD4216
      IF (PCBLID .EQ. 0.0D0 .AND. AFTFAN)  WRITE (6,21)                     ABCD4217
      IF (PCBLID .NE. 0.0D0 .AND. AFTFAN)  WRITE (6,22)                     ABCD4218
27    CALL CONOUT(2)                                                        ABCD4219
      WRITE (6,10) (W(I,IMSHOC),I=1,6),FG,FN,SFC                            ABCD4220
      IF (IGASMX .GT. 0 .OR. .NOT. FAN)  GO TO 4                            ABCD4221
      WRITE (6,11) (W(I,IDSHOC),I=1,6)                                      ABCD4222
4     WRITE (6,12) LOOPER                                                   ABCD4223
      IF (IDES .NE. 1)  GO TO 5                                             ABCD4224
      WORD(1) = AWORD2(1)                                                   ABCD4225
      WORD(2) = AWORD2(2)                                                   ABCD4226
      WRITE (6,13) WORD,ZF,PCNF,ZI,PCNI,ZC,PCNC,T4,MODE                     ABCD4227
```

```
      WRITE (6,14)                                                          ABCD4228
      WRITE (6,15)   (COM(I),  I = 33,394)                                  ABCD4229
      WRITE (6,14)                                                          ABCD4230
      WRITE (6,15)   DUMSPL, FXFN2M, FXM2CP, AFTFAN, FAN                    ABCD4231
      WRITE (6,14)                                                          ABCD4232
      WRITE (6,16)                                                          ABCD4233
24    IF (IDES .EQ. 1)  GO TO 6                                             ABCD4234
5     A8 = A8SAV                                                            ABCD4235
      A9 = A9SAV                                                            ABCD4236
      A28 = A28SAV                                                          ABCD4237
      A29 = A29SAV                                                          ABCD4238
      IF (IDUMP .NE. 2)  GO TO 6                                            ABCD4239
      WRITE (6,16)                                                          ABCD4240
      CALL SYG (2)                                                          ABCD4241
6     CALL ENGBAL                                                           ABCD4242
      RETURN                                                                ABCD4243
C                                                                           ABCD4244
C                                                                           ABCD4245
C                                                                           ABCD4246
7     FORMAT  (1H , A4, A2, 14X, 7H    AM=, F7.3, 6X, 7H  ALTP=, F7.0,    ABCD4247
     1 6X, 7H    T4=, F8.2, 25X, 7H  ETAR=, F7.4)                           ABCD4248
8     FORMAT  (1H , A4, A2, 14X, 7H    AM=, F7.3, 6X, 7H  ALTP=, F7.0,    ABCD4249
     1 6X, 7H    T4=, F8.2, 5X, 7H    T7=, F8.2, 5X, 7H  ETAR=, F7.4)       ABCD4250
9     FORMAT  (1H , A4, A2, 14X, 7H    AM=, F7.3, 6X, 7H  ALTP=, F7.0,    ABCD4251
     1 6X, 7H    T4=, F8.2, 5X, 7H   T24=, F8.2, 5X, 7H  ETAR=, F7.4)       ABCD4252
10    FORMAT  (1H0, 5HMAIN , 6A4, 9X, 3HFG=, F9.2, 18X, 3HFN=, F9.2,       ABCD4253
     1 18X, 4HSFC=, F8.5)                                                   ABCD4254
11    FORMAT (6H DUCT ,6A4)                                                 ABCD4255
12    FORMAT (16H0CONVERGED AFTER,I4,6H LOOPS,/,1H1)                        ABCD4256
13    FORMAT (1H ,A4,A2,9X,7E15.6,I4)                                       ABCD4257
14    FORMAT (1H )                                                          ABCD4258
15    FORMAT (1H ,8E15.6)                                                   ABCD4259
16    FORMAT (1H1)                                                          ABCD4260
17    FORMAT (51H0FAN AND MIDDLE SPOOL ARE ATTACHED , USE INNER AND ,       ABCD4261
     1 14HOUTER TURBINES)                                                   ABCD4262
18    FORMAT (49H0MIDDLE AND COMPRESSOR SPOOLS ARE ATTACHED , USE ,         ABCD4263
     1 25HMIDDLE AND OUTER TURBINES)                                        ABCD4264
19    FORMAT (19H0THREE SPOOL ENGINE)                                       ABCD4265
20    FORMAT (21H0NO AIRFLOW INTO WING)                                     ABCD4266
21    FORMAT (1H+,22X,14H, AFT-TURBOFAN)                                    ABCD4267
22    FORMAT (14H0 AFT-TURBOFAN)                                            ABCD4268
23    FORMAT (22H0MIDDLE SPOOL IS DUMMY)                                    ABCD4269
26    FORMAT(1H0,I4,15H SPOOL TURBOJET)                                     ABCD4270
29    FORMAT(1H1,20X,7H  TIME=,F7.4)                                        ABCD4271
      END                                                                   ABCD4272
```

Subroutine PARABO

```
      SUBROUTINE PARABO (X,Y,XD,YANS)                                       ABCD4273
      IMPLICIT  REAL*8  (A-H,O-Z)                                           ABCD4274
      DIMENSION  X(3), Y(3)                                                 ABCD4275
      X1M2 = X(1) - X(2)                                                    ABCD4276
      X1M3 = X(1) - X(3)                                                    ABCD4277
      X2M1 = X(2) - X(1)                                                    ABCD4278
```

```
      X2M3 = X(2) - X(3)                                                ABCD4278
      X3M2 = X(3) - X(2)                                                ABCD4280
      X1SQ = X(1) * X(1)                                                ABCD4281
      X2SQ = X(2) * X(2)                                                ABCD4282
      X3SQ = X(3) * X(3)                                                ABCD4283
      Y1M2 = Y(1) - Y(2)                                                ABCD4284
      Y1M3 = Y(1) - Y(3)                                                ABCD4285
      A = (X1M2 * Y1M3 - X1M3 * Y1M2) / X1M2 / X1M3 / X3M2              ABCD4286
      B = ((X1SQ - X2SQ) * Y1M3 - (X1SQ - X3SQ) * Y1M2) / X1M2 / X1M3 / ABCD4287
     1 X2M3                                                             ABCD4288
      D = (Y(1) * X2SQ - Y(2) * X1SQ - B * X(2) * X(1) * X2M1) / (X2SQ  ABCD4289
     1 - X1SQ)                                                          ABCD4290
      YANS = (A * XD + B) * XD + D                                      ABCD4291
      RETURN                                                            ABCD4292
      END                                                               ABCD4293
```

Subroutine PERF

```
      SUBROUTINE PERF                                                   ABCD4294
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD4295
      LOGICAL  SI, DUMSPL, AFTFAN, FAN                                  ABCD4296
      COMMON /COMALL/  COM(1062)                                        ABCD4297
      DIMENSION  WORD(2)                                                ABCD4298
      DIMENSION  AWORD(2)                                               ABCD4299
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (IGASMX, COM(10)),ABCD4300
     1 (P1, COM(33)), (A29, COM(60)), (V29, COM(63)), (PS29, COM(65)),  ABCD4301
     2 (WAD, COM(71)), (WFD, COM(72)), (PCNI, COM(140)),                ABCD4302
     3 (WFB, COM(192)), (CS, COM(194)), (AM, COM(195)),                 ABCD4303
     4 (WG37, COM(210)), (A39, COM(219)), (V39, COM(221)),              ABCD4304
     5 (PS39, COM(226)), (FGMWNG, COM(230)), (FGPWNG, COM(231)),        ABCD4305
     6 (FNWING, COM(232)), (FNMAIN, COM(233)), (FWOVFN, COM(234)),      ABCD4306
     7 (DELFG, COM(236)), (DELFN, COM(237)), (DELSFC, COM(238)),        ABCD4307
     8 (CVDWNG, COM(239)), (CVDNOZ, COM(240)), (CVMNOZ, COM(241)),      ABCD4308
     9 (VA, COM(242)), (VJD, COM(243)), (VJW, COM(244)),                ABCD4309
     1 (VJM, COM(245)), (WFT, COM(246)), (WGT, COM(247)),               ABCD4310
     2 (SFC, COM(248)), (TFAR, COM(249)), (FRD, COM(250)),              ABCD4311
     3 (FGMD, COM(251)), (FGMM, COM(252)), (FGPD, COM(253)),            ABCD4312
     4 (FGPM, COM(254)), (FGM, COM(255)), (FGP, COM(256)),              ABCD4313
     5 (FG, COM(257)), (FN, COM(258)), (FFOVFN, COM(259)),              ABCD4314
     6 (FCOVFN, COM(260)), (FMNOFN, COM(261)), (FNOVFD, COM(262)),      ABCD4315
     7 (WA32, COM(271)), (PCBLID, COM(305)), (CNI, COM(309)),           ABCD4316
     8 (WAI, COM(311)), (BLOB, COM(318)), (WAF, COM(319)),              ABCD4317
     9 (WG24, COM(321)), (WG7, COM(334)), (A9, COM(356)),               ABCD4318
     1 (PS9, COM(359)), (V9, COM(360)), (WFA, COM(368))                 ABCD4319
      EQUIVALENCE  (TIME, COM(993)), (TPRINT, COM(996)),                ABCD4320
     1 (ITRAN, COM(1049)), (SI, COM(1055)), (DUMSPL, COM(1057)),        ABCD4321
     2 (AFTFAN, COM(1060)), (FAN, COM(1061))                            ABCD4322
      DATA  AWORD /4H  PE, 4HRF  /                                      ABCD4323
      WORD(1) = AWORD(1)                                                ABCD4324
      WORD(2) = AWORD(2)                                                ABCD4325
      IF (SI)  GO TO 100                                                ABCD4326
      G = 32.174049D0                                                   ABCD4327
      CAPSF = 2116.2170D0                                               ABCD4328
      GO TO 101                                                         ABCD4329
```

```
100   G = 1.0D0                                                          ABCD4330
      CAPSF = 1.0D0                                                      ABCD4331
101   WFT = WFB + WFD + WFA                                              ABCD4332
      WAT = WAF - BLOB                                                   ABCD4333
      IF (AFTFAN)  WAT = WAT + WAI                                       ABCD4334
      WGT = WAT + WFT                                                    ABCD4335
      TFAR = WFT / WAT                                                   ABCD4336
      VA = AM * CS                                                       ABCD4337
      FRD = VA * WAF / G                                                 ABCD4338
      IF (AFTFAN)  FRD = FRD + VA * WAI / G                              ABCD4339
      VJM = CVMNOZ * V9                                                  ABCD4340
      FGMM = VJM * WG7 / G                                               ABCD4341
      FGPM = CAPSF * (PS9 - P1) * A9                                     ABCD4342
      IF (IGASMX .GT. 0 .OR. .NOT. FAN)  GO TO 1                         ABCD4343
      VJD = CVDNOZ * V29                                                 ABCD4344
      FGMD = VJD * WG24 / G                                              ABCD4345
      FGPD = CAPSF * (PS29 - P1) * A29                                   ABCD4346
1     VJW = 0.0D0                                                        ABCD4347
      FGMWNG = 0.0D0                                                     ABCD4348
      FGPWNG = 0.0D0                                                     ABCD4349
      FGWING = 0.0D0                                                     ABCD4350
      FNWING = 0.0D0                                                     ABCD4351
      IF (PCBLID .EQ. 0.0D0)  GO TO 2                                    ABCD4352
      VJW = CVDWNG * V39                                                 ABCD4353
      FGMWNG = VJW * WG37 / G                                            ABCD4354
      FGPWNG = CAPSF * (PS39 - P1) * A39                                 ABCD4355
      FGWING = FGMWNG + FGPWNG                                           ABCD4356
      FNWING = FGWING - VA * WA32 / G                                    ABCD4357
2     FGM1 = FGMM + FGMD                                                 ABCD4358
      FGM = FGM1 + FGMWNG                                                ABCD4359
      FGP1 = FGPM + FGPD                                                 ABCD4360
      FGP = FGP1 + FGPWNG                                                ABCD4361
      FNMAIN = FGM1 + FGP1 - VA * (WAF - WA32) / G                       ABCD4362
      IF (AFTFAN)  FNMAIN = FNMAIN - VA * WAI / G                        ABCD4363
      FG = FGM + FGP                                                     ABCD4364
      FN = FG - FRD                                                      ABCD4365
      SFC = 3600.0D0 * WFT / FN                                          ABCD4366
      FG = DELFG * FG                                                    ABCD4367
      FN = DELFN * FN                                                    ABCD4368
      SFC = DELSFC * SFC                                                 ABCD4369
      FFAN = FGMD + FGPD - VA * WAD / G                                  ABCD4370
      FCORE = FNMAIN - FFAN                                              ABCD4371
      FFOVFN = FFAN / FN                                                 ABCD4372
      FCOVFN = FCORE / FN                                                ABCD4373
      FWOVFN = FNWING / FN                                               ABCD4374
      FMNOFN = FNMAIN / FN                                               ABCD4375
      IF (IDES .EQ. 1)  FDES = FN                                        ABCD4376
      FNOVFD = FN / FDES                                                 ABCD4377
      IF (.NOT. DUMSPL)  GO TO 3                                         ABCD4378
      PCNI = 1.0D0                                                       ABCD4379
      CNI = 0.0D0                                                        ABCD4380
3     IF (ITRAN .EQ. 1 .AND. TIME .LT. TPRINT)  CALL ENGBAL              ABCD4381
      CALL OUTPUT                                                        ABCD4382
      CALL ERROR                                                         ABCD4383
      RETURN                                                             ABCD4384
      END                                                                ABCD4385
```

Subroutine PROCOM

```
      SUBROUTINE PROCOM (FARX,TEX,CSEX,AKEX,CPEX,REX,PHI,HEX)
      IMPLICIT  REAL*8  (A-H,O-Z)
      LOGICAL SI
      COMMON /COMALL/  COM(1062)
      EQUIVALENCE  (SI, COM(1055))
C      IF SI UNITS ARE USED, CONVERT TEX TO DEGREES RANKINE
      IF (SI)  TEX = TEX * 9.0D0 / 5.0D0
      IF (FARX .GT. .067623D0)  FARX = .067623D0
      IF (TEX .LT. 300.0D0)  TEX = 300.0D0
      IF (TEX .GT. 4000.0D0)  TEX = 4000.0D0
      IF (FARX .LT. 0.0D0)  FARX = 0.0D0
C     AIR PATH
      CPA = ((((((1.011554D-25 * TEX - 1.452677D-21) * TEX +
     1 7.6215767D-18) * TEX - 1.5128259D-14) * TEX - 6.7178376D-12) *
     2 TEX + 6.5519486D-08) * TEX - 5.1536879D-05) * TEX + 2.5020051D-01
      HEA = (((((((1.2644425D-26 * TEX - 2.0752522D-22) * TEX +
     1 1.270263D-18) * TEX - 3.0256518D-15) * TEX - 1.6794594D-12) *
     2 TEX + 2.1839826D-08) * TEX - 2.576844D-05) * TEX +
     3 2.5020051D-01) * TEX - 1.7558886D+00
      SEA = 2.5020051D-01 * DLOG(TEX) + ((((((1.4450767D-26 * TEX -
     1 2.4211288D-22) * TEX + 1.5243153D-18) * TEX - 3.7820648D-15) *
     2 TEX - 2.239279D-12) * TEX + 3.2759743D-08) * TEX -
     3 5.1576879D-05) * TEX + 4.54323D-02
      IF (FARX .LE. 0.0D0)  GO TO 5
C     FUEL/AIR PATH
      CPF = ((((((7.267871D-25 * TEX - 1.3335668D-20) * TEX +
     1 1.0212913D-16) * TEX - 4.2051104D-13) * TEX + 9.9686793D-10) *
     2 TEX - 1.3771901D-06) * TEX + 1.225863D-03) * TEX + 7.3816638D-02
      HEF = (((((((9.0848388D-26 * TEX - 1.9050949D-21) * TEX +
     1 1.7021525D-17) * TEX - 8.4102208D-14) * TEX + 2.4921698D-10) *
     2 TEX - 4.5906332D-07) * TEX + 6.129315D-04) * TEX +
     3 7.3816638D-02) * TEX + 3.058153D+01
      SEF = 7.3816638D-02 * DLOG(TEX) + ((((((1.038267D-25 * TEX -
     1 2.2226118D-21) * TEX + 2.0425826D-17) * TEX - 1.0512776D-13) *
     2 TEX + 3.3228928D-10) * TEX - 5.8859505D-07) * TEX +
     3 1.225863D-03) * TEX + 6.483398D-01
5     CPEX = (CPA + FARX * CPF) / (1.0D0 + FARX)
      HEX = (HEA + FARX * HEF) / (1.0D0 + FARX)
      PHI = (SEA + FARX * SEF) / (1.0D0 + FARX)
      AMW = 28.97D0 - .946186D0 * FARX
      REX = 1.986375D0 / AMW
      AKEX = CPEX / (CPEX - REX)
      CSEX = DSQRT(AKEX * REX * TEX * 25031.37D0)
      IF (.NOT. SI)  RETURN
      CPEX = CPEX * 4185.7666D0
      HEX = HEX * 2325.4259D0
      PHI = PHI * 4185.7666D0
      REX = REX * 4185.7666D0
      CSEX = CSEX * .3048D0
```

```
      TEX = TEX * 5.0D0 / 9.0D0                                          ABCD4435
      RETURN                                                             ABCD4436
      END                                                                ABCD4437
```

Subroutine PUTIN

```
      SUBROUTINE PUTIN                                                   ABCD4438
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD4439
      LOGICAL  SI, ERRER, DUMSPL, FXFN2M, FXM2CP, AFTFAN, FAN            ABCD4440
      COMMON /COMALL/  COM(1062)                                         ABCD4441
      DIMENSION  WORD(2), TIMEPT(50)                                     ABCD4442
      DIMENSION  XSAVE(396), AWORD(2)                                    ABCD4443
      EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (MODE, COM(6)),    ABCD4444
     1 (INIT, COM(7)), (IDUMP, COM(8)), (IAMTP, COM(9)),                 ABCD4445
     2 (IGASMX, COM(10)), (IDBURN, COM(11)), (IAFTBN, COM(12)),          ABCD4446
     3 (IDCD, COM(13)), (IMCD, COM(14)), (NOZFLT, COM(17)),              ABCD4447
     4 (ITRYS, COM(18)), (TOLALL, COM(23)), (AM23, COM(35)),             ABCD4448
     5 (I24, COM(42)), (A28, COM(50)), (ETAD, COM(73)),                  ABCD4449
     6 (DPDUDS, COM(75)), (T2, COM(92)), (P2, COM(93)),                  ABCD4450
     7 (T4DS, COM(101)), (AM55, COM(107)), (PS55, COM(110)),             ABCD4451
     8 (TFHPDS, COM(118)), (CNHPDS, COM(119)), (ETHPDS, COM(120)),       ABCD4452
     9 (PCNFDS, COM(124)), (PRFDS, COM(125)), (ETAFDS, COM(126)),        ABCD4453
     1 (HPEXT, COM(129)), (WACCDS, COM(130)), (ZCDS, COM(133)),          ABCD4454
     2 (WAFC, COM(135)), (PCNF, COM(137)), (PCBLF, COM(138)),            ABCD4455
     3 (ZIDS, COM(146)), (PCNIDS, COM(147)), (T4, COM(156)),             ABCD4456
     4 (PCBLHP, COM(166)), (PCBLIP, COM(167)), (PCBLLP, COM(168)),       ABCD4457
     5 (PCBLDU, COM(169)), (PCBLOB, COM(170)), (PRIDS, COM(180)),        ABCD4458
     6 (ETAIDS, COM(181)), (WAICDS, COM(183)), (ETABDS, COM(184)),       ABCD4459
     7 (WFBDS, COM(185)), (ZFDS, COM(186)), (ETAR, COM(187)),            ABCD4460
     8 (WFB,COM(192)), (AM,COM(195)), (ALTP,COM(196)), (DPCODS,COM(197)),ABCD4461
     9 (A38, COM(211)), (DPWGDS, COM(235)), (DELPG, COM(236)),           ABCD4462
     1 (DELFN, COM(237)), (DELSFC, COM(238)), (CVDWNG, COM(239))         ABCD4463
      EQUIVALENCE  (CVDNOZ, COM(240)), (CVMNOZ, COM(241)),               ABCD4464
     1 (TFLPDS, COM(275)), (CNLPDS, COM(276)), (ETLPDS, COM(277)),       ABCD4465
     2 (TFIPDS, COM(278)), (CNIPDS, COM(279)), (ETIPDS, COM(280)),       ABCD4466
     3 (PRCDS, COM(297)), (ETACDS, COM(298)), (PCNC, COM(301)),          ABCD4467
     4 (PCBLC, COM(302)), (PCNCDS, COM(303)), (PCBLI, COM(304)),         ABCD4468
     5 (PCBLID, COM(305)), (WACI, COM(310)), (WACC, COM(320)),           ABCD4469
     6 (AM6, COM(327)), (A6, COM(328)), (T7DS, COM(342)),                ABCD4470
     7 (T7, COM(343)), (A8, COM(346)), (ETAADS, COM(366)),               ABCD4471
     8 (DPAFDS, COM(367)), (WFA, COM(368)), (ETAA, COM(369)),            ABCD4472
     9 (VFAN, COM(395)), (VINTC, COM(396)), (VCOMP, COM(397)),           ABCD4473
     1 (VCOMB, COM(398)), (VHPTRB, COM(399)), (VIPTRB, COM(400)),        ABCD4474
     2 (VLPTRB, COM(401)), (VAFTBN, COM(402)), (VFDUCT, COM(403)),       ABCD4475
     3 (VWDUCT, COM(404)), (XNHPDS, COM(423)), (XNIPDS, COM(424)),       ABCD4476
     4 (XNLPDS, COM(425)), (PMIHP, COM(426)), (PMIIP, COM(427)),         ABCD4477
     5 (PMILP, COM(428)), (DELT1, COM(429)), (TIMEPT(1), COM(941)),      ABCD4478
     6 (PRFNEW, COM(991)), (PRCNEW, COM(992)), (TIME, COM(993)),         ABCD4479
     7 (DT, COM(994)), (TF, COM(995)), (IPRINT, COM(996)),               ABCD4480
     8 (DTPRNT, COM(997)), (ISPOOL, COM(1044)), (KKGO, COM(1048)),       ABCD4481
     9 (ITRAN, COM(1049)), (JTRAN, COM(1050)), (NSTEP, COM(1051)),       ABCD4482
     1 (IAMTRX, COM(1054)), (SI, COM(1055)), (ERRER, COM(1056))          ABCD4483
      EQUIVALENCE  (DUMSPL, COM(1057)), (FXFN2M, COM(1058)),             ABCD4484
     1 (FXM2CP, COM(1059)), (AFTFAN, COM(1060)), (FAN, COM(1061)),       ABCD4485
```

```
      2 (WAFCDS, COM(1062))                                                 ABCD4485
       DATA  AWORD /4HPUTI, 4HN   /                                         ABCD4487
C *** IDES  =1  FOR CALCULATING DESIGN POINT                               ABCD4488
C *** MODE  =0  FOR CONSTANT T4                                            ABCD4489
C *** MODE  =1  FOR CONSTANT PCNC                                          ABCD4490
C *** MODE  =2  FOR CONSTANT WFB                                           ABCD4491
C *** MODE  =3  FOR CONSTANT PCNF                                          ABCD4492
C *** INIT  =1  WILL NOT INITIALIZE POINT                                  ABCD4493
C *** IDUMP =1  WILL DUMP LOOPING WRITE-OUTS IF ERROR OCCURS               ABCD4494
C *** IDUMP =2  WILL DUMP LOOPING WRITE-OUTS AFTER EVERY POINT             ABCD4495
C *** IAMTP =0  WILL USE INPUT AM AND MIL SPEC ETAR                        ABCD4496
C *** IAMTP =1  WILL USE INPUT AM AND INPUT ETAR                           ABCD4497
C *** IAMTP =2  WILL USE T2 AS T1=T1+T2 AND STANDARD P1                    ABCD4498
C *** IAMTP =3  WILL USE P2 AND STANDARD T1                                ABCD4499
C *** IAMTP =4  WILL USE T2 AND P2                                         ABCD4500
C *** IAMTP =5 WILL USE RAM2 FOR SPECIAL RECOVERY                          ABCD4501
C *** IGASMX=-1 SEPARATE FLOW, INPUT A6                                    ABCD4502
C *** IGASMX=0  SEPARATE FLOW, A6=A55                                      ABCD4503
C *** IGASMX=1  WILL MIX DUCT AND MAIN STREAMS, A6=A25+A55                 ABCD4504
C *** IGASMX=2  WILL MIX DUCT AND MAIN STREAMS, INPUT A6                   ABCD4505
C *** IDBURN=1  FOR DUCT BURNING, INPUT T24                                ABCD4506
C *** IDBURN=2  FOR DUCT BURNING, INPUT WFD                                ABCD4507
C *** IAFTBN=1  FOR AFTERBURNING, INPUT T7                                 ABCD4508
C *** IAFTBN=2  FOR AFTERBURNING, INPUT WFA                                ABCD4509
C *** IDCD  =1  DUCT NOZZLE WILL BE C-D                                    ABCD4510
C *** IMCD  =1  MAIN NOZZLE WILL BE C-D                                    ABCD4511
C *** NOZFLT=1  FOR FLOATING MAIN NOZZLE                                   ABCD4512
C *** NOZFLT=2  FOR FLOATING DUCT NOZZLE                                   ABCD4513
C *** NOZFLT=3  FOR FLOATING MAIN AND DUCT NOZZLES                         ABCD4514
C *** ITRYS =N  NUMBER OF PASSES THRU ENGINE BEFORE QUITTING               ABCD4515
       NAMELIST /DATAIN/  ISPOOL,FAN,SI,DELT1,IDES,MODE,IDUMP,IAMTP,       ABCD4516
      1 IGASMX,IDBURN,IAFTBN,IDCD,IMCD,NOZFLT,ITRYS,FXFN2M,FXM2CP,         ABCD4517
      2 AFTFAN,DUMSPL,TOLALL,DELFG,DELFN,DELSFC,PCNFDS,PRFDS,ETAFDS,       ABCD4518
      3 PCNCDS,PRCDS,ETACDS,T4DS,WFBDS,ETABDS,DPCODS,ETHPDS,ETLPDS,        ABCD4519
      4 DPDUDS,T7DS,ETAADS,DPAFDS,A6,A8,A28,PS55,AM55,CVDNOZ,CVMNOZ,T2,    ABCD4520
      5 P2,T4,WAFCDS,WACCDS,HPEXT,AM,ALTP,ETAR,PCNF,PCNC,WFB,PCBLF,PCNI,   ABCD4521
      6 PCBLC,PCBLDU,PCBLOB,PCBLHP,PCBLLP,T24,ETAD,T7,WFA,ETAA,AM6,AM23,   ABCD4522
      7 DPWGDS,A38,PCNIDS,PCBLIP,ZFDS,ZCDS,ZIDS,PCBLID,TFHPDS,CNHPDS,      ABCD4523
      8 TFIPDS,CNIPDS,TFLPDS,CNLPDS,PRIDS,ETAIDS,ETIPDS,WAICDS,PCBLI,      ABCD4524
      9 CVDWNG,ITRAN,DTPRNT,TF,INIT,DT,XNHPDS,XNIPDS,XNLPDS,PMIHP,PMIIP,   ABCD4525
      1 PMILP,VFAN,VINTC,VCOMP,VCOMB,VHPTRB,VIPTRB,VLPTRB,VAFTBN,VFDUCT,    ABCD4526
      2 VWDUCT,IAMTRX                                                      ABCD4527
       WORD(1) = AWORD(1)                                                  ABCD4528
       WORD(2) = AWORD(2)                                                  ABCD4529
       ITRAN = 0                                                           ABCD4530
       JTRAN = 0                                                           ABCD4531
       TIME = 0.0D0                                                        ABCD4532
       NSTEP = 0                                                           ABCD4533
       TPRINT = 0.0D0                                                      ABCD4534
       DTPRNT = 0.0D0                                                      ABCD4535
1      CALL ZERO                                                           ABCD4536
       IF (KKGO .EQ. 1)  GO TO 5                                           ABCD4537
       IDES = 0                                                            ABCD4538
       READ (5,DATAIN)                                                     ABCD4539
```

```
      IF (.NOT. ERRER)  GO TO 102                                          ABCD4540
      IF (IAFTBN .GT. 0 .OR. IDBURN .GT. 0 .OR. NOZFLT .GT. 0)  GO TO 1    ABCD4541
102   ERRER = .FALSE.                                                      ABCD4542
C     TABLE IS REFERENCED TO COMMON/ALL/FIRST ENTRY                        ABCD4543
      IF (IDES .EQ. 0)  GO TO 7                                            ABCD4544
      IF (KKGO .NE. 2)  GO TO 3                                            ABCD4545
      TEMP1 = COM(325)                                                     ABCD4546
      TEMP2 = COM(326)                                                     ABCD4547
      TEMP3 = COM(336)                                                     ABCD4548
      TEMP4 = COM(370)                                                     ABCD4549
      DO 2  I = 1,392                                                      ABCD4550
2     COM(I+2) = XSAVE(I)                                                  ABCD4551
      DO 205  I = 1,4                                                      ABCD4552
205   COM(I+1056) = XSAVE(I+392)                                           ABCD4553
      COM(325) = TEMP1                                                     ABCD4554
      COM(326) = TEMP2                                                     ABCD4555
      COM(336) = TEMP3                                                     ABCD4556
      COM(370) = TEMP4                                                     ABCD4557
      READ (5,DATAIN)                                                      ABCD4558
C     SAVE INPUT IN CASE OF LOOP ON PRESSURE RATIOS                        ABCD4559
3     TEMP1 = XSAVE(323)                                                   ABCD4560
      TEMP2 = XSAVE(324)                                                   ABCD4561
      TEMP3 = XSAVE(334)                                                   ABCD4562
      TEMP4 = XSAVE(368)                                                   ABCD4563
      DO 4  I = 1,392                                                      ABCD4564
4     XSAVE(I) = COM(I+2)                                                  ABCD4565
      DO 405  I = 1,4                                                      ABCD4566
405   XSAVE(I+392) = COM(I+1056)                                           ABCD4567
      XSAVE(323) = TEMP1                                                   ABCD4568
      XSAVE(324) = TEMP2                                                   ABCD4569
      XSAVE(334) = TEMP3                                                   ABCD4570
      XSAVE(368) = TEMP4                                                   ABCD4571
      GO TO 7                                                              ABCD4572
5     TEMP1 = COM(325)                                                     ABCD4573
      TEMP2 = COM(326)                                                     ABCD4574
      TEMP3 = COM(336)                                                     ABCD4575
      TEMP4 = COM(370)                                                     ABCD4576
      DO 6  I = 1,392                                                      ABCD4577
6     COM(I+2) = XSAVE(I)                                                  ABCD4578
      DO 605  I = 1,4                                                      ABCD4579
605   COM(I+1056) = XSAVE(I+392)                                           ABCD4580
      COM(325) = TEMP1                                                     ABCD4581
      COM(326) = TEMP2                                                     ABCD4582
      COM(336) = TEMP3                                                     ABCD4583
      COM(370) = TEMP4                                                     ABCD4584
      WRITE (6,8) PRFDS,PRFNEW,PRCDS,PRCNEW                                ABCD4585
      PRCDS = PRCNEW                                                       ABCD4586
      PRFDS = PRFNEW                                                       ABCD4587
7     KKGO = 2                                                             ABCD4588
      IF (IAFTBN .GT. 0 .OR. IDBURN .GT. 0 .OR. NOZFLT .GT. 0)  INIT = 1   ABCD4589
      IF (MODE .EQ. 0)  WRITE (6,9) IDES,AM,ALTP,T4,T24,T7                 ABCD4590
      IF (MODE .EQ. 1)  WRITE (6,10) IDES,AM,ALTP,PCNC,T24,T7              ABCD4591
      IF (MODE .EQ. 2)  WRITE (6,11) IDES,AM,ALTP,WFB,T24,T7               ABCD4592
      IF (DUMSPL)  WAICDS = WACCDS                                         ABCD4593
```

```
      IF (IDES .NE. 1)  GO TO 101                                          ABCD4594
      WAFC = WAFCDS                                                        ABCD4595
      WACI = WAICDS                                                        ABCD4596
      WACC = WACCDS                                                        ABCD4597
101   CALL COINLT                                                          ABCD4598
      RETURN                                                               ABCD4599
C                                                                          ABCD4600
C                                                                          ABCD4601
8     FORMAT (18H0CHANGE PRFDS FROM, F9.3, 4H  TO, F9.3, 17H   AND PRCDSABCD4602
     1 FROM, F10.3, 4H  TO, F10.3)                                         ABCD4603
9     FORMAT  (1H0, 7H  IDES=, I3, 10X, 7H    AM=, F7.3, 6X, 7H  ALTP=, ABCD4604
     1 F7.0, 6X, 7H    T4=, F8.2, 5X, 7H   T24=, F8.2, 5X, 7H    T7=,   ABCD4605
     2 F8.2, 6H$$$$$$)                                                     ABCD4606
10    FORMAT  (1H0, 7H  IDES=, I3, 10X, 7H    AM=, F7.3, 6X, 7H  ALTP=, ABCD4607
     1 F7.0, 6X, 7H  PCNC=, F8.3, 5X, 7H   T24=, F8.2, 5X, 7H    T7=,   ABCD4608
     2 F8.2, 6H$$$$$$)                                                     ABCD4609
11    FORMAT  (1H0, 7H  IDES=, I3, 10X, 7H    AM=, F7.3, 6X, 7H  ALTP=, ABCD4610
     1 F7.0, 6X, 7H   WFB=, F8.4, 5X, 7H   T24=, F8.2, 5X, 7H    T7=,   ABCD4611
     2 F8.2, 6H$$$$$$)                                                     ABCD4612
      END                                                                  ABCD4613
```

Subroutine RAM

```
      SUBROUTINE RAM (AM,ETAR)                                             ABCD4614
      IMPLICIT  REAL*8  (A-H,O-Z)                                          ABCD4615
      ETAR = 1.0D0                                                         ABCD4616
      IF (AM .LE. 1.0D0)  RETURN                                           ABCD4617
      IF (AM .GT. 5.0D0)  GO TO 3                                          ABCD4618
      ETAR = 1.0D0 - 0.075D0 * ((AM - 1.0D0) ** 1.35D0)                    ABCD4619
      RETURN                                                               ABCD4620
3     ETAR = 800.0D0 / ((AM ** 4) + 935.0D0)                               ABCD4621
      RETURN                                                               ABCD4622
      END                                                                  ABCD4623
```

Subroutine RAM2

```
      SUBROUTINE RAM2 (AM,ETAR)                                            ABCD4624
      IMPLICIT  REAL*8  (A-H,O-Z)                                          ABCD4625
      DIMENSION  PRINLT(15), FMN(15), X(3), Y(3)                           ABCD4626
      DATA  FMN /0.0D0, .1D0, .2D0, .3D0, .4D0, .5D0, .8D0, 1.1D0,         ABCD4627
     1 1.2D0, 1.4D0, 1.6D0, 1.8D0, 2.2D0, 2.4D0, 2.7D0/                    ABCD4628
      DATA  PRINLT /.9D0, .932D0, .95D0, .961D0, .968D0, .97D0, .9701D0,ABCD4629
     1 .97D0, .9681D0, .958D0, .94D0, .9181D0, .858D0, .8201D0, .75D0/     ABCD4630
      M = 0                                                                ABCD4631
      DO 1  J = 1,15                                                       ABCD4632
      IF (AM .GE. FMN(J))  M = J - 1                                       ABCD4633
1     CONTINUE                                                             ABCD4634
      IF (M .EQ. 0)  M = 1                                                 ABCD4635
      IF (M .GE. 14)  M = 13                                               ABCD4636
      DO 2  I = 1,3                                                        ABCD4637
      MM = M - 1 + I                                                       ABCD4638
      X(I) = FMN(MM)                                                       ABCD4639
2     Y(I) = PRINLT(MM)                                                    ABCD4640
      CALL PARABO (X,Y,AM,ETAR)                                            ABCD4641
      RETURN                                                               ABCD4642
      END                                                                  ABCD4643
```

Subroutine ROLL

```
      SUBROUTINE ROLL                                                   ABCD4644
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD4645
      COMMON /COMALL/  COM(1062)                                        ABCD4646
      DIMENSION  FO(50,4), SO(10,6), PDATA(5,50), TIMEPT(50)            ABCD4647
      EQUIVALENCE  (FO(1,1), COM(430)), (SO(1,1), COM(631)),            AECD4648
     1 (PDATA(1,1), COM(691)), (TIMEPT(1), COM(941))                    ABCD4649
      DO 1  I = 1,50                                                    ABCD4650
      FO(I,2) = FO(I,1)                                                 ABCD4651
1     FO(I,4) = FO(I,3)                                                 ABCD4652
      DO 2  I = 1,10                                                    ABCD4653
      SO(I,6) = SO(I,5)                                                 ABCD4654
      SO(I,5) = SO(I,4)                                                 ABCD4655
      SO(I,3) = SO(I,2)                                                 ABCD4656
2     SO(I,2) = SO(I,1)                                                 ABCD4657
      DO 3  I = 1,49                                                    ABCD4658
      N1 = 51 - I                                                       ABCD4659
      NO = 50 - I                                                       ABCD4660
      TIMEPT(N1) = TIMEPT(NO)                                           ABCD4661
      DO 3  J = 1,5                                                     ABCD4662
3     PDATA(J,N1) = PDATA(J,NO)                                         ABCD4663
      RETURN                                                            ABCD4664
      END                                                               ABCD4665
```

Subroutine SEARCH

```
      SUBROUTINE SEARCH (P,A,B,C,D,AX,NA,BX,CX,DX,NO,NAM,NOM,NCODE)     ABCD4666
      IMPLICIT  REAL*8  (A-H,O-Z)                                       ABCD4667
      COMMON /COMALL/  COM(1062)                                        ABCD4668
      DIMENSION  AX(NAM), BX(NAM,NOM), CX(NAM,NOM), DX(NAM,NOM),        ABCD4669
     1 NO(NAM), Q(9)                                                    ABCD4670
      EQUIVALENCE  (TOLALL, COM(23))                                    ABCD4671
C *** NEEDS SUBROUTINE AFQUIR                                           ABCD4672
C *** AX AND BX MUST BE STORED LO TO HI                                 ABCD4673
C *** P=INPUT PROPORTION BETWEEN 0.0 AND 1.0                            ABCD4674
C     IF NOT INPUT, P MUST EQUAL -1.                                    ABCD4675
C *** NCODE=00   OK                                                     ABCD4676
C     NCODE=01   A LO                                                   ABCD4677
C     NCODE=02   A HI                                                   ABCD4678
C     NCODE=07   ERROR                                                  ABCD4679
C     NCODE=10   B LO                                                   ABCD4680
C     NCODE=20   B HI                                                   ABCD4681
      EXTR(AAA,BBB,CCC) = AAA + BBB * (CCC - AAA)                       ABCD4682
      NCODE = 0                                                         ABCD4683
      C = 0.0D0                                                         ABCD4684
      D = 0.0D0                                                         ABCD4685
C *** FIND A                                                            ABCD4686
      DO 1  I = 1,NA                                                    ABCD4687
      IH = I                                                            ABCD4688
      IF (A .LT. AX(I))  GO TO 2                                        ABCD4689
```

```
1       CONTINUE                                                        ABCD4690
        IF (A .GT. AX(IH))  NCODE = 2                                   ABCD4691
        A = AX(IH)                                                      ABCD4692
        GO TO 3                                                         ABCD4693
2       IF (IH .GT. 1)  GO TO 3                                         ABCD4694
        NCODE = 1                                                       ABCD4695
        IH = 2                                                          ABCD4696
        A = AX(1)                                                       ABCD4697
3       IL = IH - 1                                                     ABCD4698
        LIMH = NO(IH)                                                   ABCD4699
        LIML = NO(IL)                                                   ABCD4700
C ***   FIND B                                                          ABCD4701
        PRM = (A - AX(IL)) / (AX(IH) - AX(IL))                          ABCD4702
        PP = P                                                          ABCD4703
        IF (P .GE. 0.0D0)  GO TO 5                                      ABCD4704
        BL = EXTR(BX(IL,1), PRM, BX(IH,1))                              ABCD4705
        BH = EXTR(BX(IL,LIML), PRM, BX(IH,LIMH))                        ABCD4706
        IF (B .GE. BL)  GO TO 4                                         ABCD4707
        NCODE = NCODE + 10                                              ABCD4708
        B = BL                                                          ABCD4709
        GO TO 5                                                         ABCD4710
4       IF (B .LE. BH)  GO TO 5                                         ABCD4711
        NCODE = NCODE + 20                                              ABCD4712
        BHM = EXTR(BX(IL,LIML-1), PRM, BX(IH,LIMH-1))                   ABCD4713
        CHM = EXTR(CX(IL,LIML-1), PRM, CX(IH,LIMH-1))                   ABCD4714
        DHM = EXTR(DX(IL,LIML-1), PRM, DX(IH,LIMH-1))                   ABCD4715
        CH = EXTR(CX(IL,LIML), PRM, CX(IH,LIMH))                        ABCD4716
        DH = EXTR(DX(IL,LIML), PRM, DX(IH,LIMH))                        ABCD4717
        CSLOPE = (CH - CHM) / (BH - BHM)                                ABCD4718
        DSLOPE = (DH - DHM) / (BH - BHM)                                ABCD4719
        C = CH + CSLOPE * (B - BH)                                      ABCD4720
        D = DH + DSLOPE * (B - BH)                                      ABCD4721
        RETURN                                                          ABCD4722
5       PP = 0.5D0                                                      ABCD4723
        Q(2) = 0.0D0                                                    ABCD4724
        Q(3) = 0.0D0                                                    ABCD4725
6       BH = EXTR(BX(IH,1), PP, BX(IH,LIMH))                           ABCD4726
        BL = EXTR(BX(IL,1), PP, BX(IL,LIML))                           ABCD4727
        DO 7  J = 2,LIMH                                                ABCD4728
        JH = J                                                          ABCD4729
        IF (BH .LT. BX(IH,J))  GO TO 8                                  ABCD4730
7       CONTINUE                                                        ABCD4731
8       JL = JH - 1                                                     ABCD4732
        DO 9  K = 2,LIML                                                ABCD4733
        KH = K                                                          ABCD4734
        IF (BL .LT. BX(IL,K))  GO TO 10                                 ABCD4735
9       CONTINUE                                                        ABCD4736
10      KL = KH - 1                                                     ABCD4737
        PR = (BX(IH,JL) - BH) / (BX(IH,JH) - BX(IH,JL))                 ABCD4738
        CH = EXTR(CX(IH,JL), -PR, CX(IH,JH))                            ABCD4739
        DH = EXTR(DX(IH,JL), -PR, DX(IH,JH))                            ABCD4740
        PR = (BX(IL,KL) - BL) / (BX(IL,KH) - BX(IL,KL))                 ABCD4741
        CL = EXTR(CX(IL,KL), -PR, CX(IL,KH))                            ABCD4742
        DL = EXTR(DX(IL,KL), -PR, DX(IH,KH))                            ABCD4743
```

```
      BT = EXTR(BL, PRM, BH)                                                    ABCD4744
      CT = EXTR(CL, PRM, CH)                                                    ABCD4745
      DT = EXTR(DL, PRM, DH)                                                    ABCD4746
      IF (P .GE. 0.0D0)  GO TO 13                                               ABCD4747
      DIR = DSQRT(B / BT)                                                       ABCD4748
      ERR = (B - BT) / B                                                        ABCD4749
      CALL AFQUIR (Q(1),PP,ERR,0.0D0,25.0D0,1.0D0*TOLALL,DIR,PT,ICON)           ABCD4750
      GO TO (11,13,12), ICON                                                    ABCD4751
11    PP = PT                                                                   ABCD4752
      IF (PP .LT. 0.0D0)  PP = 0.0D0                                            ABCD4753
      IF (PP .GT. 1.0D0)  PP = 1.0D0                                            ABCD4754
      GO TO 6                                                                   ABCD4755
12    NCODE = 7                                                                 ABCD4756
13    B = BT                                                                    ABCD4757
      C = CT                                                                    ABCD4758
      D = DT                                                                    ABCD4759
      RETURN                                                                    ABCD4760
      END                                                                       ABCD4761
```

Subroutine SYG

```
      SUBROUTINE SYG (ICON)                                                     ABCD4762
      DIMENSION  WORD(132)                                                      ABCD4763
      DATA  ONEDOL /4H$   /                                                     ABCD4764
      GO TO (1,2), ICON                                                         ABCD4765
1     REWIND 8                                                                  ABCD4766
      RETURN                                                                    ABCD4767
C        TERMINATE THE FILE                                                     ABCD4768
2     WRITE (8,10)                                                              ABCD4769
      REWIND 8                                                                  ABCD4770
C        READ RECORD                                                            ABCD4771
3     READ (8,11) (WORD(I),I=1,132)                                             ABCD4772
C        CHECK FOR 12 LEADING DOLLAR SIGNS                                      ABCD4773
      DO 4  I = 1,12                                                            ABCD4774
      IF (WORD(I) .NE. ONEDOL)  GO TO 5                                         ABCD4775
4     CONTINUE                                                                  ABCD4776
      RETURN                                                                    ABCD4777
C        CHECK FOR 6 TRAILING DOLLAR SIGNS                                      ABCD4778
5     DO 8  I = 1,132                                                           ABCD4779
      IF (WORD(I) .NE. ONEDOL)  GO TO 8                                         ABCD4780
      K = I + 5                                                                 ABCD4781
      DO 7  J = I,K                                                             ABCD4782
      IF (WORD(J) .NE. ONEDOL)  GO TO 8                                         ABCD4783
7     CONTINUE                                                                  ABCD4784
      GO TO 9                                                                   ABCD4785
8     CONTINUE                                                                  ABCD4786
      WRITE (6,12)                                                              ABCD4787
      RETURN                                                                    ABCD4788
C        PRINT LINE                                                             ABCD4789
9     I = I - 1                                                                 ABCD4790
      WRITE (6,11) (WORD(M),M=1,I)                                              ABCD4791
      GO TO 3                                                                   ABCD4792
C                                                                               ABCD4793
C                                                                               ABCD4794
```

```
10    FORMAT (12H$$$$$$$$$$$$)                                          ABCD4795
11    FORMAT (132A1)                                                     ABCD4796
12    FORMAT (1H0,12HERROR IN SYG)                                       ABCD4797
      END                                                                ABCD4798
```

Subroutine THCOMP

```
      SUBROUTINE THCOMP (PR,ETA,T,H,S,P,TO,HO,SO,PO)                     ABCD4799
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD4800
      LOGICAL  SI                                                        ABCD4801
      COMMON /COMALL/  COM(1062)                                         ABCD4802
      EQUIVALENCE  (TOLALL, COM(23)), (SI, COM(1055))                    ABCD4803
      CPG = .25D0                                                        ABCD4804
      IF (SI)  CPG = 1048.0D0                                            ABCD4805
      PO = P * PR                                                        ABCD4806
      TP = T * PR ** 0.28572D0                                           ABCD4807
      DO 1  I = 1,25                                                     ABCD4808
      CALL THERMO (PO,HP,TP,SP,X1,0,X2,0)                                ABCD4809
      DELS = SP - S                                                      ABCD4810
      IF (DABS(DELS) .LE. .05D0 * TOLALL * S)  GO TO 2                   ABCD4811
1     TP = TP / DEXP(DELS / CPG)                                         ABCD4812
      CALL ERROR                                                         ABCD4813
2     HO = H + ((HP - H) / ETA)                                          ABCD4814
      CALL THERMO (PO,HO,TO,SO,X1,0,X2,1)                                ABCD4815
      RETURN                                                             ABCD4816
      END                                                                ABCD4817
```

Subroutine THERMO

```
      SUBROUTINE THERMO (PX,HX,TX,SX,AMX,L,FAR,K)                        ABCD4818
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD4819
      LOGICAL  SI                                                        ABCD4820
      COMMON /COMALL/  COM(1062)                                         ABCD4821
      EQUIVALENCE  (TOLALL, COM(23)), (SI, COM(1055))                    ABCD4822
      IF (SI)  GO TO 100                                                 ABCD4823
      DEM = 1.986375D0                                                   ABCD4824
      CPG = .25D0                                                        ABCD4825
      PSTD = 1.0D0                                                       ABCD4826
      GO TO 101                                                          ABCD4827
100   DEM = 8316.41D0                                                    ABCD4828
      CPG = 1048.0D0                                                     ABCD4829
      PSTD = 101325.0D0                                                  ABCD4830
101   FX = 0.0D0                                                         ABCD4831
      IF (L .EQ. 1)  FX = FAR                                            ABCD4832
      IF (K .EQ. 1)  GO TO 1                                             ABCD4833
      CALL PROCOM (FX,TX,CS,AK,CP,R,PHI,HX)                              ABCD4834
      GO TO 3                                                            ABCD4835
1     TX = HX / CPG                                                      ABCD4836
      DO 2  I = 1,50                                                     ABCD4837
      CALL PROCOM (FX,TX,CS,AK,CP,P,PHI,H)                               ABCD4838
      DELH = HX - H                                                      ABCD4839
      IF (DABS(DELH) .LE. .01D0 * TOLALL * HX)  GO TO 3                  ABCD4840
2     TX = TX + DELH / CPG                                               ABCD4841
      WRITE (8,4)                                                        ABCD4842
```

```
3       SX = PHI - R * DLOG(PX / PSTD)                                  ABCD4843
        AMX = DEM / R                                                   ABCD4844
        RETURN                                                          ABCD4845
C                                                                       ABCD4846
C                                                                       ABCD4847
4       FORMAT (31H0NO CONVERGENCE IN THERMO$$$$$$)                     ABCD4848
        END                                                             ABCD4849
```

Subroutine THTURB

```
        SUBROUTINE THTURB (DH,ETA,FAR,H,S,P,TO,HO,SO,PO)                ABCD4850
        IMPLICIT  REAL*8  (A-H,O-Z)                                     ABCD4851
        LOGICAL  SI                                                     ABCD4852
        COMMON /COMALL/  COM(1062)                                      ABCD4853
        EQUIVALENCE  (TOLALL, COM(23)), (SI, COM(1055))                 ABCD4854
        DEM = 1.986375D0                                                ABCD4855
        IF (SI)  DEM = 8316.41D0                                        ABCD4856
        HO = H - DH                                                     ABCD4857
        HOP = H - DH / ETA                                              ABCD4858
        PT = P / 2.0D0                                                  ABCD4859
        DO 1  I = 1,25                                                  ABCD4860
        CALL THERMO (PT,HOP,TT,ST,AMWT,1,FAR,1)                         ABCD4861
        DELS = ST - S                                                   ABCD4862
        IF (DABS(DELS) .LE. .05D0 * TOLALL * S)  GO TO 2                ABCD4863
1       PT = P * DEXP(DELS * AMWT / DEM + DLOG(PT / P))                 ABCD4864
        CALL ERROR                                                      ABCD4865
2       PO = PT                                                         ABCD4866
        CALL THERMO (PO,HO,TO,SO,X1,1,FAR,1)                            ABCD4867
        RETURN                                                          ABCD4868
        END                                                             ABCD4869
```

Subroutine WDUCTI

```
        SUBROUTINE WDUCTI                                               ABCD4870
        IMPLICIT  REAL*8  (A-H,O-Z)                                     ABCD4871
        LOGICAL  SI                                                     ABCD4872
        COMMON /COMALL/  COM(1062)                                      ABCD4873
        DIMENSION  WORD(2), ERR(9)                                      ABCD4874
        DIMENSION  Q(9), XZERO(25), AWORD(2)                            ABCD4875
        EQUIVALENCE  (WORD(1), COM(1)), (IDES, COM(3)), (IDSHOC, COM(15)),ABCD4876
       1 (TOLALL, COM(23)), (EPR(1), COM(24)), (P1, COM(33)), (WAC,     ABCD4877
       2 COM(191)), (WG37, COM(210)), (XZERO(1), COM(210)), (A38,       ABCD4878
       3 COM(211)), (AM38, COM(212)), (V38, COM(213)), (T38, COM(214)), ABCD4879
       4 (H38, COM(215)), (P38, COM(216)), (TS38, COM(217)),            ABCD4880
       5 (PS38, COM(218)), (A39, COM(219)), (AM39, COM(220)),           ABCD4881
       6 (V39, COM(221)), (T39, COM(222)), (H39, COM(223)),             ABCD4882
       7 (P39, COM(224)), (TS39, COM(225)), (PS39, COM(226)),           ABCD4883
       8 (WA32DS, COM(227)), (DPWING, COM(228)), (BPRINT, COM(229)),    ABCD4884
       9 (DPWGDS, COM(235)), (T21, COM(263)), (H21, COM(264)),          ABCD4885
       1 (WA32, COM(271)), (PCBLID, COM(305)), (P21, COM(377)),         ABCD4886
       2 (P37, COM(393)), (U37, COM(394)), (VWDUCT, COM(404)),          ABCD4887
       3 (SI, COM(1055))                                                ABCD4888
        DATA  AWORD /4HWDUC, 4HTI  /                                    ABCD4889
        WORD(1) = AWORD(1)                                              ABCD4890
```

```
      WORD(2) = AWORD(2)                                                ABCD4891
      IF (SI)  GO TO 100                                                ABCD4892
      RA = .0252D0                                                      ABCD4893
      AJ = 2.719D0                                                      ABCD4894
      GO TO 101                                                         ABCD4895
100   RA = 286.9D0                                                      ABCD4896
      AJ = 1.0D0                                                        ABCD4897
101   IF (PCBLID .GT. 0.0D0)  GO TO 3                                   ABCD4898
      DO 1  I = 1,25                                                    ABCD4899
1     XZERO(I) = 0.0D0                                                  ABCD4900
      RETURN                                                            ABCD4901
3     P32 = P21                                                         ABCD4902
      H32 = H21                                                         ABCD4903
      T32 = T21                                                         ABCD4904
      BPRINT = WA32 / WAC                                               ABCD4905
      WA32C = WA32 * DSQRT(T32) / P32                                   ABCD4906
      IF (IDES .EQ. 1)  WA32DS = WA32C                                  ABCD4907
      DPWING = DPWGDS * WA32C / WA32DS                                  ABCD4908
      DPWING = DMIN1(1.0D0,DPWING)                                      ABCD4909
      P36 = P32 * (1.0D0 - DPWING)                                      ABCD4910
      T36 = T32                                                         ABCD4911
      H36 = H32                                                         ABCD4912
      CALL THERMO (P36,H36,T36,S36,XX2,1,0.0D0,0)                       ABCD4913
      WG37 = WA32                                                       ABCD4914
      T37 = T36                                                         ABCD4915
      P37 = P36                                                         ABCD4916
      H37 = H36                                                         ABCD4917
      S37 = S36                                                         ABCD4918
      IF (VWDUCT .EQ. 0.0D0)  GO TO 21                                  ABCD4919
      Q(2) = 0.0D0                                                      ABCD4920
      Q(3) = 0.0D0                                                      ABCD4921
      WG37P = WG37                                                      ABCD4922
      H37P = H37                                                        ABCD4923
      P37DOT = DERIV(22,P37)                                            ABCD4924
18    CALL THERMO (P37,H37,T37,S37,XX2,1,0.0D0,0)                       ABCD4925
      WG37 = WG37P - P37DOT * VWDUCT / T37 / 1.4D0 / RA                 ABCD4926
      U37 = H37 - RA * AJ * T37                                         ABCD4927
      U37DOT = DERIV(23,U37)                                            ABCD4928
      H37X = (WG37P * H37P - (WG37P - WG37) * U37 - U37DOT * P37 *      ABCD4929
     1 VWDUCT / T37 / RA) / WG37                                        ABCD4930
      ERRW = (H37 - H37X) / H37                                         ABCD4931
      DIR = DSQRT(DABS(H37 / H37X))                                     ABCD4932
      CALL AFQUIR (Q(1),T37,ERRW,0.0D0,20.0D0,.1D0*TOLALL,DIR,T37T,IGO) ABCD4933
      GO TO (19,21,20), IGO                                             ABCD4934
19    T37 = T37T                                                        ABCD4935
      GO TO 18                                                          ABCD4936
20    CALL ERROR                                                        ABCD4937
21    CALL CONVRG (T37,H37,P37,S37,0.0D0,WG37,P1,IDES,A38,P38R,T38,H38, ABCD4938
     1 P38,S38,TS38,PS38,V38,AM38,ICON)                                 ABCD4939
      GO TO (5,5,5,4), ICON                                             ABCD4940
4     CALL ERROR                                                        ABCD4941
5     T39 = T38                                                         ABCD4942
      H39 = H38                                                         ABCD4943
      P39 = P38                                                         ABCD4944
```

```
      TS39 = TS38                                                        ABCD4945
      V39 = V38                                                          ABCD4946
      AM39 = AM38                                                        ABCD4947
      A39 = A38                                                          ABCD4948
      PS39 = PS38                                                        ABCD4949
      IDSHOC = ICON + 3                                                  ABCD4950
      ERR(7) = (P38R - P38) / P38R                                       ABCD4951
      IF (IDES .EQ. 1)  WRITE (6,6) A38,AM38,A39,AM39                    ABCD4952
      RETURN                                                             ABCD4953
C                                                                        ABCD4954
C                                                                        ABCD4955
6     FORMAT (18H0INTER DUCT DESIGN,5X,8H    A38=,E15.8,8H   AM38=,     ABCD4956
     1 E15.8,8H    A39=,E15.8,8H   AM39=,E15.8)                         ABCD4957
      END                                                                ABCD4958
```

Subroutine ZERO

```
      SUBROUTINE ZERO                                                    ABCD4959
      IMPLICIT  REAL*8  (A-H,O-Z)                                        ABCD4960
      COMMON /COMALL/  COM(1062)                                         ABCD4961
      EQUIVALENCE  (IDES, COM(3)), (JDES, COM(4)), (INIT, COM(7)),       ABCD4962
     1 (IDBURN, COM(11)), (IAFTBN, COM(12)), (IDSHOC, COM(15)),          ABCD4963
     2 (IMSHOC, COM(16)), (NOZFLT, COM(17))                              ABCD4964
      IDES = 0                                                           ABCD4965
      JDES = 0                                                           ABCD4966
      INIT = 0                                                           ABCD4967
      IDBURN = 0                                                         ABCD4968
      IAFTBN = 0                                                         ABCD4969
      IDSHOC = 3                                                         ABCD4970
      IMSHOC = 3                                                         ABCD4971
      NOZFLT = 0                                                         ABCD4972
      COM(33) = 0.0D0                                                    ABCD4973
      DO 5  I = 37,90                                                    ABCD4974
      IF (I .EQ. 41 .OR. I .EQ. 50 .OR. I .EQ. 51)  GO TO 5              ABCD4975
      IF (I .EQ. 60 .OR. I .EQ. 61 .OR. I .EQ. 75)  GO TO 5              ABCD4976
      COM(I) = 0.0D0                                                     ABCD4977
5     CONTINUE                                                           ABCD4978
      COM(94) = 0.0D0                                                    ABCD4979
      COM(95) = 0.0D0                                                    ABCD4980
      COM(99) = 0.0D0                                                    ABCD4981
      DO 6  I = 102,113                                                  ABCD4982
      IF (I .EQ. 107 .OR. I .EQ. 109 .OR. I .EQ. 110)  GO TO 6           ABCD4983
      COM(I) = 0.0D0                                                     ABCD4984
6     CONTINUE                                                           ABCD4985
      DO 7  I = 131,135                                                  ABCD4986
      IF (I .EQ. 133)  GO TO 7                                           ABCD4987
      COM(I) = 0.0D0                                                     ABCD4988
7     CONTINUE                                                           ABCD4989
      DO 8  I = 149,160                                                  ABCD4990
      IF (I .EQ. 155 .OR. I .EQ. 156)  GO TO 8                           ABCD4991
      COM(I) = 0.0D0                                                     ABCD4992
8     CONTINUE                                                           ABCD4993
      DO 9  I = 171,174                                                  ABCD4994
9     COM(I) = 0.0D0                                                     ABCD4995
```

```
      DO 10  I = 190,209                                                    ABCD4996
      IF (I .EQ. 192 .OR. (I .GE. 195 .AND. I .LE. 197) .OR. I .EQ. 199)ABCD4997
     1 GO TO 10                                                           ABCD4998
      COM(I) = 0.0D0                                                      ABCD4999
10    CONTINUE                                                            ABCD5000
      DO 11  I = 242,258                                                  ABCD5001
      IF (I .EQ. 244)  GO TO 11                                           ABCD5002
      COM(I) = 0.0D0                                                      ABCD5003
11    CONTINUE                                                            ABCD5004
      DO 12  I = 263,265                                                  ABCD5005
12    COM(I) = 0.0D0                                                      ABCD5006
      DO 13  I = 270,274                                                  ABCD5007
      IF (I .EQ. 271)  GO TO 13                                           ABCD5008
      COM(I) = 0.0D0                                                      ABCD5009
13    CONTINUE                                                            ABCD5010
      DO 14  I = 290,293                                                  ABCD5011
14    COM(I) = 0.0D0                                                      ABCD5012
      DO 15  I = 306,320                                                  ABCD5013
      IF ((I .GE. 309 .AND. I .LE. 312) .OR. I .EQ. 314)  GO TO 15        ABCD5014
      COM(I) = 0.0D0                                                      ABCD5015
15    CONTINUE                                                            ABCD5016
      DO 16  I = 321,355                                                  ABCD5017
      IF (I .EQ. 325 .OR. I .EQ. 326 .OR. I .EQ. 328)  GO TO 16           ABCD5018
      IF (I .EQ. 329 .OR. I .EQ. 336 .OR. I .EQ. 341)  GO TO 16           ABCD5019
      IF (I .EQ. 342 .OR. I .EQ. 346 .OR. I .EQ. 347)  GO TO 16           ABCD5020
      COM(I) = 0.0D0                                                      ABCD5021
15    CONTINUE                                                            ABCD5022
      DO 17  I = 358,371                                                  ABCD5023
      IF (I .EQ. 366 .OR. I .EQ. 367 .OR. I .EQ. 370)  GO TO 17           ABCD5024
      COM(I) = 0.0D0                                                      ABCD5025
17    CONTINUE                                                            ABCD5026
      CALL SYG (1)                                                        ABCD5027
      RETURN                                                              ABCD5028
      END                                                                 ABCD5029
```

APPENDIX C

SAMPLE CASE INPUT AND OUTPUT

PCNF
CNF
ZF
PRF
WAFC
WAF
PCNC
CNC
ZC
PRC
WACC
WAC
T2
P2
T21
P21
T3
P3
PCBLF
BLF
PCBLC
BLC
PCBLOB
BLOB
PCBLHP
BLHP
PCBLLP
BLLP
T4
P4
WA3
WFB
WG4
FAR4
ETAB
DPCOM
TFFHP
CNHP
DHTCHP
DHTC
T5
P5
TFFLP
CNLP
DHTCLP
DHTF
T55
P55
PCBLDU
BLDU
T24
P24
T25
P25

```
WAD
WFD
WG24
FAR24
ETAD
DPDUC
ETAF
ETAC
ETATHP
ETATLP
AM55
AM25
T6
P6
PS6
AM6
V6
WG6
T7
WFA
WG7
FAR7
ETAA
DPAFT
PS8
AM8
V8
PS9
AM9
V9
PS28
AM28
V28
PS29
AM29
V29
BYPASS
HPEXT
WFT
WGT
VA
FRD
CVMNOZ
VJM
CVDNOZ
VJD
FGM
FGP
THEEND
 $DATAIN ISPOOL=2,FAN=.TRUE.,SI=.FALSE.,IDES=1,MODE=0,IDUMP=1,IAMTP=0,
 IGASMX=2,ITRYS=200,FXFN2M=.FALSE.,FXM2CP=.FALSE.,AFTFAN=.FALSE.,
 DUMSPL=.TRUE.,TOLALL=.0001 ,DELFG=1.0,DELFN=1.0,DELSFC=1.0,PCNFDS=102.31,
 PRFDS=2.996,ETAFDS=.8499,PCNCDS=98.73,PRCDS=8.462,ETACDS=.8136,T4DS=2892.04,
 ETABDS=1.00,DPCODS=.0561,ETHPDS=.8713,ETLPDS=.9021,DPDUDS=.0584,T7DS=3583.6,
```

```
ETAADS=.8430,DPAFDS=.0599,AM55=.283,AM6=.243,CVMNOZ=.9494,WAFCDS=221.573,
WACCDS=54.988,HPEXT=0.0,AM=0.0,ALTP=0.0,PCBLF=0.0,PCBLC=.16,PCBLDU=.208,
PCBLOB=0.0,PCBLHP=.726,PCBLLP=.066,AM23=.170,ZFDS=.8333,ZCDS=.8143,
TFHPDS=50.0,CNHPDS=2.0,TFLPDS=130.0,CNLPDS=2.3,XNHPDS=10070.,XNLPDS=9651.,
PMIHP=3.80,PMILP=4.50,VFAN=2.31,VCOMP=1.65,VCOMB=1.65,VHPTRB=.505,
VLPTRB=.618,VAFTBN=49.77,VFDUCT=10.08  &END
&DATAIN  MODE=2,WFB=2.75,ITRAN=0,IDES=0  &END
&DATAIN  MODE=2,WFB=2.75,ITRAN=1,IAMTRX=1  &END
```

FAN DESIGN PRFCP= 0.47523476E 01 ETAFCP= 0.98548257E 00 WAFCP= 0.35527831E 00 T2DS= 0.51855992E 03

MIDDLE SPOOL DESIGN PRICP= 0.10000000E 01 ETAICP= 0.10000000E 01 WAICP= 0.10000000E 01 T22DS= 0.74235402E 03

COMPRESSOR DESIGN PRCCP= 0.10922125E 01 ETACCP= 0.95051735E 00 WACCP= 0.56173771E 00 T21DS= 0.74236402E 03

COMBUSTOR DESIGN WA3CDS= 0.11338571E 02 ETABCP= 0.10000000E 01

H.P. TURBINE DESIGN CNHPCP= 0.91021469E 00 TFHPCP= 0.27621708E 01 ETHPCP= 0.96811114E 00 DHHPCP= 0.12238288E 01

L.P. TURBINE DESIGN CNLPCP= 0.10310318E 01 TFLPCP= 0.17567140E 01 ETLPCP= 0.10225143E 01 DHLPCP= 0.24622325E 01

TURBINE/DUCT AREA DESIGN A55= 0.41817585E 01 AM55= 0.29335887E 00 A25= 0.12855694E 01 AM25= 0.38353284E 00
AFTERBURNER DESIGN AREA A6 6.943

NOZZLE DESIGN A8= 0.29487834E 01 AM8= 0.10000000E 01 A9= 0.29487884E 01 AM9= 0.10000000E 01
OUTPUT AM= 0.000 ALTP= 0. T4= 2892.04 ETAR= 1.0000

MIDDLE SPOOL IS DUMMY

NO AIRFLOW INTO WING
THE OUTPUT IS IN ENGLISH UNITS

PCNF	CNF	ZF	PRF	WAFC	WAF
0.102310E 03	0.102310E 01	0.833300E 00	0.299600E 01	0.221573E 03	0.221573E 03
PCNC	CNC	ZC	PRC	WACC	WAC
0.119165E 03	0.997300E 00	0.814300E 00	0.846200E 01	0.549898E 02	0.137543E 03
T2	P2	T21	P21	T3	P3
0.518670E 03	0.100000E 01	0.742964E 03	0.299600E 01	0.145753E 04	0.253522E 02
PCBLF	BLF	PCBLC	BLC	PCBLDB	BLDB
0.000000	0.000000	0.160000E 00	0.220237E 02	0.000000	0.000000
PCBLHP	BLHP	PCBLLP	BLLP	T4	P4
0.726000E 00	0.159892E 02	0.660000E-01	0.145357E 01	0.289204E 04	0.239299E 02
WA3	WFB	WG4	FAR4	ETAB	DPCOM
0.115624E 03	0.274976E 01	0.118374E 03	0.237819E-01	0.100000E 01	0.561000E-01
TFFHP	CNHP	DHTCHP	DHTC	T5	P5
0.500000E 02	0.200000E 01	0.734297E-01	0.212362E 03	0.210342E 04	0.566634E 01
TFFLP	CNLP	DHTCLP	DHTF	T55	P55
0.130000E 03	0.230000E 01	0.422988E-01	0.889511E 02	0.178909E 04	0.259784E 01
PCBLDU	BLDU	T24	P24	T25	P25
0.208000E 00	0.458093E 01	0.781926E 03	0.232103E 01	0.781926E 03	0.232103E 01
WAD	WFD	WG24	FAR24	ETAD	DPDUC
0.885053E 02	0.000000	0.885053E 02	0.000000	0.000000	0.594000E-01
ETAF	ETAC	ETAFHP	ETAFLP	AM55	AM25
0.849900E 00	0.813600E 00	0.871300E 00	0.902100E 00	0.293359E 00	0.383683E 00
T6	P6	PS6	AM6	V6	WG6
0.141349E 04	0.271359E 01	0.260828E 01	0.243000E 00	0.437208E 03	0.224322E 03

T7	WFA	WG7	FAR7	EFAA	DPAFT
0.141349E 04	0.000000	0.224322E 03	0.124102E-01	0.000000	0.599000E-01
PS8	AM8	V8	PS9	AM9	V9
0.137620E 01	0.100000E 01	0.167416E 04	0.137620E 01	0.100000E 01	0.167416E 04
PS28	AM28	V28	PS29	AM29	V29
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
BYPASS	HPEXT	WFT	WGT	VA	FRD
0.609702E 00	0.000000	0.274975E 01	0.224322E 03	0.000000	0.000000
CVMNOZ	VJM	CVDNOZ	VJD	FGM	FGP
0.949400E 00	0.158944E 04	0.000000	0.000000	0.110319E 05	0.234760E 04

MAIN SONIC CONVERGENT NOZZLE FG= 13429.44 FN= 13429.44 SFC= 0.73712

CONVERGED AFTER 1 LOOPS

COMMON	0.833300E 00	0.102310E 03	0.571429E 00	0.100000E 01	0.814300E 00	0.118165E 03	0.239204E 0
0.100000E 01	0.177859E 03	0.170000E 00	0.826058E 03	0.781926E 03	0.299600E 01	0.187312E 03	0.161458E 0
0.254881E 01	0.781926E 03	0.187311E 03	0.161871E 01	0.383683E 00	0.781926E 03	0.282103E 01	0.187311E 0
0.161871E 01	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
0.000000	0.000000	0.000000	0.000000	0.000000	0.509702E 00	0.885053E 02	0.00000
0.000000	0.584000E-01	0.534000E-01	0.100000E 01	0.885053E 02	0.000000	0.781925E 03	0.282103E 0
0.187311E 03	0.161871E 01	0.135817E 03	0.206645E-01	0.178909E 04	0.258794E 01	0.457067E 03	0.184187E 0
0.274976E 01	0.000000	0.518670E 03	0.518670E 03	0.100000E 01	0.123918E 03	0.159103E 01	0.160218E 0
0.742964E 03	0.128557E 01	0.517369E 03	0.289204E 04	0.289204E 04	0.210342E 04	0.547064E 03	0.183705E 0
0.134363E 03	0.208927E-01	0.283359E 00	0.567671E 03	0.418175E 01	0.254984E 01	0.177375E 01	0.260828E 0
0.437208E 03	0.276217E 01	0.910215E 00	0.968111E 00	0.122383E 01	0.500000E 02	0.200000E 01	0.871300E 0
0.475235E 01	0.985483E 00	0.356278E 00	0.102310E 03	0.299600E 01	0.849900E 00	0.221573E 03	0.00000
0.000000	0.549880E 02	0.299600E 01	0.849900E 00	0.814300E 00	0.102310E 01	0.221573E 03	0.833300E 0
0.102310E 03	0.000000	0.571429E 00	0.100000E 01	0.000000	0.000000	0.000000	0.00000
0.000000	0.750000E 00	0.100000E 03	0.000000	0.518670E 03	0.123918E 03	0.159103E 01	0.146753E 0
0.360485E 03	0.115624E 03	0.113895E 02	0.289204E 04	0.784627E 03	0.183459E 01	0.118374E 03	0.237313E-0
0.210342E 04	0.547064E 03	0.183705E 01	0.134363E 03	0.208927E-01	0.726000E 00	0.000000	0.550000E-0
0.208000E 00	0.000000	0.200000E 01	0.871300E 00	0.734297E-01	0.212352E 03	0.500000E 02	0.100000E 0
0.100000E 01	0.100000E 01	0.100000E 01	0.150000E 01	0.100000E 01	0.137648E 03	0.549880E 02	0.100000E 0
0.274976E 01	0.833300E 00	0.100000E 01	0.100000E 01	0.100000E 01	0.100000E 01	0.137648E 03	0.274976E 0
0.220237E 02	0.111699E 04	0.000000	0.000000	0.561000E-01	0.561000E-01	0.102310E 03	0.100000E 0
0.742964E 03	0.299600E 01	0.177859E 03	0.160218E 01	0.360485E 03	0.221573E 03	0.137648E 03	0.00000
0.458093E 01	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00000
0.134294E 05	0.000000	0.000000	0.100000E 01	0.100000E 01	0.100000E 01	0.000000	0.00000
0.949400E 00	0.000000	0.000000	0.000000	0.158944E 04	0.274975E 01	0.224322E 03	0.737123E 0
0.124102E-01	0.000000	0.000000	0.110818E 05	0.000000	0.234760E 04	0.110819E 05	0.234760E 0
0.134294E 05	0.134294E 05	0.000000	0.100000E 01	0.100000E 01	0.100000E 01	0.742954E 03	0.177859E 0
0.160218E 01	0.137648E 03	0.742964E 03	0.742964E 03	0.137648E 03	0.152547E 01	0.000000	0.178909E 0
0.457067E 03	0.184187E 01	0.130000E 03	0.230000E 01	0.902100E 00	0.000000	0.000000	0.00000
0.175671E 01	0.103103E 01	0.102251E 01	0.246223E 01	0.000000	0.000000	0.000000	0.00000
0.130000E 03	0.230000E 01	0.902100E 00	0.422888E-01	0.889511E 02	0.109221E 01	0.950517E 00	0.561738E 0
0.846200E 01	0.813600E 00	0.137648E 03	0.814300E 00	0.118165E 03	0.160000E 00	0.987300E 02	0.00000
0.000000	0.987300E 00	0.846200E 01	0.813600E 00	0.000000	0.549880E 02	0.137648E 03	0.00000
0.159892E 02	0.000000	0.145357E 01	0.000000	0.458093E 01	0.000000	0.221573E 03	0.549880E 0
0.885053E 02	0.000000	0.135817E 03	0.206645E-01	0.398790E 02	0.243000E 00	0.243000E 00	0.594234E 0
0.310795E 04	0.224322E 03	0.141349E 04	0.271359E 01	0.350635E 03	0.224322E 03	0.124102E-01	0.463560E-0
0.139714E 04	0.243842E 01	0.455909E 03	0.259681E 00	0.694294E 01	0.358360E 04	0.141349E 04	0.350536E 0
0.177798E 01	0.294879E 01	0.000000	0.120059E 04	0.137620E 01	0.157415E 04	0.100000E 01	0.141349E 0
0.255105E 01	0.350636E 03	0.177798E 01	0.294879E 01	0.000000	0.120059E 04	0.137620E 01	0.167416E 0
0.100000E 01	0.141349E 04	0.255105E 01	0.350636E 03	0.177798E 01	0.843000E 00	0.599000E-01	0.00000
0.000000	0.843000E 00	0.599000E-01	0.118992E 05	0.000000	0.997395E 04	0.299600E 01	0.126952E 0
0.299600E 01	0.000000	0.253522E 02	0.259932E 03	0.239299E 02	0.586467E 03	0.566534E 01	0.402940E 0
0.566534E 01	0.000000	0.268794E 01	0.334481E 03	0.255105E 01	0.253785E 03	0.282103E 01	0.133735E 0
0.000000	0.000000						
0.514756E-84	0.000000	0.000000	0.000000	0.514756E-84			

OUTPUT AM= 0.000 ALTP= 0. T4= 2892.13 ETAR= 1.0000

MIDDLE SPOOL IS DUMMY

NO AIRFLOW INTO WING
THE OUTPUT IS IN ENGLISH UNITS

PCNF	CNF	ZF	PRF	WAFC	WAF
0.102310E 03	0.102310E 01	0.833378E 00	0.299619E 01	0.221553E 03	0.221553E 03
PCNC	CNC	ZC	PRC	WACC	WAC
0.118165E 03	0.987289E 00	0.814304E 00	0.846187E 01	0.549870E 02	0.137553E 03
T2	P2	T21	P21	T3	P3
0.518670E 03	0.100000E 01	0.742981E 03	0.299619E 01	0.145756E 04	0.253533E 02
PCBLF	BLF	PCBLC	BLC	PCBLDB	BLDB
0.000000	0.000000	0.160000E 00	0.220244E 02	0.000000	0.000000
PCBLHP	BLHP	PCBLLP	BLLP	T4	P4
0.725000E 00	0.159897E 02	0.660000E-01	0.145361E 01	0.289213E 04	0.239310E 02
WA3	WFB	WG4	FAR4	ETAB	DPCOM
0.115628E 03	0.275000E 01	0.118378E 03	0.237831E-01	0.100000E 01	0.560995E-01
TFFHP	CNHP	DHTCHP	DHTC	T5	P5
0.500001E 02	0.199997E 01	0.734333E-01	0.212365E 03	0.210350E 04	0.566668E 01
TFFLP	CNLP	DHTCLP	DHTF	T55	P55
0.130000E 03	0.229996E 01	0.422881E-01	0.889531E 02	0.179315E 04	0.258813E 01
PCBLDU	BLDU	T24	P24	T25	P25
0.200000E 00	0.458108E 01	0.781949E 03	0.282123E 01	0.781949E 03	0.282123E 01
WAD	WFD	WG24	FAR24	ETAD	DPDUC
0.884967E 02	0.000000	0.884967E 02	0.000000	0.000000	0.583915E-01
ETAF	ETAC	ETAFHP	ETATLP	AM55	AM25
0.849891E 00	0.813597E 00	0.871292E 00	0.902100E 00	0.283297E 00	0.383563E 00
T6	P6	PS6	AM6	V6	W36
0.141358E 04	0.271363E 01	0.260814E 01	0.243014E 00	0.437243E 03	0.224318E 03
T7	WFA	WG7	FAR7	ETAA	DPAFT
0.141358E 04	0.000000	0.224318E 03	0.124115E-01	0.000000	0.598998E-01
PS8	AM8	V8	PS9	AM9	V9
0.137622E 01	0.100000E 01	0.167421E 04	0.137622E 01	0.100000E 01	0.167421E 04
PS28	AM28	V28	PS29	AM29	V29
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
BYPASS	HPEXT	WFT	WGT	VA	FRD
0.609619E 00	0.000000	0.275000E 01	0.224318E 03	0.000000	0.000000
CVMNOZ	VJM	CVDNOZ	VJD	FGM	FGP
0.949400E 00	0.158949E 04	0.000000	0.000000	0.110320E 05	0.234772E 04

MAIN SONIC CONVERGENT NOZZLE FG= 13429.68 FN= 13429.68 SFC= 0.73717

CONVERGED AFTER 44 LOOPS

```
 THIS IS THE S.S. SOLUTION
 1.1899E 04  9.8740E 03  1.3374E 02  2.5511E 00  2.5380E 02  2.5994E 02  2.5353E 01  5.8549E 02  2.8212E 00  1.2695E 02
 4.0296E 02  3.3450E 02  5.6667E 00  2.3931E 01  2.9962E 00  2.6881E 00
    1   THE VARIABLE IS NO.    1
 THIS COLUMN OF X =
-7.3171E 01 -3.0739E 01 -3.2766E-01 -1.1803E-02  4.1193E-01 -6.2664E-01 -2.4549E-01  3.1082E 00 -9.3225E-03 -2.3577E-01
 2.2132E 00  2.1314E 00 -4.8520E-02 -2.2784E-01 -8.9229E-03 -1.3619E-02
IGIN =     1
THE NEXT TRY FOR PVRDOT =   2.1000E-02
    1   THE VARIABLE IS NO.    1
 THIS COLUMN OF X =
-7.6818E 01 -3.2274E 01 -3.4394E-01 -1.2393E-02  4.3283E-01 -6.5794E-01 -2.5772E-01  3.2555E 00 -9.7893E-03 -2.1706E-01
 2.3252E 00  2.2392E 00 -5.0937E-02 -2.3920E-01 -9.3700E-03 -1.4299E-02
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.4505E-03
    1   THE VARIABLE IS NO.    1
 THIS COLUMN OF X =
-2.3305E 01 -9.7902E 00 -1.0484E-01 -3.8113E-03  1.2784E-01 -2.0025E-01 -7.8123E-02  9.9134E-01 -2.9705E-03 -5.5122E-02
 6.9470E-01  6.6957E-01 -1.5597E-02 -7.2506E-02 -2.8418E-03 -4.4042E-03
IGIN =     3
THE NEXT TRY FOR PVRDOT =   6.5946E-03
    1   THE VARIABLE IS NO.    1
 THIS COLUMN OF X =
-2.3825E 01 -1.0018E 01 -1.0736E-01 -3.9733E-03  1.3139E-01 -2.0512E-01 -7.9945E-02  1.0038E 00 -3.0451E-03 -6.7770E-02
 7.1127E-01  6.8585E-01 -1.5908E-02 -7.4196E-02 -2.9137E-03 -4.4701E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   6.5946E-03
    2   THE VARIABLE IS NO.    3
 THIS COLUMN OF X =
 1.8782E 01 -2.0343E 01 -1.6850E-01 -7.1504E-03  7.8330E-02 -1.8911E-01  1.3976E-02 -4.7353E-01 -1.0910E-02 -2.3541E-01
-4.9248E-01 -5.1828E-01 -8.0747E-03  1.3228E-02 -1.2515E-02 -6.3936E-03
IGIN =     1
THE NEXT TRY FOR PVRDOT =   2.1000E-02
    2   THE VARIABLE IS NO.    3
 THIS COLUMN OF X =
 1.9717E 01 -2.1369E 01 -1.7688E-01 -7.5106E-03  8.2510E-02 -1.9845E-01  1.4638E-02 -4.9554E-01 -1.1457E-02 -2.4718E-01
-5.1652E-01 -5.4375E-01 -8.4892E-03  1.3855E-02 -1.3142E-02 -6.7167E-03
IGIN =     2
THE NEXT TRY FOR PVRDOT =   1.9419E-02
    2   THE VARIABLE IS NO.    3
 THIS COLUMN OF X =
 1.9239E 01 -1.9748E 01 -1.6362E-01 -6.9412E-03  7.5908E-02 -1.8368E-01  1.3591E-02 -4.6017E-01 -1.0592E-02 -2.2959E-01
-4.7846E-01 -5.0350E-01 -7.8342E-03  1.2864E-02 -1.2151E-02 -6.2061E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   1.9419E-02
```

```
    3   THE VARIABLE IS NO.  21
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.1338E-02 -3.0195E-06 -1.2791E-02 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-15 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
    3   THE VARIABLE IS NO.  21
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -4.7036E-02 -4.5319E-06 -1.9198E-02 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-15 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
    3   THE VARIABLE IS NO.  21
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -9.4153E-02 -9.0710E-06 -3.8429E-02 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-15 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     3
THE NEXT TRY FOR PVRDOT =   1.2004E-01
    3   THE VARIABLE IS NO.  21
 THIS COLUMN OF X =
 4.0185E-01  3.9502E-01 -1.9268E-01 -2.1106E-04 -9.3159E-02 -5.5546E-07 -2.0481E-04  6.4551E-03 -3.2376E-04 -2.2589E-03
 3.1548E-03 -7.5327E-04 -6.1424E-05 -1.8479E-04 -2.7794E-04 -1.4439E-04
IGIN =     4
THE NEXT TRY FOR PVRDOT =   1.6561E-01
    3   THE VARIABLE IS NO.  21
 THIS COLUMN OF X =
 5.3336E-01  5.4212E-01 -2.6585E-01 -3.0777E-04 -1.2829E-01  4.2093E-04 -3.7640E-04  1.1053E-02 -4.4530E-04 -2.7974E-03
 6.1796E-03  6.3404E-06 -9.8338E-05 -3.4170E-04 -3.8156E-04 -2.2155E-04
IGIN =     5
THE NEXT TRY FOR PVRDOT =   1.6662E-01
    3   THE VARIABLE IS NO.  21
 THIS COLUMN OF X =
 5.3630E-01  5.4541E-01 -2.6748E-01 -3.0397E-04 -1.2907E-01  4.3032E-04 -3.8023E-04  1.1155E-02 -4.4802E-04 -2.8095E-03
 6.2472E-03  2.3337E-05 -9.9260E-05 -3.4521E-04 -3.8387E-04 -2.2332E-04
IGIN =     0
THE NEXT TRY FOR PVRDOT =   1.6662E-01
    4   THE VARIABLE IS NO.  18
 THIS COLUMN OF X =
 4.8549E-01  5.8564E-01 -8.0090E-04 -3.0337E-04 -4.2444E-03  4.5776E-03  6.9387E-04 -3.9955E-03 -1.4725E-04  5.3672E-04
-7.1111E-03 -1.4450E-02 -1.4293E-04  5.4181E-04 -1.0061E-04 -3.3633E-04
IGIN =     1
THE NEXT TRY FOR PVRDOT =   2.1000E-02
    4   THE VARIABLE IS NO.  18
 THIS COLUMN OF X =
 5.0691E-01  6.1446E-01 -8.0422E-04 -3.2066E-04 -4.4055E-03  4.8619E-03  7.1635E-04 -3.9158E-03 -1.5436E-04  7.1102E-04
-7.2232E-03 -1.5036E-02 -1.5185E-04  5.6260E-04 -1.0532E-04 -3.5605E-04
IGIN =     2
THE NEXT TRY FOR PVRDOT =   2.9749E-01
```

```
    4   THE VARIABLE IS NO.  18
 THIS COLUMN OF X =
 7.4853E 00  7.6371E 00 -1.1174E-02 -4.6989E-03 -2.9016E-02  7.0791E-02  7.0036E-03 -1.1935E-02 -2.7414E-03  3.8188E-03
-6.7440E-02 -1.7154E-01 -3.2351E-03  6.4545E-03 -2.1053E-03 -5.1110E-03
IGIN =     3
THE NEXT TRY FOR PVRDOT =   3.2296E-01
    4   THE VARIABLE IS NO.  18
 THIS COLUMN OF X =
 8.1189E 00  8.2885E 00 -1.2105E-02 -5.1039E-03 -3.1395E-02  7.6857E-02  7.5681E-03 -1.2495E-02 -2.9770E-03  9.6040E-03
-7.2774E-02 -1.8588E-01 -3.5186E-03  6.9745E-03 -2.2865E-03 -5.5522E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   3.2296E-01
    5   THE VARIABLE IS NO.  19
 THIS COLUMN OF X =
 2.9764E-01  3.3295E-01 -7.7204E-04 -1.5171E-04 -7.1944E-02  2.0837E-03  4.9677E-04 -4.6943E-03 -9.5005E-05 -1.5115E-05
-6.0843E-03 -9.3137E-03 -6.4744E-05  4.5946E-04 -5.9319E-05 -1.5342E-04
IGIN =     1
THE NEXT TRY FOR PVRDOT =   2.1000E-02
    5   THE VARIABLE IS NO.  19
 THIS COLUMN OF X =
 3.0968E-01  3.4915E-01 -7.7388E-04 -1.5143E-04 -7.5492E-02  2.2435E-03  5.0940E-04 -4.8895E-03 -8.8995E-05  2.6675E-05
-6.1501E-03 -9.6429E-03 -6.9755E-05  4.7114E-04 -6.1965E-05 -1.7450E-04
IGIN =     2
THE NEXT TRY FOR PVRDOT =   1.4280E-01
    5   THE VARIABLE IS NO.  19
 THIS COLUMN OF X =
 2.0673E 00  2.0963E 00 -3.3099E-03 -1.2482E-03 -5.0414E-01  1.8802E-02  2.1195E-03 -7.6445E-03 -7.3376E-04  2.0611E-03
-2.2128E-02 -4.9459E-02 -8.2344E-04  1.9540E-03 -5.6105E-04 -1.3538E-03
IGIN =     3
THE NEXT TRY FOR PVRDOT =   1.4378E-01
    5   THE VARIABLE IS NO.  19
 THIS COLUMN OF X =
 2.0806E 00  2.1100E 00 -3.3360E-03 -1.2569E-03 -5.0760E-01  1.8923E-02  2.1286E-03 -7.6311E-03 -7.3916E-04  2.0739E-03
-2.2217E-02 -4.9741E-02 -8.2946E-04  1.9523E-03 -5.6526E-04 -1.3631E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   1.4378E-01
    6   THE VARIABLE IS NO.   9
 THIS COLUMN OF X =
 1.8406E 00  6.6172E-01  5.1501E-03  2.4570E-04 -1.9900E-02 -1.5722E-02  5.1967E-03 -1.0336E-01  1.3953E-04  3.5769E-03
-7.2034E-02 -6.7192E-02  1.1057E-03  4.8496E-03  1.2492E-04  2.9665E-04
IGIN =     1
THE NEXT TRY FOR PVRDOT =   2.1000E-02
    6   THE VARIABLE IS NO.   9
 THIS COLUMN OF X =
 1.9285E 00  6.9367E-01  5.4355E-03  2.5570E-04 -2.0846E-02 -1.6475E-02  5.4387E-03 -1.0976E-01  1.4668E-04  3.7921E-03
-7.5376E-02 -7.0390E-02  1.1584E-03  5.0756E-03  1.3121E-04  3.0843E-04
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.9021E-01
```

```
    6   THE VARIABLE IS NO.    9
 THIS COLUMN OF X =
 5.4457E 01  2.2088E 01  1.9765E-01  5.8094E-03 -6.5715E-01 -5.1162E-01  1.7537E-01 -3.4958E 00  4.3349E-03  1.4168E-01
-2.4174E 00 -2.2798E 00  3.5173E-02  1.6364E-01  3.8307E-03  8.1555E-03
IGIN =      3
THE NEXT TRY FOR PVRDOT =    7.0137E-01
    6   THE VARIABLE IS NO.    9
 THIS COLUMN OF X =
 6.5490E 01  2.2441E 01  2.0082E-01  5.9177E-03 -6.6760E-01 -5.1973E-01  1.7817E-01 -3.5513E 00  4.4647E-03  1.4395E-01
-2.4557E 00 -2.3159E 00  3.5735E-02  1.6625E-01  3.8914E-03  8.2852E-03
IGIN =      0
THE NEXT TRY FOR PVRDOT =    7.0137E-01
    7   THE VARIABLE IS NO.    8
 THIS COLUMN OF X =
-7.0654E-01 -6.1130E-01 -6.3327E-03 -2.3381E-04  1.0078E-02 -6.7603E-03 -4.7135E-03  7.1450E-02 -1.3711E-04 -4.1435E-03
 4.7439E-02  4.5692E-02 -9.8905E-04 -4.3726E-03 -1.7809E-04 -2.6809E-04
IGIN =      1
THE NEXT TRY FOR PVRDOT =    2.1000E-02
    7   THE VARIABLE IS NO.    8
 THIS COLUMN OF X =
-7.4542E-01 -6.4280E-01 -6.6548E-03 -2.4621E-04  1.0595E-02 -7.1215E-03 -4.9591E-03  7.5125E-02 -1.3642E-04 -4.3555E-03
 4.9915E-02  4.8059E-02 -1.0403E-03 -4.6004E-03 -1.8691E-04 -2.8247E-04
IGIN =      2
THE NEXT TRY FOR PVRDOT =    2.0730E-01
    7   THE VARIABLE IS NO.    8
 THIS COLUMN OF X =
-9.3066E 00 -7.1411E 00 -7.2577E-02 -2.7067E-03  1.1190E-01 -8.9587E-02 -5.5507E-02  7.8255E-01 -2.3945E-03 -4.5758E-02
 5.4264E-01  5.2051E-01 -1.1326E-02 -5.1507E-02 -1.9949E-03 -3.1325E-03
IGIN =      3
THE NEXT TRY FOR PVRDOT =    1.8961E-01
    7   THE VARIABLE IS NO.    8
 THIS COLUMN OF X =
-8.5012E 00 -6.5277E 00 -6.6355E-02 -2.4745E-03  1.0225E-01 -8.1834E-02 -5.0733E-02  7.1534E-01 -1.9152E-03 -4.2756E-02
 4.9582E-01  4.7563E-01 -1.0356E-02 -4.7077E-02 -1.8240E-03 -2.8636E-03
IGIN =      0
THE NEXT TRY FOR PVRDOT =    1.8961E-01
    8   THE VARIABLE IS NO.   11
 THIS COLUMN OF X =
-3.5174E-01 -2.9757E-01 -4.4384E-03 -1.5810E-04 -1.3499E-02 -9.3136E-03 -2.2992E-03 -2.7042E-02 -1.7015E-04 -3.6535E-03
-1.8618E-02 -1.3897E-02 -4.7123E-04 -2.1765E-03 -1.7268E-04 -1.7090E-04
IGIN =      1
THE NEXT TRY FOR PVRDOT =    2.1000E-02
    8   THE VARIABLE IS NO.   11
 THIS COLUMN OF X =
-3.8259E-01 -3.1296E-01 -4.6602E-03 -1.5704E-04 -1.4181E-02 -9.7882E-03 -2.4205E-03 -2.8349E-02 -1.7843E-04 -3.8476E-03
-1.9495E-02 -1.4561E-02 -4.9692E-04 -2.2912E-03 -1.8106E-04 -1.8089E-04
IGIN =      2
THE NEXT TRY FOR PVRDOT =    8.9727E-01
```

```
    8   THE VARIABLE IS NO.  11
 THIS COLUMN OF X =
-1.9214E 01 -1.4494E 01 -1.9529E-01 -3.2114E-03 -5.8079E-01 -4.2135E-01 -1.1405E-01 -1.0868E 00 -7.7539E-03 -1.5599E-01
-7.1142E-01 -5.3335E-01 -2.3360E-02 -1.0775E-01 -7.8561E-03 -9.1385E-03
ISIN =     3
THE NEXT TRY FOR PVRDOT =   9.6865E-01
    8   THE VARIABLE IS NO.  11
 THIS COLUMN OF X =
-2.0750E 01 -1.5653E 01 -2.1079E-01 -3.8548E-03 -6.2669E-01 -4.5485E-01 -1.2315E-01 -1.1725E 00 -8.3774E-03 -1.6826E-01
-7.6735E-01 -5.7521E-01 -2.5219E-02 -1.1635E-01 -8.4824E-03 -9.8655E-03
ISIN =     0
THE NEXT TRY FOR PVRDOT =   9.6855E-01
    9   THE VARIABLE IS NO.  20
 THIS COLUMN OF X =
     0.0000 -1.8190E-12  8.9617E-03  8.5175E-06  1.4912E-02 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
ISIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
    9   THE VARIABLE IS NO.  20
 THIS COLUMN OF X =
     0.0000 -3.6380E-12  9.2855E-03  3.7789E-05  1.9986E-02 -5.2849E-03 -1.1735E-03  7.9125E-03 -2.1499E-04 -3.1190E-03
 8.4588E-03  9.9725E-03  3.5079E-07 -1.1116E-03 -1.9962E-04  9.7371E-05
ISIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
    9   THE VARIABLE IS NO.  20
 THIS COLUMN OF X =
 5.4729E-01  6.0113E-01  2.3183E-02 -2.9048E-04  2.1840E-02  2.3942E-03  5.6384E-05  5.5545E-03 -3.1451E-04 -1.4056E-03
 1.0454E-03 -6.2582E-03 -1.2821E-04  5.6207E-05 -2.8370E-04 -2.9365E-04
ISIN =     3
THE NEXT TRY FOR PVRDOT =   1.2866E 00
    9   THE VARIABLE IS NO.  20
 THIS COLUMN OF X =
 1.1725E 01  9.8375E 00  4.7984E-01 -5.2927E-03  5.4140E-01  4.7602E-02 -7.9595E-03  2.1043E-01 -8.3429E-03 -4.7408E-02
 8.7273E-02 -2.9457E-02 -5.8257E-03 -7.3874E-03 -7.7876E-03 -5.9902E-03
ISIN =     4
THE NEXT TRY FOR PVRDOT =   8.1576E-01
    9   THE VARIABLE IS NO.  20
 THIS COLUMN OF X =
 7.5117E 00  6.4114E 00  3.0664E-01 -4.0748E-03  3.3765E-01  3.2000E-02 -4.2747E-03  1.2143E-01 -5.5039E-03 -2.8533E-02
 4.4781E-02 -3.1347E-02 -3.6815E-03 -3.9566E-03 -4.8658E-03 -3.9246E-03
ISIN =     5
THE NEXT TRY FOR PVRDOT =   8.2120E-01
    9   THE VARIABLE IS NO.  20
 THIS COLUMN OF X =
 7.6608E 00  6.4536E 00  3.0867E-01 -4.1022E-03  3.3991E-01  3.2221E-02 -4.3089E-03  1.2234E-01 -5.5410E-03 -2.8775E-02
 4.5156E-02 -3.1488E-02 -3.7071E-03 -3.9984E-03 -4.8984E-03 -3.9510E-03
ISIN =     0
THE NEXT TRY FOR PVRDOT =   8.2120E-01
```

```
   10   THE VARIABLE IS NO.    5
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -2.6030E-03  1.4415E-04 -3.7263E-04 -5.1384E-03  5.0135E-04 -1.5291E-02  9.9429E-06 -3.0326E-03
-1.2695E-02 -5.2099E-03  1.1571E-05  4.6941E-04 -1.1102E-15  1.8515E-04
IGIN =      1
THE NEXT TRY FOR PVRDOT =    3.0010E-02
   10   THE VARIABLE IS NO.    5
 THIS COLUMN OF X =
-2.7209E-01  2.8711E-01 -2.6338E-03  1.1297E-04 -6.4521E-03 -5.6752E-03  2.4645E-04 -1.0045E-02  1.0105E-04 -2.0190E-03
-5.6660E-03 -3.8532E-03  2.1621E-04  2.3069E-04  1.1417E-04  1.2241E-04
IGIN =      2
THE NEXT TRY FOR PVRDOT =    2.8510E-02
   10   THE VARIABLE IS NO.    5
 THIS COLUMN OF X =
-2.5536E-01  2.7342E-01 -2.4997E-03  1.0823E-04 -6.1344E-03 -5.3770E-03  2.4199E-04 -9.5157E-03  9.5847E-05 -1.9207E-03
-5.4644E-03 -3.7201E-03  2.0747E-04  2.2638E-04  1.0827E-04  1.1754E-04
IGIN =      3
THE NEXT TRY FOR PVRDOT =    3.8728E 00
   10   THE VARIABLE IS NO.    5
 THIS COLUMN OF X =
-4.3430E 01  3.5910E 01 -3.3321E-01  9.7761E-03 -9.3519E-01 -7.4909E-01  1.4317E-02 -1.3103E 00  1.4035E-02 -2.4153E-01
-7.0357E-01 -5.4429E-01  1.8621E-02  1.3824E-02  1.5841E-02  9.1294E-03
IGIN =      4
THE NEXT TRY FOR PVRDOT =    4.0697E 00
   10   THE VARIABLE IS NO.    5
 THIS COLUMN OF X =
-4.5638E 01  3.7737E 01 -3.5009E-01  1.0274E-02 -9.8253E-01 -7.8717E-01  1.5019E-02 -1.3771E 00  1.4751E-02 -2.5399E-01
-7.3901E-01 -5.7167E-01  1.9568E-02  1.4503E-02  1.6648E-02  9.5942E-03
IGIN =      0
THE NEXT TRY FOR PVRDOT =    4.0597E 00
   11   THE VARIABLE IS NO.   13
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -2.2204E-16  8.1712E-14 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =      1
THE NEXT TRY FOR PVRDOT =    3.0010E-02
   11   THE VARIABLE IS NO.   13
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -2.2204E-16  8.1712E-14 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =      2
THE NEXT TRY FOR PVRDOT =    6.0020E-02
   11   THE VARIABLE IS NO.   13
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -2.7330E-05 -4.5072E-03 -5.6843E-14 -1.0659E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
-7.7347E-03 -7.5829E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15 -3.7880E-05
IGIN =      3
THE NEXT TRY FOR PVRDOT =    1.2004E-01
```

```
   11   THE VARIABLE IS NO.  13
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -5.3552E-05 -8.8316E-03 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
-1.5156E-02 -1.4858E-02 -2.6645E-15 -1.0658E-14 -1.1102E-15 -7.4225E-05
IGIN =     4
THE NEXT TRY FOR PVRDOT =   2.4008E-01
   11   THE VARIABLE IS NO.  13
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -1.0601E-04 -1.7481E-02 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
-2.9999E-02 -2.9411E-02 -2.6645E-15 -1.0658E-14 -1.1102E-15 -1.4693E-04
IGIN =     5
THE NEXT TRY FOR PVRDOT =   4.8016E-01
   11   THE VARIABLE IS NO.  13
 THIS COLUMN OF X =
     0.0000 -3.6380E-12 -4.1595E-03 -1.8593E-04 -3.7174E-02 -5.2849E-03 -1.1735E-03  7.9125E-03 -2.1439E-04 -3.1190E-03
-5.1237E-02 -4.8551E-02  3.5079E-07 -1.1116E-03 -1.9962E-04 -1.9500E-04
IGIN =     6
THE NEXT TRY FOR PVRDOT =   9.6032E-01
   11   THE VARIABLE IS NO.  13
 THIS COLUMN OF X =
 1.0859E 00 -1.8118E-01 -4.6142E-03 -3.3352E-04 -6.9070E-02 -1.9642E-03  1.6783E-03 -2.8232E-02 -3.7431E-04 -5.2295E-03
-1.4873E-01 -1.2969E-01 -3.2917E-04  1.5640E-03 -3.9927E-04 -3.5480E-04
IGIN =     7
THE NEXT TRY FOR PVRDOT =   4.1972E 00
   11   THE VARIABLE IS NO.  13
 THIS COLUMN OF X =
 4.8427E 00 -1.0551E 00 -1.9953E-02 -1.5346E-03 -2.9521E-01 -7.7446E-03  6.5998E-03 -1.1578E-01 -1.7758E-03 -2.7393E-02
-5.4421E-01 -5.5954E-01 -1.7660E-03  6.1499E-03 -1.8968E-03 -1.6174E-03
IGIN =     8
THE NEXT TRY FOR PVRDOT =   5.2535E 00
   11   THE VARIABLE IS NO.  13
 THIS COLUMN OF X =
 5.0222E 00 -1.3477E 00 -2.5029E-02 -1.9348E-03 -3.6885E-01 -9.3468E-03  8.0852E-03 -1.4199E-01 -2.2320E-03 -3.4312E-02
-8.0437E-01 -6.9857E-01 -2.2544E-03  7.5344E-03 -2.3838E-03 -2.0408E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   5.2535E 00
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -2.2204E-16  8.1712E-14 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -2.2204E-16  8.1712E-14 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
```

```
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  1.0397E-06 -2.2858E-03 -5.5843E-14 -1.0658E-14  5.5843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13 -3.8457E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     3
THE NEXT TRY FOR PVRDOT =   1.2004E-01
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  2.1541E-06 -4.7358E-03 -5.5843E-14 -1.0658E-14  5.5843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13 -7.9674E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     4
THE NEXT TRY FOR PVRDOT =   2.4008E-01
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  4.3830E-06 -9.6360E-03 -5.5843E-14 -1.0658E-14  5.5843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13 -1.6212E-02 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     5
THE NEXT TRY FOR PVRDOT =   4.8016E-01
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  8.8417E-06 -1.9438E-02 -5.5843E-14 -1.0658E-14  5.5843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13 -3.2703E-02 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     6
THE NEXT TRY FOR PVRDOT =   9.6032E-01
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  1.7762E-05 -3.9049E-02 -5.5843E-14 -1.0658E-14  5.5843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13 -6.5695E-02 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     7
THE NEXT TRY FOR PVRDOT =   1.9206E 00
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
 3.9552E-01  4.7398E-01 -3.8744E-04 -2.0027E-04 -9.4820E-02  3.7991E-03  6.9672E-04 -5.5206E-03 -9.2349E-05  5.3702E-04
-7.7955E-03 -1.4507E-01 -1.1108E-04  5.4386E-04 -5.7360E-05 -2.6730E-04
IGIN =     8
THE NEXT TRY FOR PVRDOT =   8.2593E 00
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
 1.5193E 00  2.0168E 00  7.6251E-04 -1.0128E-03 -4.0596E-01  2.0086E-02  2.1978E-03 -5.7387E-03 -3.9423E-04  5.5663E-03
-1.7851E-02 -6.1737E-01 -6.1637E-04  2.0306E-03 -2.2664E-04 -1.3558E-03
IGIN =     9
THE NEXT TRY FOR PVRDOT =   8.9522E 00
   12   THE VARIABLE IS NO.  17
 THIS COLUMN OF X =
 1.8955E 00  2.0020E 00 -9.1089E-04 -1.0094E-03 -4.3617E-01  1.9679E-02  2.4932E-03 -1.2742E-02 -5.1259E-04  3.6500E-03
-2.5673E-02 -6.6903E-01 -7.5975E-04  2.2994E-03 -3.8696E-04 -1.3196E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   8.9522E 00
```

```
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -8.8678E-06 -5.0423E-04 -5.5843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13 -3.8686E-04 -2.6645E-15 -1.0658E-14 -1.1102E-15 -1.2609E-05
IGIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -1.3306E-05 -7.5660E-04 -5.5843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13 -5.8049E-04 -2.6645E-15 -1.0658E-14 -1.1102E-15 -1.8919E-05
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -2.6612E-05 -1.5132E-03 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13 -1.1610E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15 -3.7839E-05
IGIN =     3
THE NEXT TRY FOR PVRDOT =   1.2004E-01
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -5.3226E-05 -3.0265E-03 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 1.1369E-13 -2.3220E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15 -7.5678E-05
IGIN =     4
THE NEXT TRY FOR PVRDOT =   2.4008E-01
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14 -8.0770E-05 -1.8176E-03 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.8818E-16 -3.1974E-14
 7.2683E-03  2.4816E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15 -1.1576E-04
IGIN =     5
THE NEXT TRY FOR PVRDOT =   4.8016E-01
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
 9.9247E-01  1.7515E-02 -1.8998E-03 -1.7536E-04 -7.2601E-03  1.4806E-03  2.0194E-03 -2.9592E-02 -2.1343E-04 -3.3925E-03
-1.6184E-02 -1.4545E-02 -3.1541E-04  1.8749E-03 -2.2926E-04 -1.7893E-04
IGIN =     6
THE NEXT TRY FOR PVRDOT =   8.8666E 00
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
 1.8107E 01 -6.4604E-01 -3.0056E-02 -3.7380E-03 -1.0301E-01  3.5734E-02  3.2839E-02 -4.8557E-01 -4.3510E-03 -5.7700E-02
-2.4168E-01 -2.2034E-01 -7.2746E-03  3.0196E-02 -4.6813E-03 -3.9431E-03
IGIN =     7
THE NEXT TRY FOR PVRDOT =   1.3759E 01
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
 2.8548E 01 -8.2890E-01 -4.4691E-02 -5.8131E-03 -1.6268E-01  5.2591E-02  5.2273E-02 -7.7595E-01 -6.7134E-03 -8.8325E-02
-3.8983E-01 -3.5637E-01 -1.0936E-02  4.8339E-02 -7.2368E-03 -6.0187E-03
IGIN =     8
THE NEXT TRY FOR PVRDOT =   1.4288E 01
```

```
   13   THE VARIABLE IS NO.  12
 THIS COLUMN OF X =
 2.9730E 01 -8.7094E-01 -4.6476E-02 -6.0407E-03 -1.6865E-01  5.2700E-02  5.4215E-02 -8.0540E-01 -5.3810E-03 -9.1454E-02
-4.0382E-01 -3.6915E-01 -1.1374E-02  5.0338E-02 -7.5181E-03 -6.2548E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   1.4288E 01
   14   THE VARIABLE IS NO.  10
 THIS COLUMN OF X =
     0.0000 -3.6380E-12 -4.1595E-03 -2.5862E-04 -9.1864E-03 -5.2849E-03 -1.1735E-03  1.5527E-02 -2.1499E-04 -3.1190E-03
 5.4629E-03  3.1605E-03 -5.0135E-04 -1.1116E-03 -1.9962E-04 -2.8112E-04
IGIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
   14   THE VARIABLE IS NO.  10
 THIS COLUMN OF X =
     0.0000 -3.6380E-12 -4.1595E-03 -4.0059E-04 -1.2589E-02 -5.2849E-03 -1.1735E-03  2.0333E-02 -2.1499E-04 -3.1190E-03
 3.9630E-03 -2.4956E-04 -7.5246E-04 -1.1116E-03 -1.9962E-04 -4.7056E-04
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
   14   THE VARIABLE IS NO.  10
 THIS COLUMN OF X =
-6.2617E-01 -6.8938E-01 -1.1250E-02 -5.6169E-04 -3.8006E-03 -2.7031E-02 -7.4835E-03  4.7507E-02 -4.7522E-04 -9.4907E-03
 2.5057E-02  2.5627E-02 -1.6906E-03 -7.2159E-03 -4.8537E-04 -6.3162E-04
IGIN =     3
THE NEXT TRY FOR PVRDOT =   2.5994E-01
   14   THE VARIABLE IS NO.  10
 THIS COLUMN OF X =
-3.4406E 00 -3.4482E 00 -5.1329E-02 -2.3626E-03  1.5491E-03 -1.2425E-01 -3.5967E-02  2.5293E-01 -2.2350E-03 -4.2451E-02
 1.5453E-01  1.5466E-01 -7.4286E-03 -3.4580E-02 -2.2773E-03 -2.6063E-03
IGIN =     4
THE NEXT TRY FOR PVRDOT =   3.5684E-01
   14   THE VARIABLE IS NO.  10
 THIS COLUMN OF X =
-4.7544E 00 -4.7632E 00 -7.0456E-02 -3.2546E-03  2.9223E-03 -1.7068E-01 -4.9563E-02  3.4969E-01 -3.0787E-03 -5.8224E-02
 2.1434E-01  2.1422E-01 -1.0229E-02 -4.7645E-02 -3.1374E-03 -3.5904E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   3.5684E-01
   15   THE VARIABLE IS NO.   4
 THIS COLUMN OF X =
     0.0000 -1.8190E-12  4.0510E-04  4.3128E-06  3.0482E-03 -5.6843E-14 -1.0658E-14  5.6843E-14  1.0130E-05 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
   15   THE VARIABLE IS NO.   4
 THIS COLUMN OF X =
     0.0000 -1.8190E-12  1.7331E-03 -3.9492E-05  5.7380E-03  2.2212E-03 -2.1672E-04  9.0485E-03  1.0977E-05  1.3109E-03
 7.2173E-03  3.9439E-03  2.3541E-05 -2.0292E-04 -1.1102E-15 -5.8379E-05
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
```

```
   15   THE VARIABLE IS NO.   4
 THIS COLUMN OF X =
     0.0000 -1.8190E-12  3.8516E-03 -3.0159E-05  1.0774E-02  4.4333E-03 -4.3254E-04  1.5665E-02  2.1972E-05  2.6165E-03
 1.3202E-02  6.6952E-03  2.7140E-05 -4.0499E-04 -1.1102E-15 -1.3157E-04
IGIN =     3
THE NEXT TRY FOR PVRDOT =   1.2004E-01
   15   THE VARIABLE IS NO.   4
 THIS COLUMN OF X =
 7.1928E-01  2.2771E-02  2.3292E-03 -2.4320E-04  1.1986E-02  6.9162E-03 -2.0021E-04  1.3209E-02 -3.3479E-04  5.3631E-04
 5.6192E-03  9.1895E-04 -2.5950E-04 -1.8868E-04 -3.4477E-04 -2.3794E-04
IGIN =     4
THE NEXT TRY FOR PVRDOT =   1.1032E 00
   15   THE VARIABLE IS NO.   4
 THIS COLUMN OF X =
 6.4595E 00 -5.8194E-01  2.0431E-02 -2.3006E-03  1.3963E-01  6.7292E-02 -4.7452E-03  1.7544E-01 -3.1730E-03  3.3237E-03
 9.4424E-02  5.4532E-02 -3.1400E-03 -4.4463E-03 -3.5743E-03 -2.1800E-03
IGIN =     5
THE NEXT TRY FOR PVRDOT =   1.8393E 00
   15   THE VARIABLE IS NO.   4
 THIS COLUMN OF X =
 1.1749E 01 -8.0856E-01  3.5883E-02 -3.7814E-03  2.2837E-01  1.0619E-01 -6.1253E-03  2.5089E-01 -5.3231E-03  6.1838E-03
 1.2595E-01  6.4543E-02 -5.0577E-03 -5.7793E-03 -6.0082E-03 -3.5475E-03
IGIN =     0
THE NEXT TRY FOR PVRDOT =   1.8393E 00
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  5.5080E-07 -1.8722E-04 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-15 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     1
THE NEXT TRY FOR PVRDOT =   3.0010E-02
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  9.2647E-07 -2.8093E-04 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     2
THE NEXT TRY FOR PVRDOT =   6.0020E-02
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  1.6529E-06 -5.6185E-04 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-15 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     3
THE NEXT TRY FOR PVRDOT =   1.2004E-01
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  3.3059E-06 -1.1237E-03 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13  1.7053E-13 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     4
THE NEXT TRY FOR PVRDOT =   2.4008E-01
```

```
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  5.3704E-06  4.8161E-04 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-15 -3.1974E-14
 1.1369E-13  4.5914E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     5
THE NEXT TRY FOR PVRDOT =   4.8016E-01
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
     0.0000 -1.8190E-12 -3.9080E-14  1.0916E-05  7.9900E-04 -5.6843E-14 -1.0658E-14  5.6843E-14 -8.3818E-16 -3.1974E-14
 1.1369E-13  8.9068E-03 -2.6645E-15 -1.0658E-14 -1.1102E-15  7.5495E-15
IGIN =     6
THE NEXT TRY FOR PVRDOT =   9.6032E-01
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
 3.0830E-01  3.5610E-01 -3.9782E-04 -1.4345E-04 -1.1121E-02  2.6153E-03  5.9942E-04 -5.8535E-03 -5.5501E-05  3.1574E-04
-7.2489E-03  6.6445E-03 -7.4446E-05  5.5387E-04 -3.9586E-05 -1.8619E-04
IGIN =     7
THE NEXT TRY FOR PVRDOT =   1.4345E 01
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
 4.3807E 00  4.7079E 00 -6.1342E-04 -2.5272E-03 -1.4436E-01  4.8174E-02  5.6199E-03 -2.5284E-02 -1.2350E-03  1.0124E-02
-5.6371E-02  1.3582E-01 -1.9184E-03  5.1781E-03 -8.6925E-04 -3.2296E-03
ISIN =     8
THE NEXT TRY FOR PVRDOT =   2.3785E 01
   16   THE VARIABLE IS NO.  16
 THIS COLUMN OF X =
 7.2121E 00  7.7767E 00 -8.6033E-04 -4.2119E-03 -2.3882E-01  7.9947E-02  9.0461E-03 -3.8192E-02 -2.0534E-03  1.7201E-02
-9.0033E-02  2.2739E-01 -3.2342E-03  8.3327E-03 -1.4523E-03 -5.3824E-03
ISIN =     0
THE NEXT TRY FOR PVRDOT =   2.3785E 01
```

```
 I INVERSE =
 1.2744E-02  5.2152E-03  4.4875E-02  1.2138E 00  2.7403E-02  5.4851E-03  2.0902E-01  1.7503E-03  4.3163E-01  1.9355E-03
 4.7238E-04  3.3395E-04  1.2351E-02  1.1710E-01  1.8145E-01  1.5640E-02
AINV =
-3.0361E-01 -1.2767E-01 -1.3682E-03 -4.9360E-05  1.6744E-03 -2.6140E-03 -1.0188E-03  1.2731E-02 -3.3805E-05 -3.5354E-04
 9.0641E-03  8.7402E-03 -2.0272E-04 -9.4552E-04 -3.7131E-05 -5.6965E-05
 9.5122E-02 -1.0299E-01 -8.5334E-04 -3.5200E-05  3.9588E-04 -9.5795E-04  7.0880E-05 -2.3999E-03 -5.5239E-05 -1.1921E-03
-2.4953E-03 -2.6258E-03 -4.0857E-05  6.7086E-05 -6.3368E-05 -3.2366E-05
 2.4067E-02  2.4475E-02 -1.2003E-02 -1.3310E-05 -5.7922E-03  1.9311E-05 -1.7063E-05  5.0050E-04 -2.0105E-05 -1.2608E-04
 2.8034E-04  1.0473E-06 -4.4543E-06 -1.5491E-05 -1.7226E-05 -1.0021E-05
 9.8543E 00  1.0060E 01 -1.4693E-02 -5.1949E-03 -3.8106E-02  9.3286E-02  9.1358E-03 -1.5155E-02 -3.5138E-03  1.1657E-02
-8.8330E-02 -2.2562E-01 -4.2708E-03  8.4653E-03 -2.7753E-03 -6.7390E-03
 5.7015E-02  5.7822E-02 -9.1418E-05 -3.8042E-05 -1.3910E-02  5.1855E-04  5.8330E-05 -2.0911E-04 -2.0255E-05  5.6828E-05
-6.0880E-04 -1.3630E-03 -2.2730E-05  5.3773E-05 -1.5490E-05 -3.7353E-05
 3.5922E-01  1.2309E-01  1.1015E-03  3.7944E-05 -3.6619E-03 -2.8508E-03  9.7730E-04 -1.9479E-02  2.1490E-05  7.8958E-04
-1.3470E-02 -1.2703E-02  1.9601E-04  9.1192E-04  2.1345E-05  4.5445E-05
-1.7685E 00 -1.3579E 00 -1.3803E-02 -5.1475E-04  2.1270E-02 -1.7023E-02 -1.0654E-02  1.4991E-01 -3.9840E-04 -3.8943E-03
 1.0314E-01  9.8943E-02 -2.1542E-03 -9.7930E-03 -3.7944E-04 -5.9570E-04
-3.6525E-02 -2.7554E-02 -3.7105E-04 -1.5604E-05 -1.1031E-03 -8.0066E-04 -2.1677E-04 -2.0539E-03 -1.4746E-05 -2.9617E-04
-1.3507E-03 -1.0125E-03 -4.4391E-05 -2.0480E-04 -1.4931E-05 -1.7366E-05
 3.3066E 00  2.7855E 00  1.3323E-01 -1.7706E-03  1.4572E-01  1.3907E-02 -1.8593E-03  5.2804E-02 -2.4350E-03 -1.2420E-02
 1.9491E-02 -1.3591E-02 -1.5001E-03 -1.7258E-03 -2.1143E-03 -1.7054E-03
-8.8333E-02  7.3039E-02 -6.7760E-04  1.9886E-05 -1.9017E-03 -1.5236E-03  2.9059E-05 -2.6553E-03  2.8551E-05 -4.9141E-04
-1.4304E-03 -1.1065E-03  3.7873E-05  2.8070E-05  3.2223E-05  1.8570E-05
 2.8448E-03 -6.3663E-04 -1.1823E-05 -9.1399E-07 -1.7424E-04 -4.4153E-06  3.8193E-05 -5.7023E-05 -1.0544E-06 -1.5209E-05
-3.7997E-04 -3.2999E-04 -1.0649E-06  3.5591E-06 -1.1261E-06 -9.6402E-07
 5.3301E-04  6.6855E-04 -3.0419E-07 -3.3709E-07 -1.4566E-04  6.5718E-06  8.3259E-07 -4.2551E-06 -1.3119E-07  1.2139E-06
-8.5734E-06 -2.2342E-04 -2.5372E-07  7.6788E-07 -1.2923E-07 -4.4068E-07
 3.6719E-01 -1.0757E-02 -5.7402E-04 -7.4508E-05 -2.0830E-03  6.5090E-04  6.6961E-04 -3.9474E-03 -8.5222E-05 -1.1297E-03
-4.9875E-03 -4.5593E-03 -1.4048E-04  6.2172E-04 -9.2856E-05 -7.7253E-05
-5.5675E-01 -5.5779E-01 -8.2506E-03 -3.8112E-04  3.4221E-04 -1.9987E-02 -5.8039E-03  4.0950E-02 -3.6053E-04 -6.8182E-03
 2.5099E-02  2.5085E-02 -1.1978E-03 -5.5794E-03 -3.6740E-04 -4.2045E-04
 2.1319E 00 -1.4672E-01  6.5111E-03 -5.8616E-04  4.1439E-02  1.9268E-02 -1.1115E-03  4.5523E-02 -9.6591E-04  1.1221E-03
 2.2853E-02  1.1712E-02 -9.1774E-04 -1.0487E-03 -1.0902E-03 -6.4372E-04
 1.1280E-01  1.2163E-01 -1.3456E-05 -6.5875E-05 -3.7351E-03  1.2504E-03  1.4143E-04 -5.9733E-04 -3.2193E-05  2.6539E-04
-1.4081E-03  3.5564E-03 -5.0583E-05  1.3032E-04 -2.2714E-05 -8.4181E-05
```

A =

```
-5.0565E 00 -3.0710E-02 -2.9468E-01  3.9335E 01 -5.3391E-01 -2.6047E 02  2.0381E 03  2.7527E 01 -6.5545E 01  5.9342E 02
 8.3973E-02 -5.7299E-02 -3.4526E 03  4.5147E 02 -1.4176E 04  4.3767E 00
 2.3346E-02 -4.8023E 00  3.0603E 00 -2.2896E 02 -3.1802E-02 -7.0417E-01 -5.2383E 02  6.5972E-01  3.1051E 03 -5.0291E 02
 3.5671E 01 -4.3288E-02  1.1222E 04  5.3972E 02 -1.3135E 03 -1.6550E 04
 1.0294E-02  1.6695E-02 -7.5582E 01  1.0153E 04  2.2164E-01  2.3439E 00  2.4440E 02  1.9153E-01 -2.4997E 04  7.0109E 01
 2.1929E-02  8.1418E-01  3.5630E 01 -2.3794E 02  1.6168E 04 -2.7766E 03
-2.9561E-04 -8.3654E-04 -4.0383E-02 -1.2232E 03  4.1008E-01  8.5963E-03  3.2784E 00 -5.3816E-03  3.5745E 02  3.1630E-02
-5.4918E-04 -2.7182E-01 -7.5532E-01 -3.4222E 00 -9.0175E 00  7.8874E 02
 5.0849E-03  7.9130E-03  4.9072E 01 -4.6135E 04 -7.3045E 01  3.5386E-01 -5.9434E 01  5.0753E-02  3.3201E 03 -2.7018E 00
 7.3962E-02  3.6321E 01  4.3765E 01  5.1096E 01  3.3498E 02  3.9300E 04
 1.1725E 00 -2.0090E-01  6.2355E-01  6.1426E 02 -4.8913E-02 -1.6955E 02 -8.0099E 03  3.2409E 00 -1.9497E 01  2.6140E 02
 7.1633E-01  4.1844E-02  1.2136E 02  8.5777E 03  1.3736E 03 -8.0441E 02
 6.6468E-01 -3.6682E-03  3.8230E-03  5.9720E 00 -2.2886E-03  7.3205E 00 -3.0473E 03  2.5755E-02 -3.2135E 01 -4.6793E 01
-1.0932E-03  4.6000E-04  9.8476E-02  3.0318E 03  1.6012E 03 -6.1656E 00
 2.2990E-01 -7.2786E-02 -2.3702E 00 -9.7965E 02  4.5040E-02  3.7147E 02 -3.4792E 04 -2.5199E 02 -1.2709E 03  8.5757E 00
 2.3863E-01 -3.5829E-01  2.4379E 01  3.2969E 04  1.4294E 03  9.3226E 02
 7.7884E-04  1.1386E-03  6.2887E-01  7.5500E 02  1.6265E-02 -7.4471E-02 -1.2388E 01  1.4297E-02 -2.0390E 03 -9.2605E-01
 9.0396E-04  6.0542E-02  2.5044E 00  1.2848E 01  1.4551E 03 -2.0622E 02
 1.2849E-01  9.5445E 00  1.1839E 00 -1.4713E 02  6.3233E-01  1.9940E 01 -1.5712E 04  1.8281E 00  7.2239E 04 -7.8616E 02
 1.0101E-01  1.1290E-01 -2.9718E 02  1.5921E 04 -6.4339E 04  6.9841E 00
-3.9575E 00  1.9362E 01 -3.8123E 01  1.5028E 03  8.1694E 00 -7.7309E 01 -4.3532E 04  1.9189E 03 -2.0335E 04  7.5304E 02
-3.0769E 03  2.1250E 00 -8.5180E 04  5.2646E 04  1.4233E 04  1.2376E 05
-2.2171E-01 -1.6654E 01 -5.2295E 02  2.3339E 06 -1.1979E 01  1.0143E 02 -5.8164E 03  1.9571E 00 -4.0040E 05 -9.2325E 01
 3.7857E 03 -4.0282E 03  7.1729E 04  5.3449E 03 -6.6422E 04 -2.3473E 06
-8.0010E-02  9.6115E-01 -2.4339E 00  1.4322E 02  1.5497E-01 -7.9874E 00  1.6109E 03 -1.2131E 01 -2.2242E 03  1.1273E 01
 3.4646E 00  7.1001E-02 -6.8456E 03 -8.3737E 02  2.0386E 03  6.8357E 03
 3.2134E-02 -1.8142E-02 -3.5590E-02  5.1426E-01  1.6311E-02 -1.1417E 01  5.2277E 03  4.9554E 00 -2.1763E 02 -1.8499E 00
-1.7387E-02 -2.7979E-03  3.5362E 01 -5.5485E 03  2.3932E 02 -9.6110E-01
-3.5033E-03  4.7999E-01 -3.0807E-02  5.1570E 00 -2.1147E-02  3.1878E 00 -9.6545E 02 -5.1237E-02  5.8749E 03  3.2905E 00
 3.2289E-03 -2.6589E-03  8.2812E 00  9.6004E 02 -6.0850E 03 -2.4889E 00
-9.6409E-03 -6.5289E-01 -1.5944E 01  8.7379E 04 -3.7206E-01  1.6765E 00  8.2207E 01  3.3614E-02 -1.2211E 04 -4.1532E 00
-6.6185E 00  2.5525E 01  4.6443E 03 -1.0187E 02 -2.1588E 03 -7.7551E 04
```

C =

```
-6.4699E-07 -2.0282E-06  6.8625E-05 -4.4260E-01  6.5684E-06  2.3009E-05  6.6218E-03 -2.1431E-05  2.2785E-02  3.1351E-05
-9.7537E-07  9.5610E-08 -1.9250E-03 -6.8742E-03 -7.9298E-03  1.1598E-02
 1.1987E-02  4.6434E-02 -1.2509E 00  8.0195E 03 -7.7002E-02 -5.9215E-01 -1.6055E 02  2.8252E-01 -3.1748E 02 -1.1673E 00
 4.8001E-02 -5.6520E-03  3.6014E 01  1.6665E 02  6.3557E 01 -2.0253E 02
 1.1987E-02  4.6434E-02 -1.2509E 00  8.0195E 03 -7.7002E-02 -5.9215E-01 -1.6055E 02  2.3252E-01 -3.1748E 02 -1.1573E 00
 4.8001E-02 -5.6520E-03  3.6014E 01  1.6665E 02  6.3557E 01 -2.0253E 02
```

OUTPUT AM= 0.000 ALTP= 0. T4= 2900.59 ETAR= 1.0000

MIDDLE SPOOL IS DUMMY

NO AIRFLOW INTO WING
THE OUTPUT IS IN ENGLISH UNITS

PCNF	CNF	ZF	PRF	WAFC	WAF
0.102552E 03	0.102551E 01	0.834671E 00	0.300924E 01	0.222300E 03	0.222300E 03
PCNC	CNC	ZC	PRC	WACC	WAC
0.118482E 03	0.988871E 00	0.814720E 00	0.848852E 01	0.551328E 02	0.138470E 03
T2	P2	T21	P21	T3	P3
0.518670E 03	0.100000E 01	0.744583E 03	0.300924E 01	0.1471146E 04	0.255440E 02
PCBLF	BLF	PCBLC	BLC	PCBLOB	BLOB
0.000000	0.000000	0.160000E 00	0.221552E 02	0.000000	0.000000
PCBLHP	BLHP	PCBLLP	BLLP	T4	P4
0.726000E 00	0.160847E 02	0.660000E-01	0.146224E 01	0.290059E 04	0.241114E 02
WA3	WFB	WG4	FAR4	ETAB	DPCOM
0.116315E 03	0.277750E 01	0.119092E 03	0.238793E-01	0.100000E 01	0.560859E-01
TFFHP	CNHP	DHTCHP	DHTC	T5	P5
0.499981E 02	0.200241E 01	0.734620E-01	0.213088E 03	0.210970E 04	0.570544E 01
TFFLP	CNLP	DHTCLP	DHTF	T55	P55
0.130079E 03	0.230199E 01	0.423525E-01	0.893510E 02	0.1798115E 04	0.270347E 01
PCBLDU	BLDU	T24	P24	T25	P25
0.208000E 00	0.460828E 01	0.793940E 03	0.283418E 01	0.793940E 03	0.283418E 01
WAD	WFD	WG24	FAR24	ETAD	DPDUC
0.884383E 02	0.000000	0.884383E 02	0.000000	0.000000	0.581737E-01
ETAF	ETAC	ETATHP	ETATLP	AM55	AM25
0.847718E 00	0.813972E 00	0.871273E 00	0.902172E 00	0.2833515 00	0.381747E 00
T6	P6	PS6	AM6	V6	WG6
0.141910E 04	0.272825E 01	0.262221E 01	0.243037E 00	0.438078E 03	0.225077E 03
T7	WFA	WG7	FAR7	ETAA	DPAFT
0.141910E 04	0.000000	0.225077E 03	0.1249448-01	0.000000	0.598972E-01
PS8	AM8	V8	PS9	AM9	V9
0.138387E 01	0.100000E 01	0.167738E 04	0.138387E 01	0.100000E 01	0.167738E 04
PS28	AM28	V28	PS29	AM29	V29
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
BYPASS	HPEXT	WFT	WGT	VA	FRD
0.605403E 00	0.000000	0.277750E 01	0.225077E 03	0.000000	0.000000
CVMNOZ	VJM	CVDNOZ	VJD	FGM	FGP
0.949400E 00	0.1592595 04	0.000000	0.000000	0.111405E 05	0.239540E 04

MAIN SONIC CONVERGENT NOZZLE FG= 13536.00 FN= 13536.00 SFC= 0.73870

CONVERGED AFTER 9 LOOPS

OUTPUT AM= 0.000 ALTP= 0. T4= 2891.96 ETAP= 1.0000

MIDDLE SPOOL IS DUMMY

NO AIRFLOW INTO WING
THE OUTPUT IS IN ENGLISH UNITS

PCNF	CNF	ZF	PRF	WAFC	WAF
0.102579E 03	0.102579E 01	0.825774E 00	0.298893E 01	0.222785E 03	0.222784E 03
PCNC	CNC	ZC	PRC	WACC	WAC
0.118421E 03	0.989320E 00	0.814219E 00	0.849040E 01	0.551772E 02	0.137779E 03
T2	P2	T21	P21	T3	P3
0.518670E 03	0.100000E 01	0.743148E 03	0.298893E 01	0.146874E 04	0.253772E 02
PCBLF	BLF	PCBLC	BLC	PCBLOB	BLOB
0.000000	0.000000	0.160000E 00	0.220446E 02	0.000000	0.000000
PCBLHP	BLHP	PCBLLP	BLLP	T4	P4
0.726000E 00	0.160044E 02	0.660000E-01	0.145494E 01	0.289196E 04	0.239530E 02
WA3	WFB	WG4	FAR4	ETAB	DPCOM
0.115734E 03	0.275000E 01	0.118484E 03	0.237613E-01	0.100000E 01	0.561209E-01
TFFHP	CNHP	DHTCHP	DHTC	T5	P5
0.499974E 02	0.200437E 01	0.735434E-01	0.212685E 03	0.210247E 04	0.555582E 01
TFFLP	CNLP	DHTCLP	DHTF	T55	P55
0.130333E 03	0.230656E 01	0.425347E-01	0.894282E 02	0.178644E 04	0.267101E 01
PCBLDU	BLDU	T24	P24	T25	P25
0.208000E 00	0.458528E 01	0.781733E 03	0.281184E 01	0.781733E 03	0.281184E 01
WAD	WFD	WG24	FAR24	ETAD	DPDUC
0.895909E 02	0.000000	0.895909E 02	0.000000	0.000000	0.592483E-01
ETAF	ETAC	ETATHP	ETATLP	AM55	AM25
0.847070E 00	0.914197E 00	0.871218E 00	0.902435E 00	0.285327E 00	0.390315E 00
T6	P6	PS6	AM6	V6	WG6
0.140908E 04	0.269846E 01	0.259177E 01	0.245431E 00	0.440899E 03	0.225534E 03
T7	WFA	WG7	FAR7	ETAA	DPAFT
0.140908E 04	0.000000	0.225534E 03	0.123437E-01	0.000000	0.604555E-01
PS8	AM8	V8	PS9	AM9	V9
0.136756E 01	0.100000E 01	0.167161E 04	0.136756E 01	0.100000E 01	0.167161E 04
PS28	AM28	V28	PS29	AM29	V29
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
BYPASS	HPEXT	WFT	WGT	VA	FRD
0.616972E 00	0.000000	0.275000E 01	0.225534E 03	0.000000	0.000000
CVMNOZ	VJM	CVDNOZ	VJD	FGM	FGP
0.949400E 00	0.158703E 04	0.000000	0.000000	0.111248E 05	0.231661E 04

MAIN SONIC CONVERGENT NOZZLE FG= 13441.39 FN= 13441.39 SFC= 0.73653

CONVERGED AFTER 10 LOOPS

```
B =
 -3.3971E 02 -1.8844E 01
 -7.6357E 00  8.1031E 01
  2.7458E 01 -3.6972E 01
 -7.3679E-01 -7.4372E 01
  2.5007E 01 -2.0494E 03
  8.0562E 01 -5.9852E 01
  4.4730E 00 -1.0116E 00
  3.3675E 04 -4.6442E 01
  1.9901E 00 -2.8772E 00
  5.3471E 02 -1.2480E 01
  5.3638E 03 -1.0988E 02
  2.0804E 03  1.1746E 03
  9.2601E 00 -3.0955E 01
  9.7612E 00 -1.6070E 00
 -1.0343E 01 -3.4356E 00
 -5.7335E 00  4.6618E 01
D =
  2.6713E-01 -2.4535E-01
  2.9886E 01  4.4594E 03
  2.9886E 01  4.4594E 03
AR =
-5.8400E 00  1.9103E 00  3.2736E 00 -4.8070E 03  1.5055E-01 -2.8179E 02  2.0889E 03  3.5559E 01 -4.1954E 03
-2.6600E-01 -5.4974E 00 -8.8566E-01 -5.3708E 03 -1.6537E-01  1.2807E 00  1.2805E 03  9.4515E 00 -3.5689E 03
 3.5361E-02  1.7256E 00 -7.5892E 01  5.8571E 03  2.6559E-01  3.7053E 00 -1.6503E 02  3.0925E 00 -6.8853E 03
-1.0269E-03 -2.0348E-03 -2.3390E-01 -2.7788E 02  4.0726E-01 -1.4569E-02  5.9931E 00 -1.0465E-01  2.0315E 02
-5.3101E-02  3.6424E-02  3.9359E 01  2.7522E 03 -7.3122E 01 -1.7047E 00  1.5611E 02  2.3756E 01 -3.8977E 03
 1.2848E 00  1.3330E 00  1.1671E 00 -3.1032E 02  2.5177E-01 -1.9499E 02  3.4758E 01  1.3915E 01  2.7358E 03
 6.7089E-01 -1.2437E-01 -1.3357E-01  5.1651E 01 -4.7405E-02  2.8472E 00 -1.9902E 02  2.4825E 00  9.3588E 02
 4.1724E-01  8.1581E-02 -2.9698E 00  4.3834E 02  1.3473E-01  3.0396E 02 -3.7220E 03 -2.2243E 02 -5.7708E 01
 1.4055E-03  1.1600E-01  6.6580E-01  5.1543E 02  1.2537E-02  1.9911E-01 -1.5713E 01  2.3179E-01 -5.9552E 02
BR =
  2.0662E 02  2.3575E 01
 -2.8947E 02  1.8122E 01
  6.5854E 01 -4.5920E 01
 -6.7336E-01 -7.4075E 01
  1.2090E 02 -2.0314E 03
  3.0057E 02 -5.6996E 01
 -2.9802E 01 -4.3939E 00
  3.3730E 04 -5.8417E 01
 -2.3988E-01 -3.8822E 00
CR =
-5.6621E-07 -2.8563E-06  6.7167E-05 -4.3107E-01  6.5199E-06  3.6167E-05  5.8821E-05 -2.6394E-05  1.3592E-02
 1.2342E-02  5.1938E-02 -1.2284E 00  7.8221E 03 -7.6378E-02 -9.1569E-01 -9.2350E-01  4.1135E-01 -2.3808E 02
 1.2342E-02  5.1938E-02 -1.2284E 00  7.8221E 03 -7.6378E-02 -9.1569E-01 -9.2350E-01  4.1135E-01 -2.3808E 02
DR =
  2.6715E-01 -2.4533E-01
  2.9270E 01  4.4590E 03
  2.9270E 01  4.4590E 03
```

REFERENCES

1. Sellers, James F.; and Daniele, Carl J.: DYNGEN - A Program for Calculating Steady-State and Transient Performance of Turbojet and Turbofan Engines. NASA TN D-7901, 1975.

2. Koenig, Robert W.; and Fishbach, Laurence H.: GENENG - A Program for Calculating Design and Off-Design Performance for Turbojet and Turbofan Engines. NASA TN D-6552, 1972.

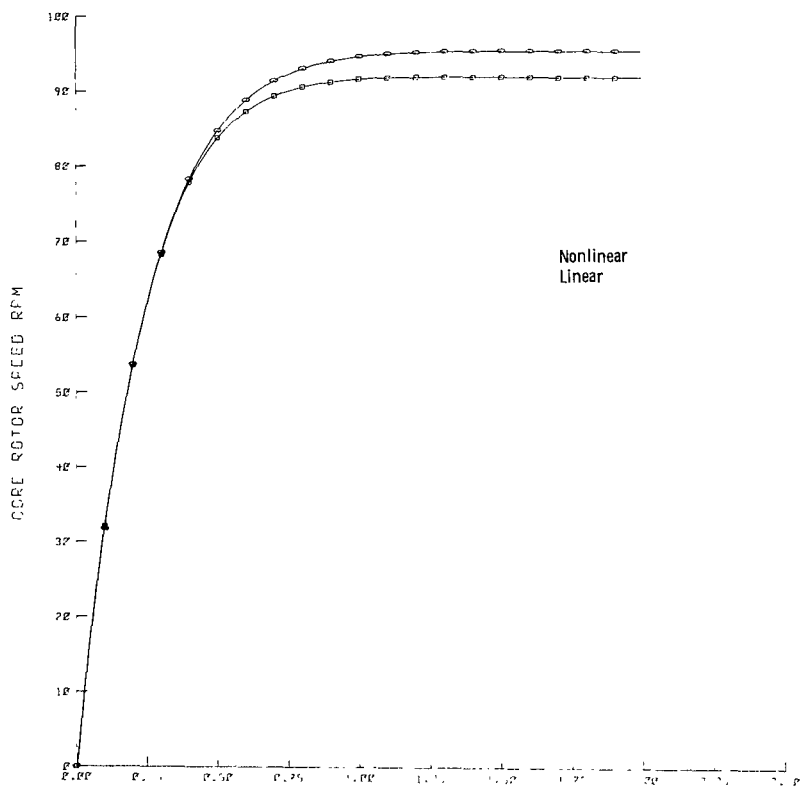
3. Fishbach, Laurence H.; and Koenig, Robert W.: GENENG II - A Program for Calculating Design and Off-Design Performance of Two- and Three-Spool Turbofans with as Many as Three Nozzles. NASA TN D-6553, 1972.

4. McKinney, John S.: Simulation of Turbofan Engine. Part I. Description of Method and Balancing Technique. AFAPL-TR-67-125-pt. 1, Air Force Aero Propulsion Lab., 1967. (AD-825197.)

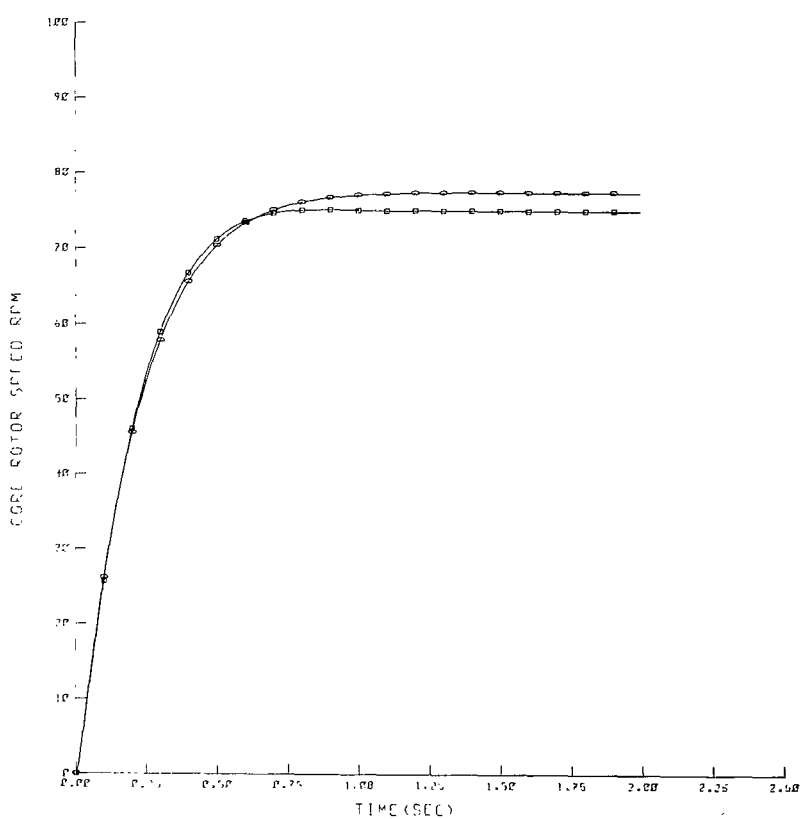
5. McKinney, John S.: Simulation of Turbofan Engine. Part II. User's Manual and Computer Program Listing. AFAPL-TR-67-125-pt. 2, Air Force Aero Propulsion Lab., 1967. (AD-825198.)

6. Zadeh, Lotfi A.; and Desoer, Charles A.: Linear System Theory; the State Space Approach. McGraw-Hill Book Co., 1963.

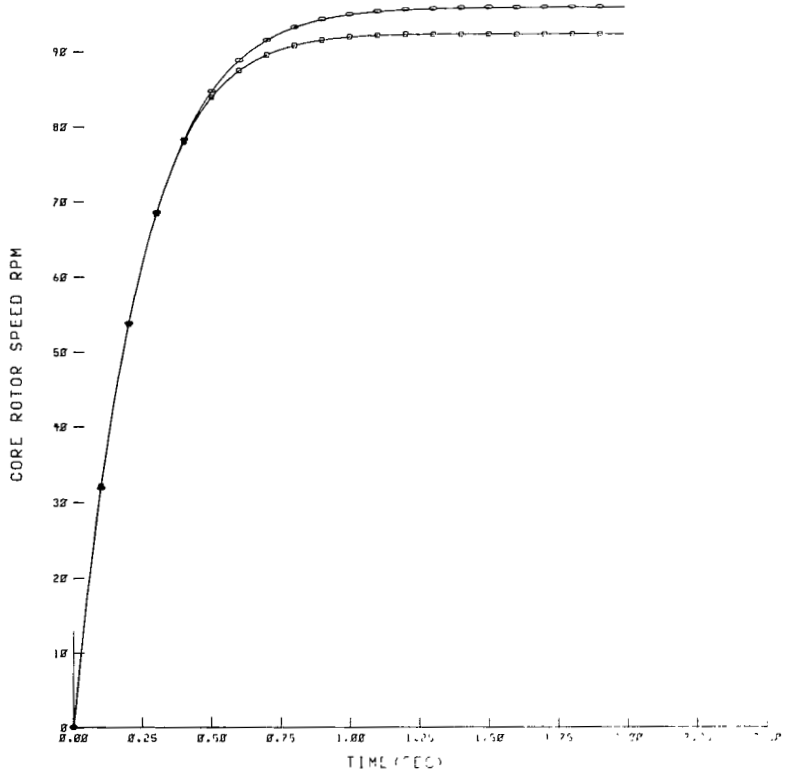
7. Weinberg, Marc S.; and Adams, Gary R.: Low Order Linearized Models of Turbine Engines. ASD-TR-75-24, Aeronautical Systems Division, 1975. (AD-A018841.)



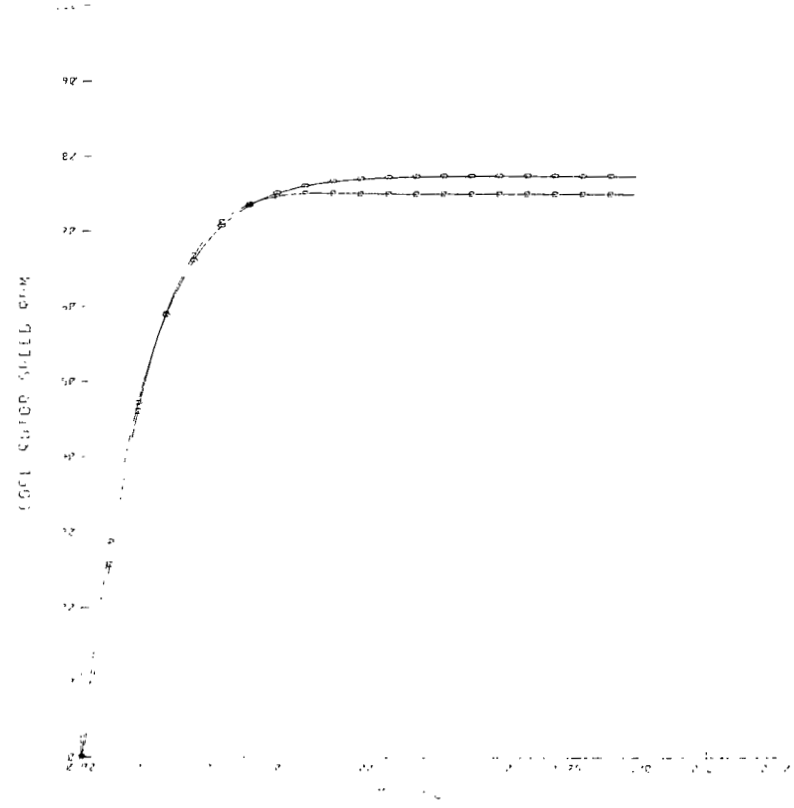
(a-1) With 3-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



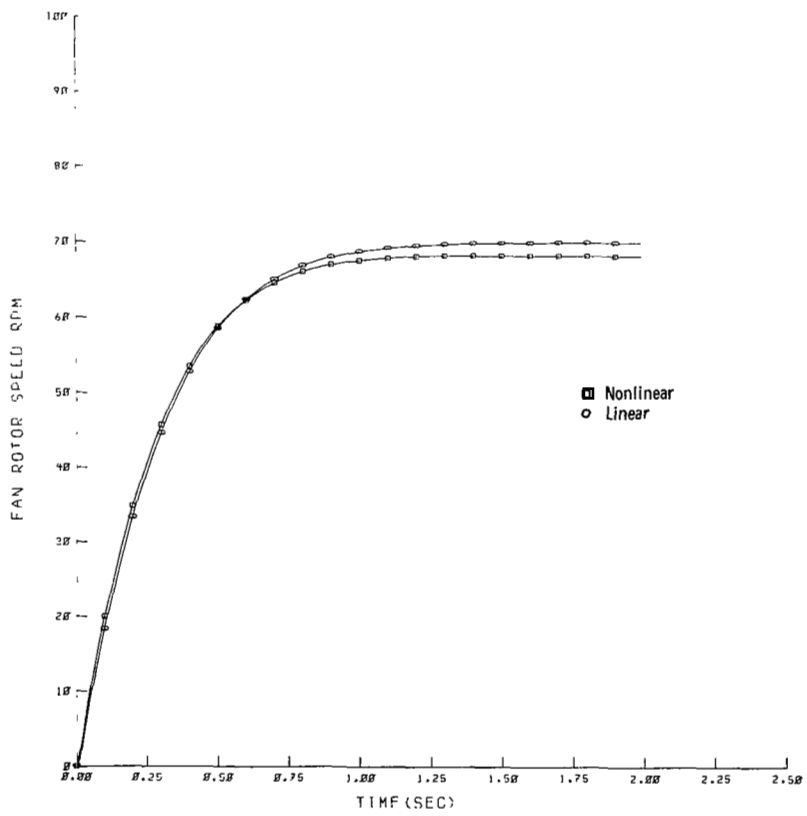
(b-1) With 3-percent step change in main fuel flow.



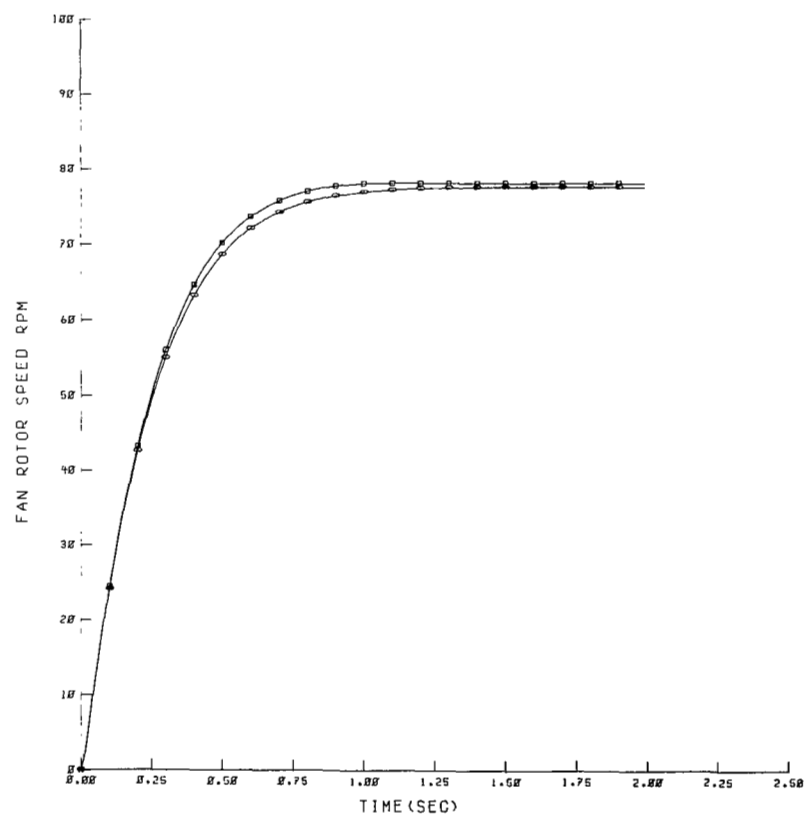
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 1. - Response of state 1 - core rotor speed.

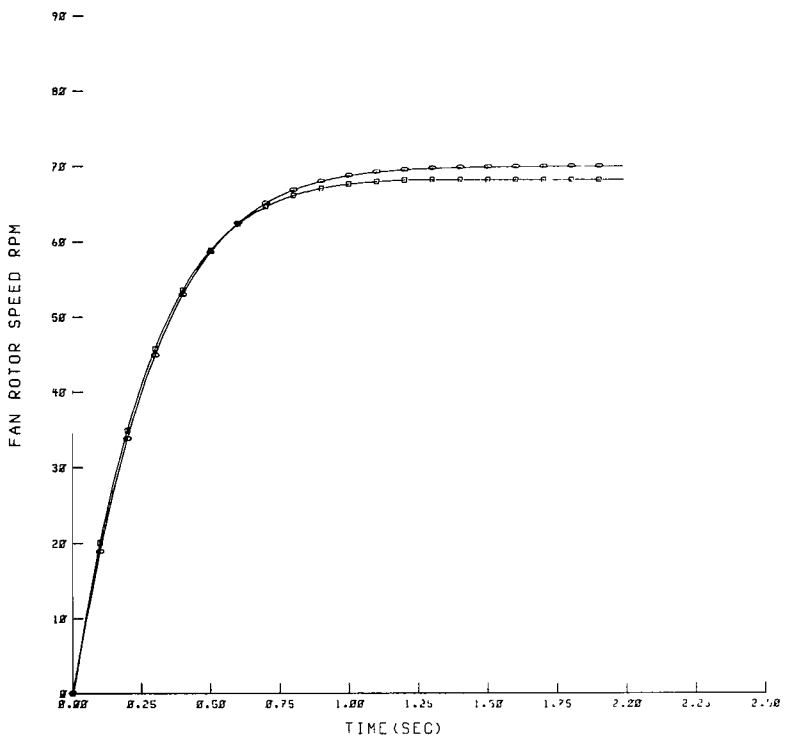


(a-1) With 3-percent step change in main fuel flow.

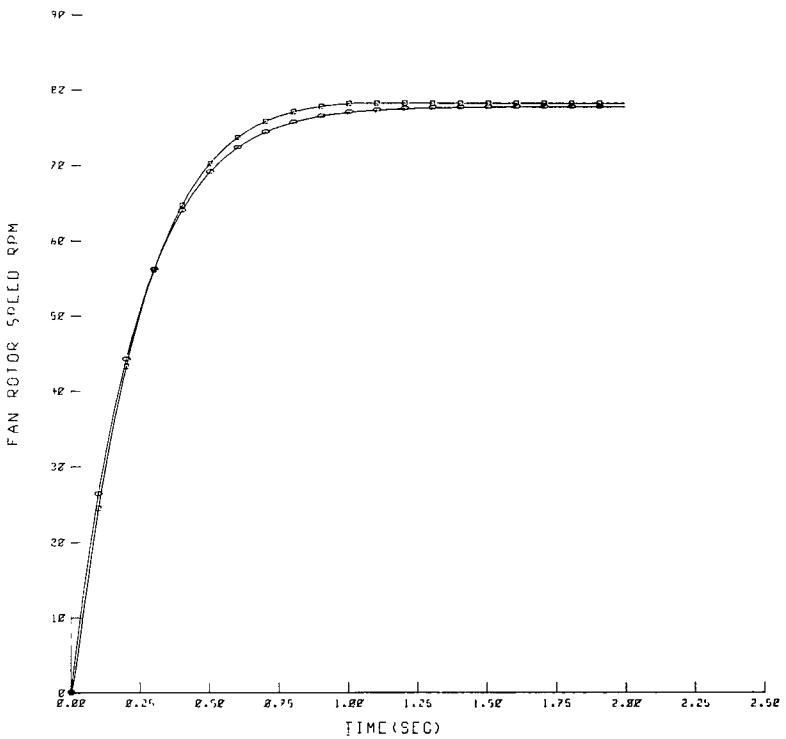


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



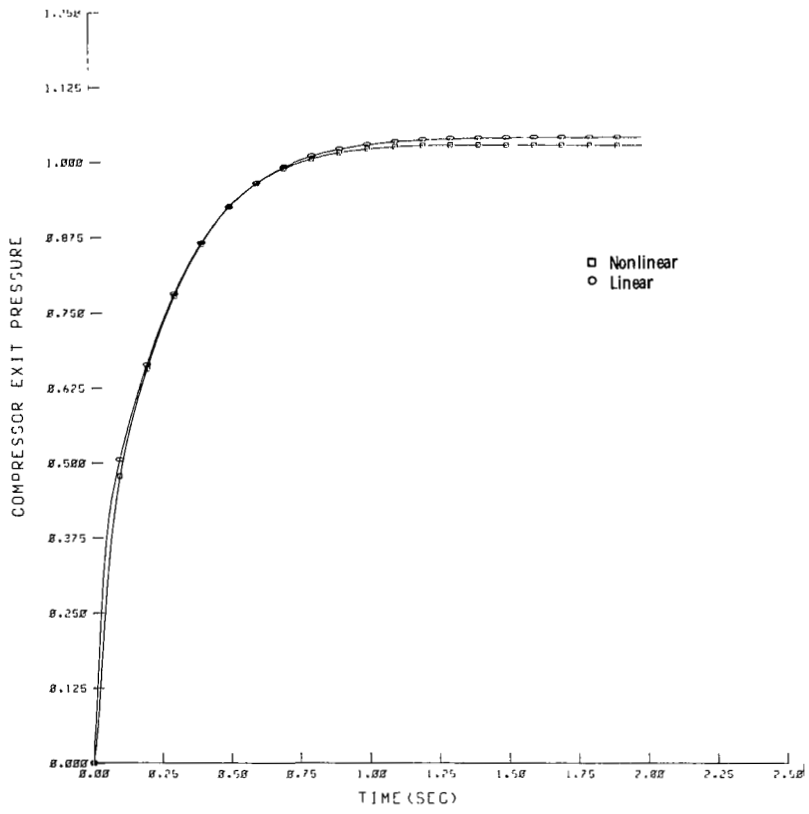
(b-1) With 3-percent step change in main fuel flow.



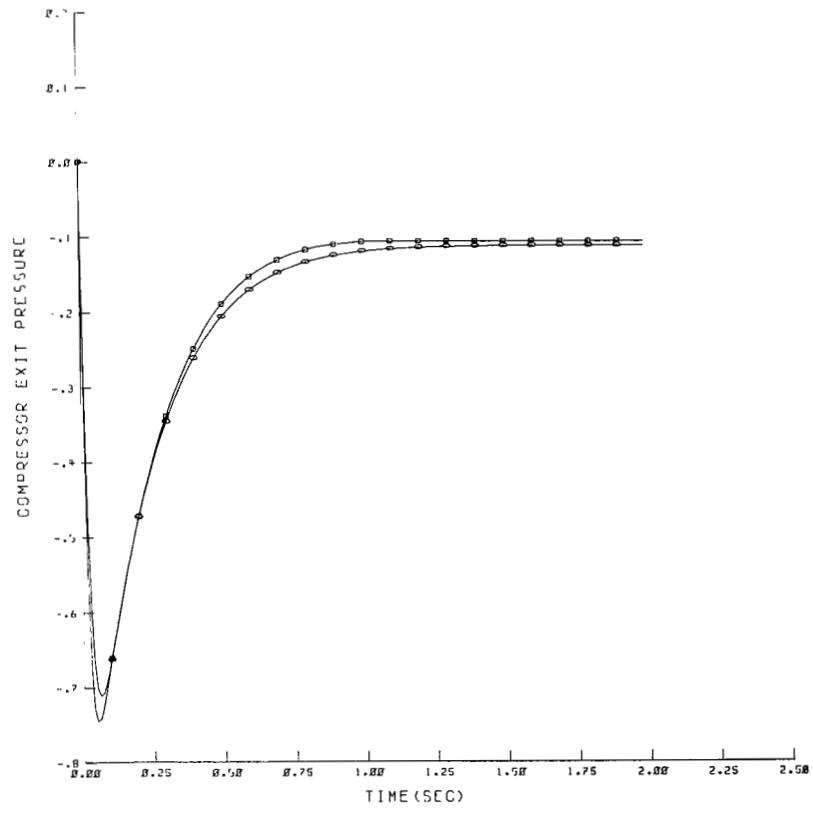
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 2. - Response of state 2 - fan rotor speed.

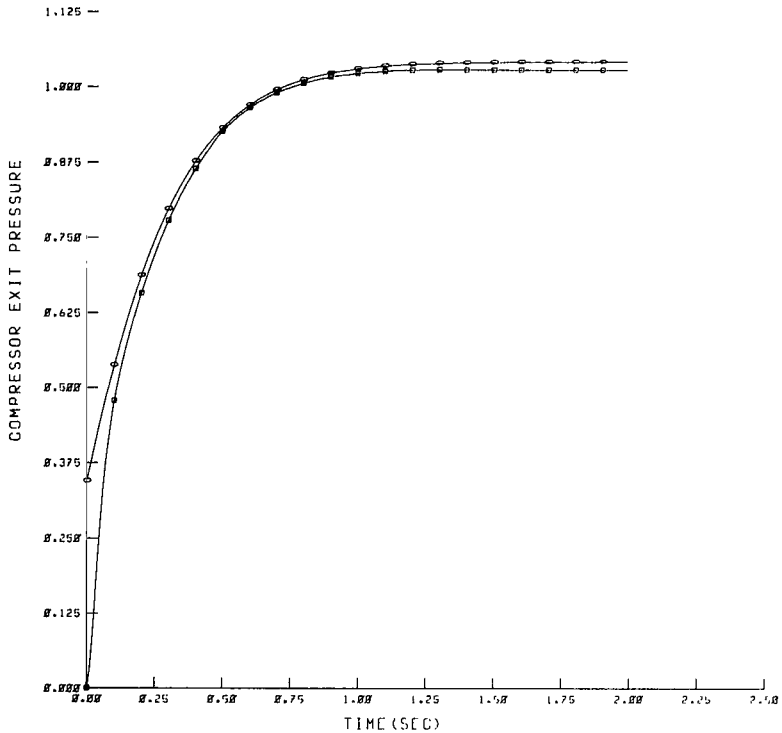


(a-1) With 3-percent step change in main fuel flow.

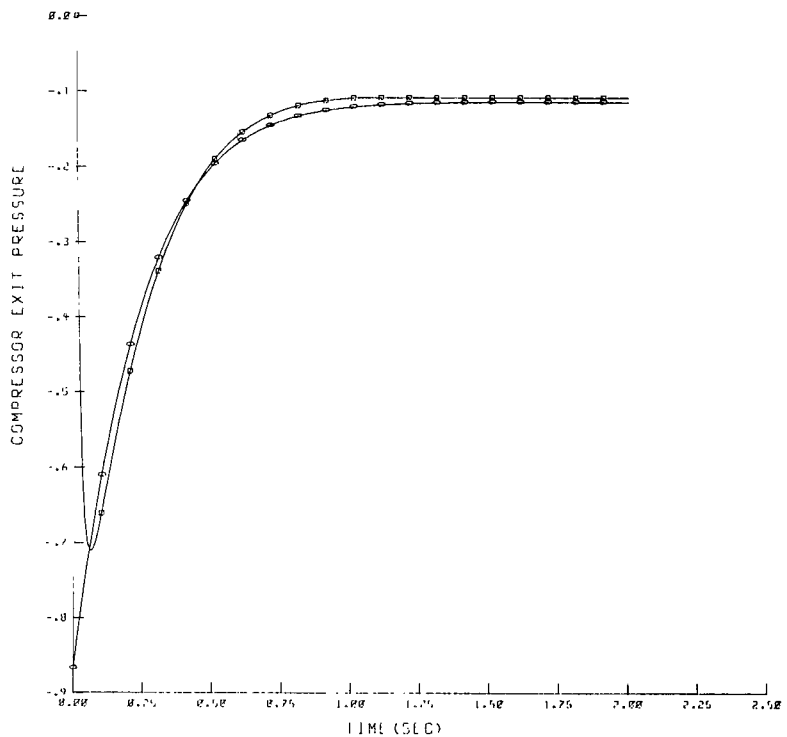


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



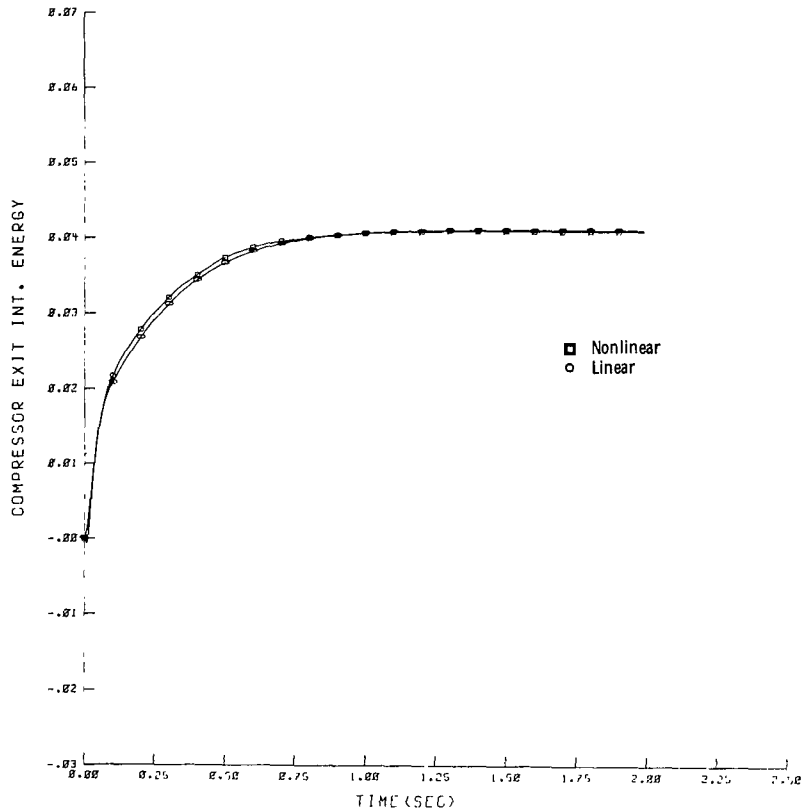
(b-1) With 3-percent step change in main fuel flow.



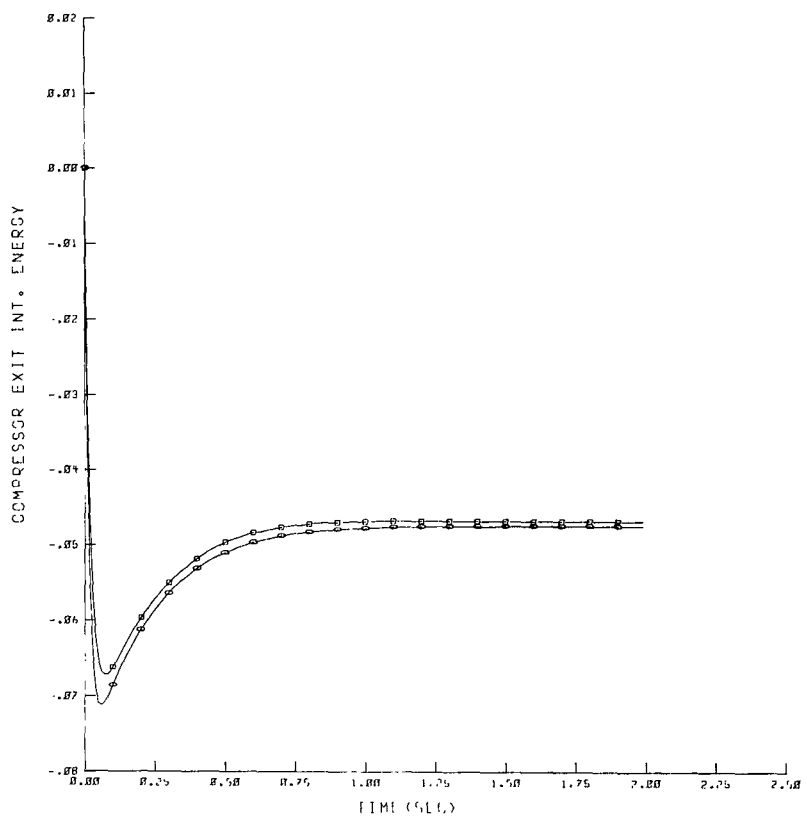
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 3. - Response of state 3 - compressor-exit pressure.

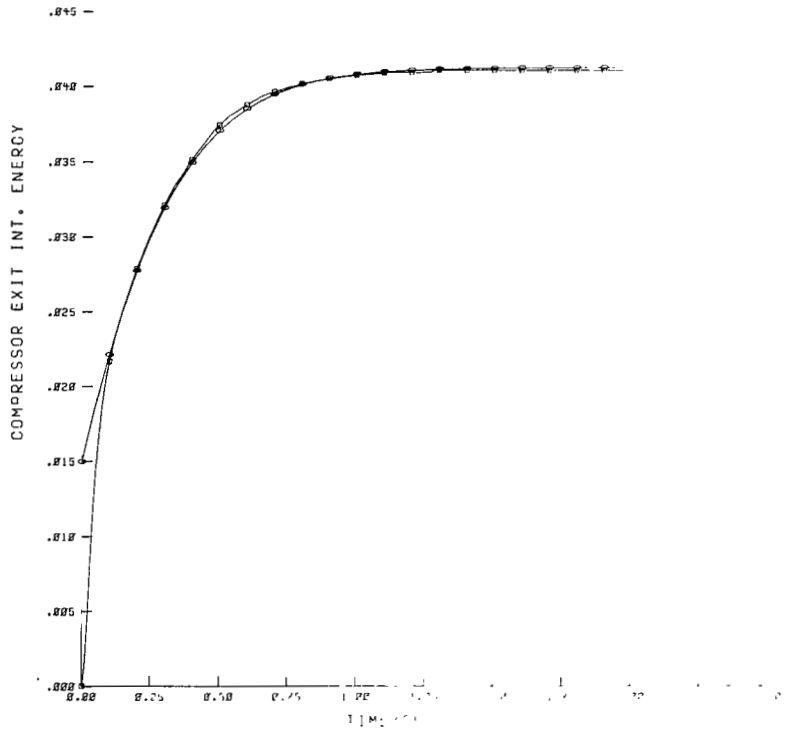


(a-1) With 3-percent step change in main fuel flow.

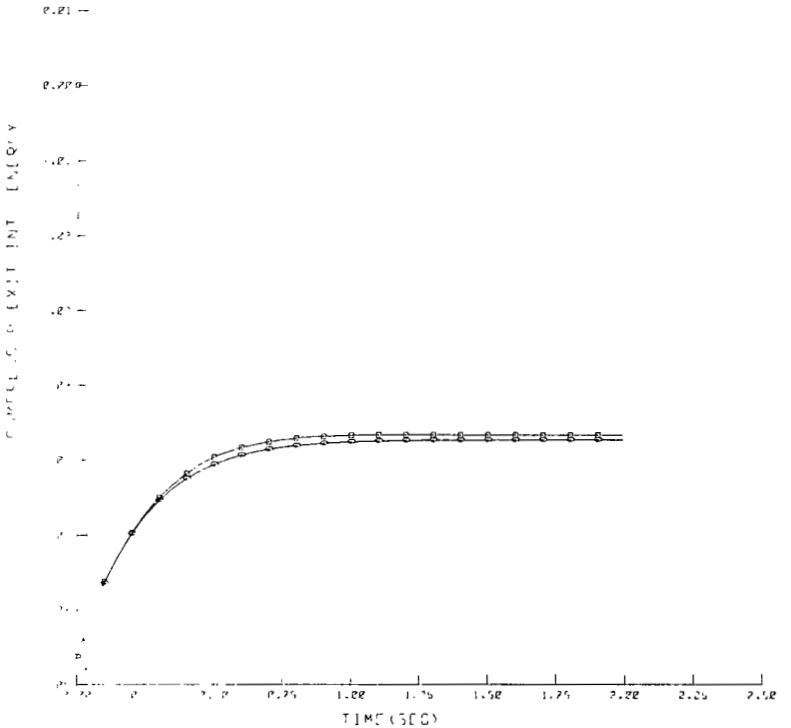


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



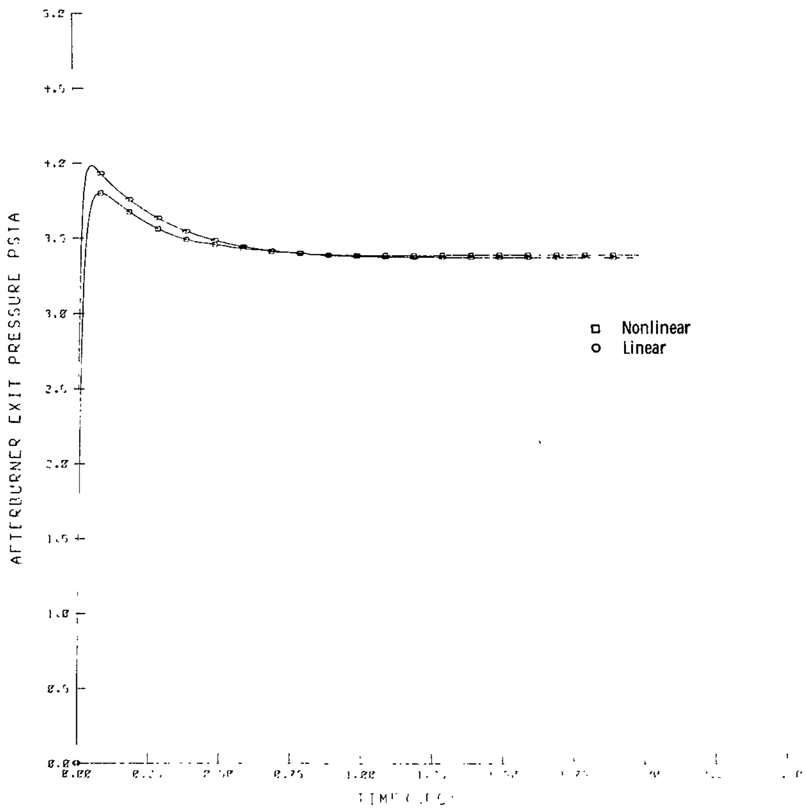
(b-1) With 3-percent step change in main fuel flow.



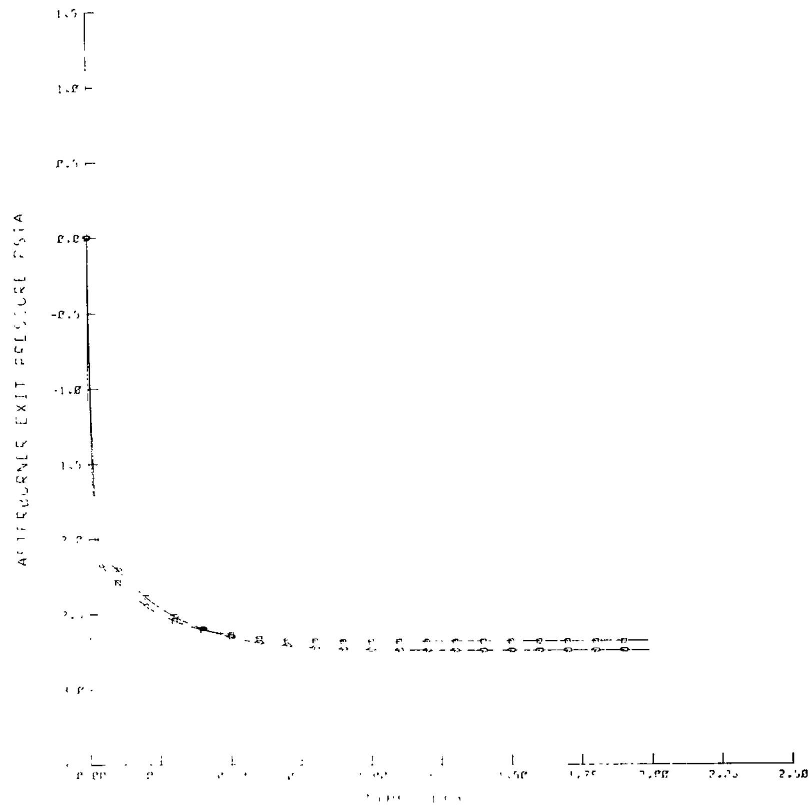
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 4. - Response of state 4 - compressor-exit internal energy.

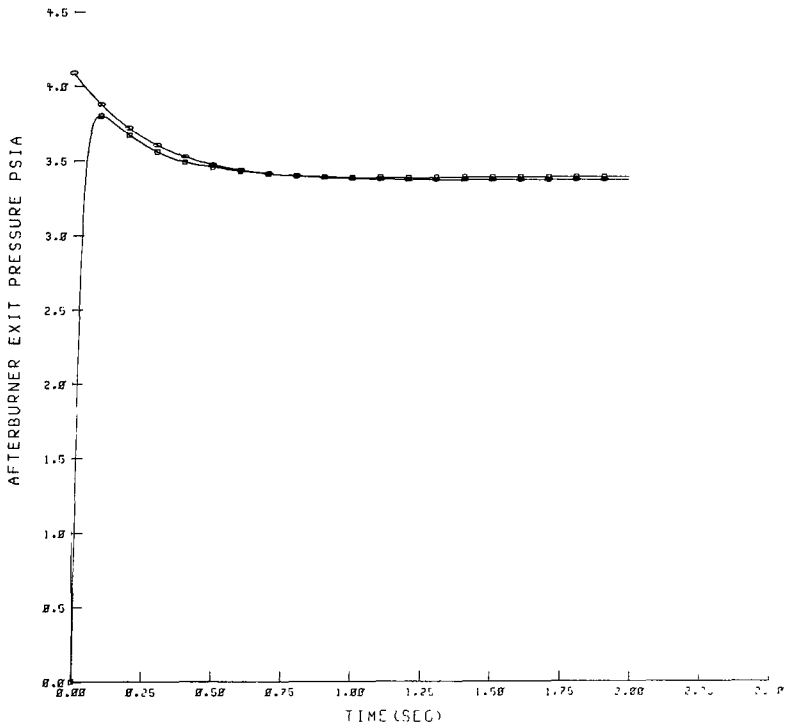


(a-1) With 3-percent step change in main fuel flow.

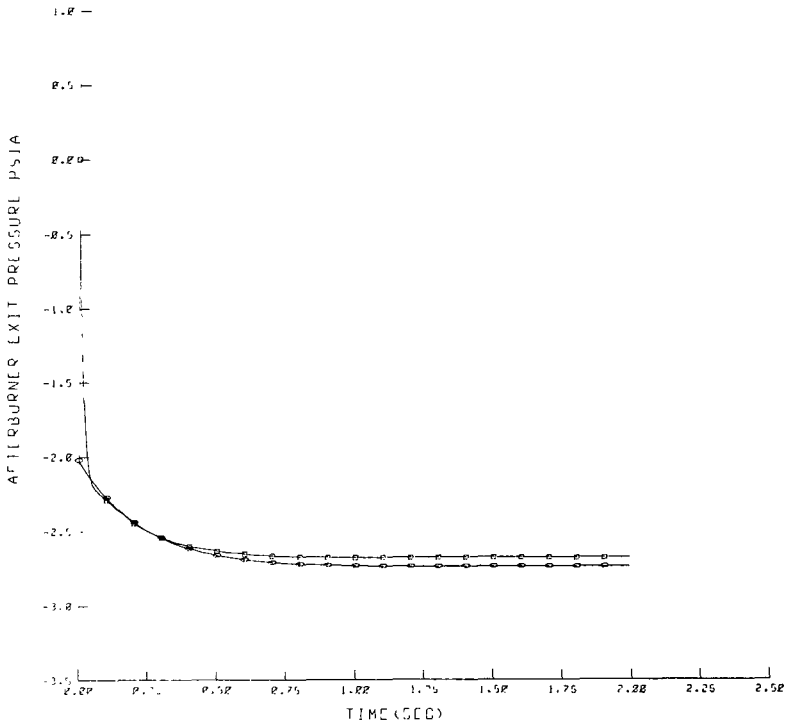


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



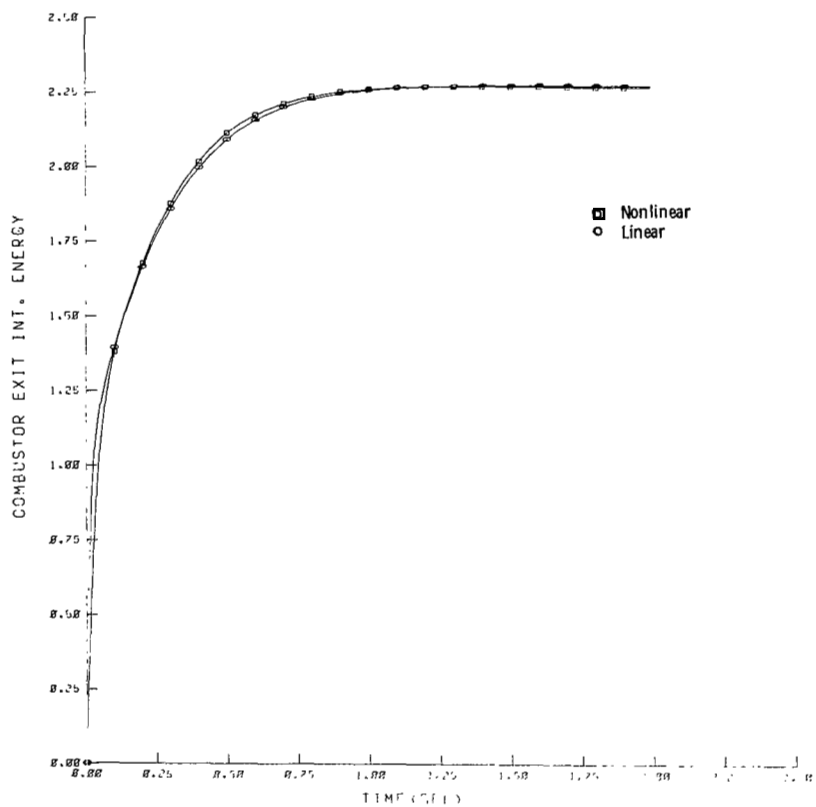
(b-1) With 3-percent step change in main fuel flow.



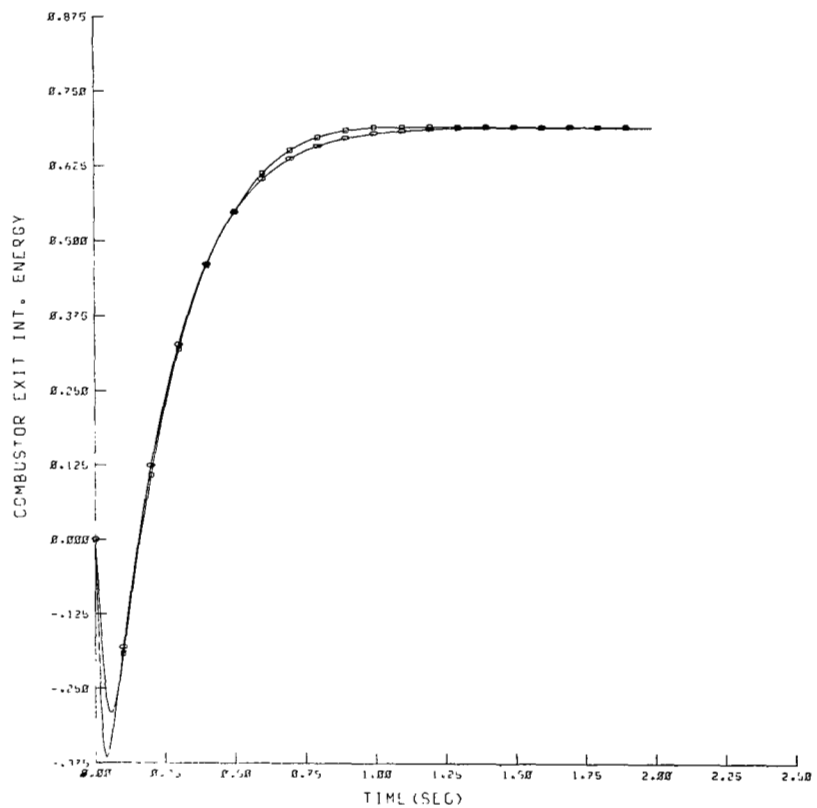
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

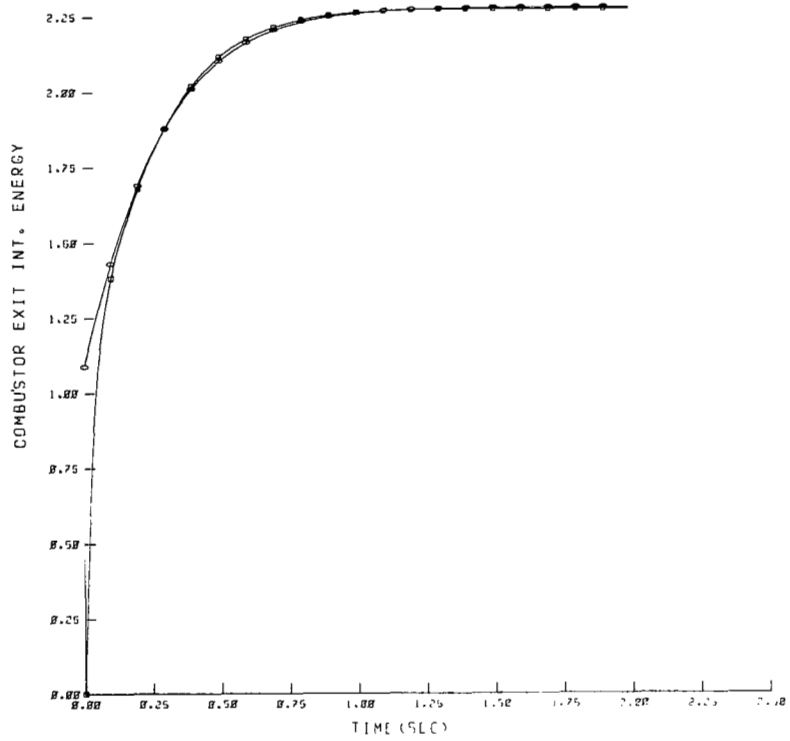
Figure 5. - Response of state 5 - afterburner-exit pressure.



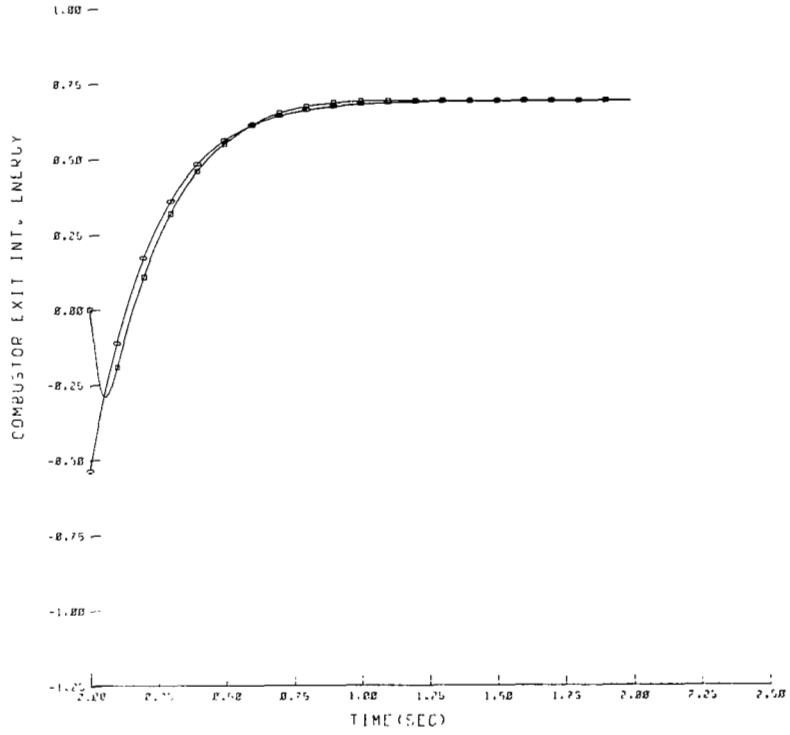
(a-1) With 3-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



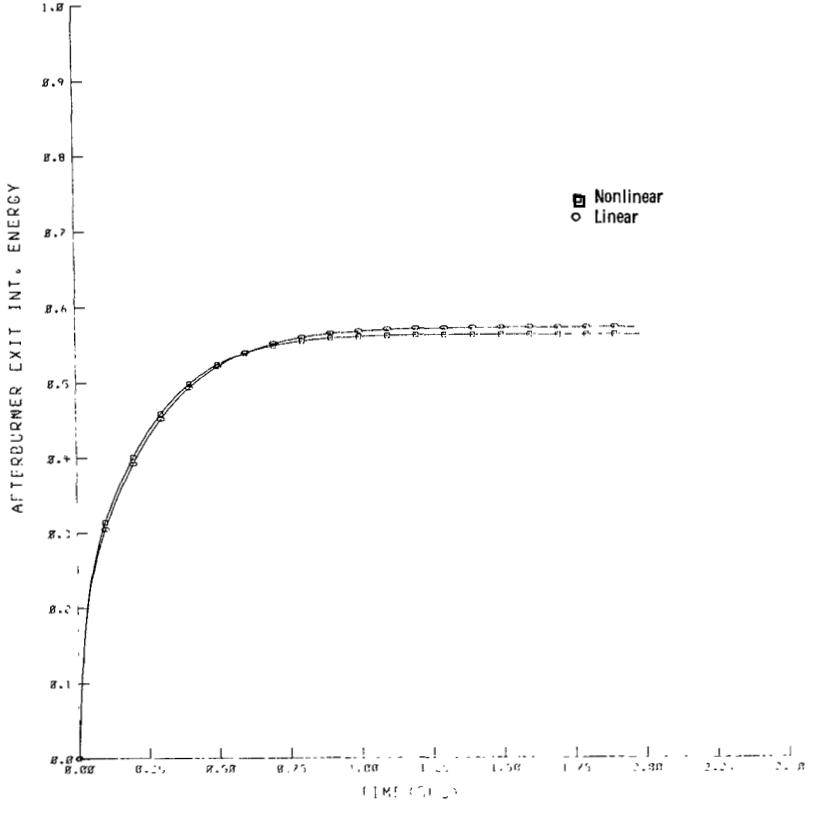
(b-1) With 3-percent step change in main fuel flow.



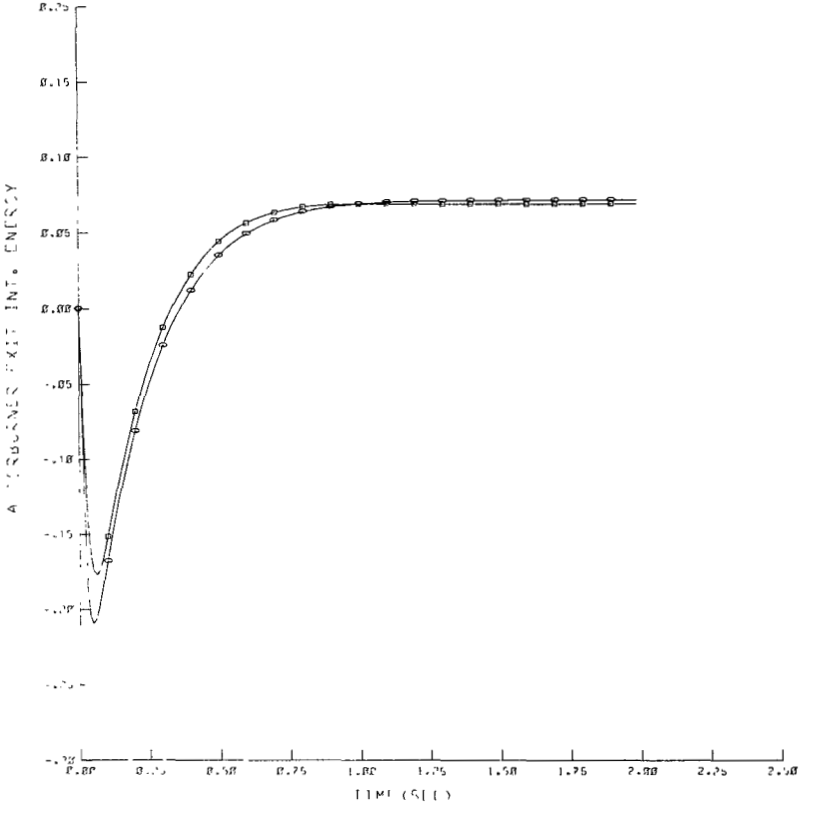
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 6. - Response of state 6 - combustor-exit internal energy.

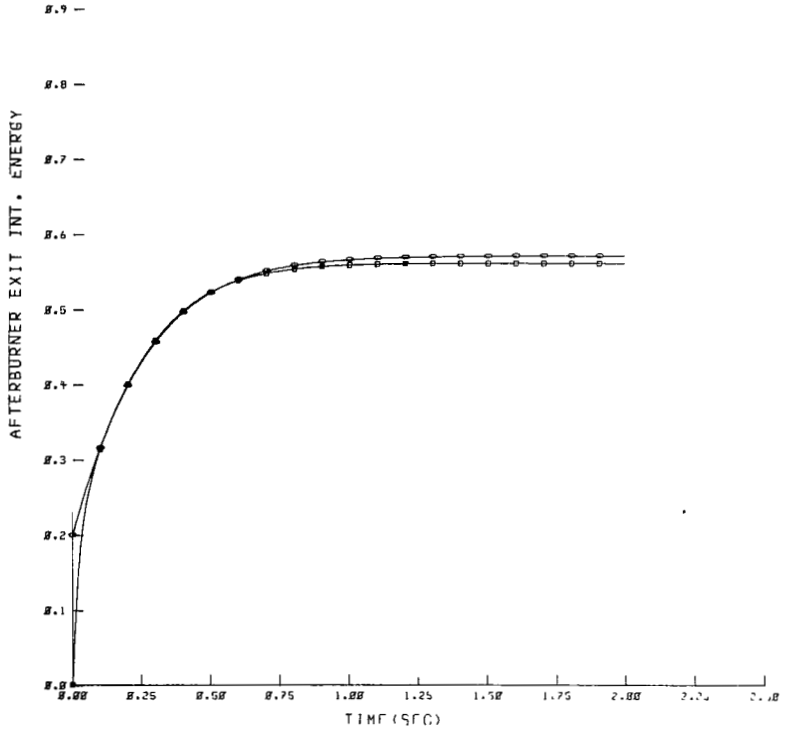


(a-1) With 3-percent step change in main fuel flow.

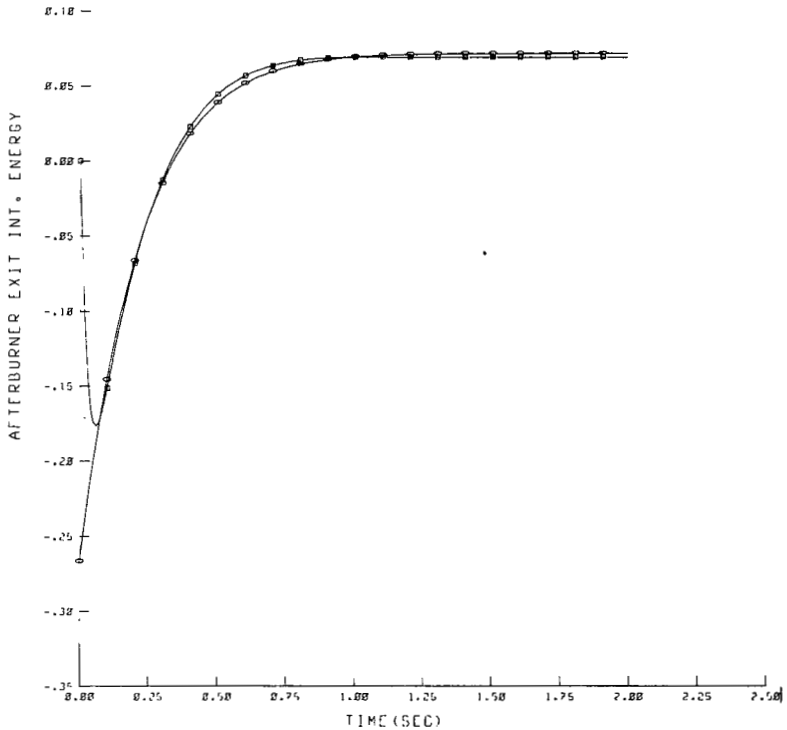


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs



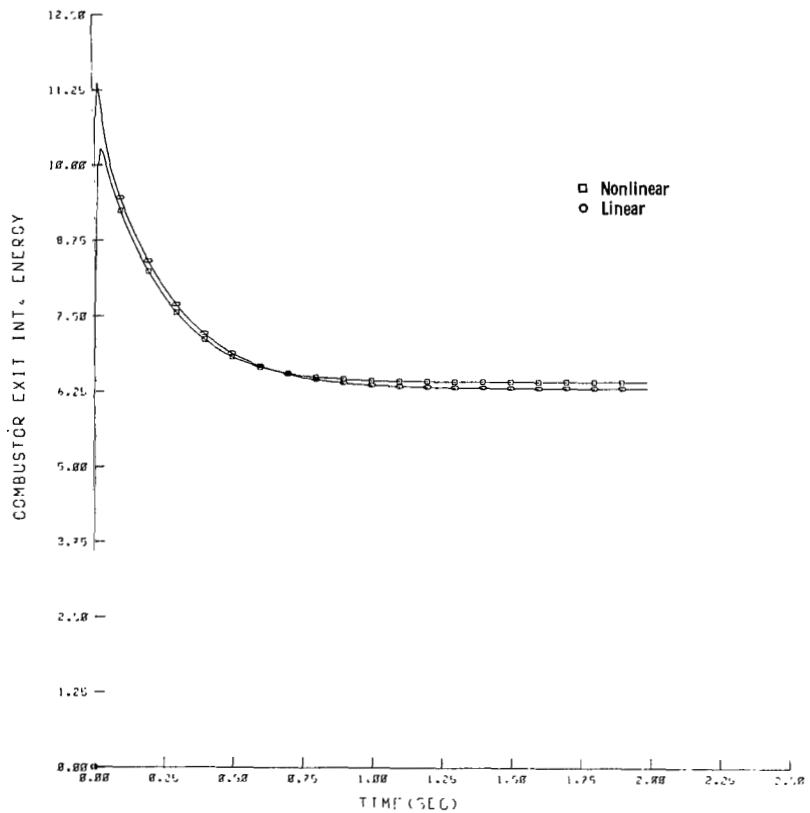
(b-1) With 3-percent step change in main fuel flow.



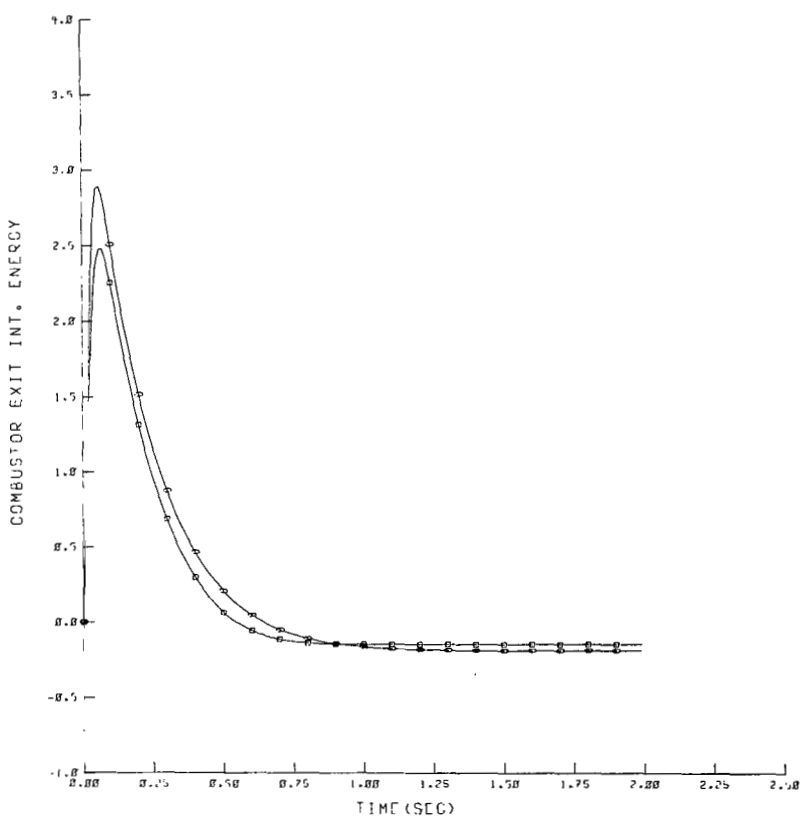
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

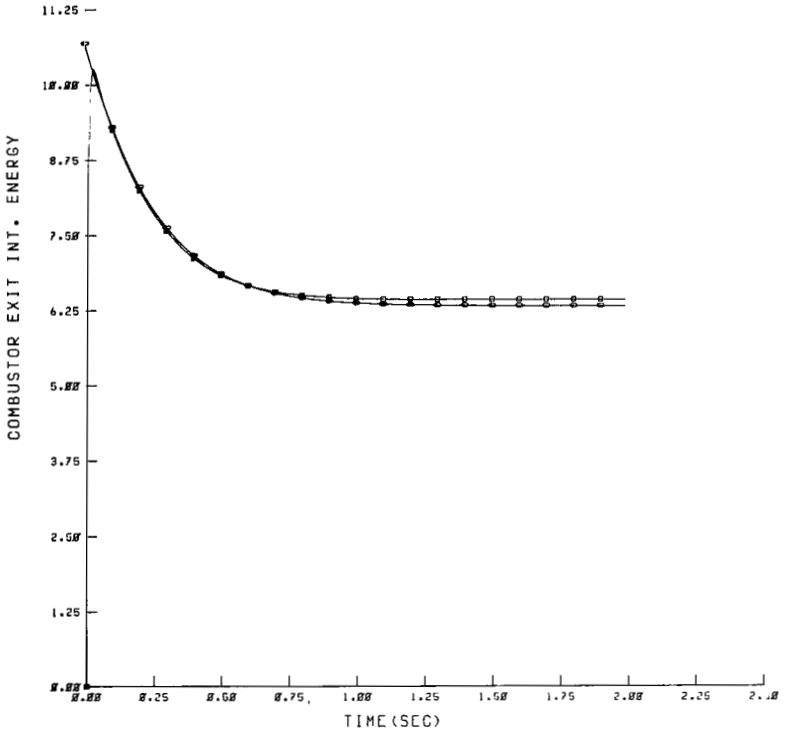
Figure 7. - Response of state 7 - afterburner-exit internal energy.



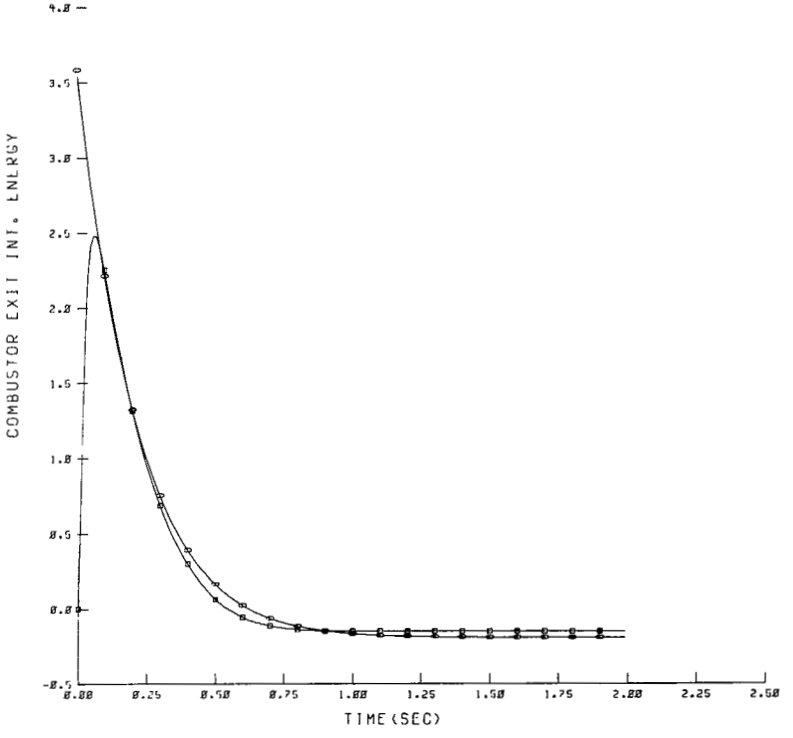
(a-1) With 3-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



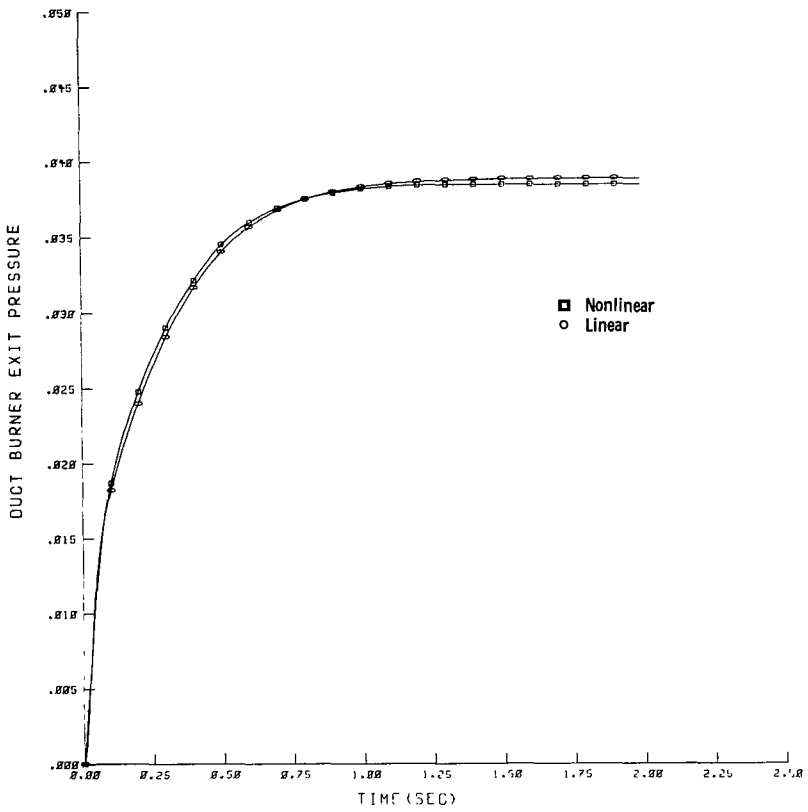
(b-1) With 3-percent step change in main fuel flow.



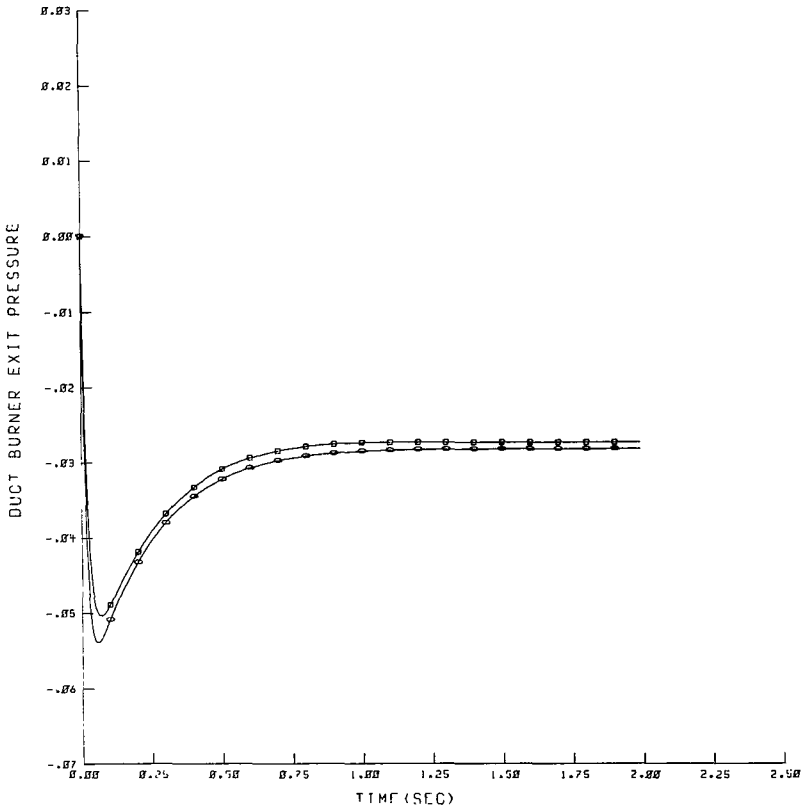
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 8. - Response of state 8 - combustor-exit internal energy.

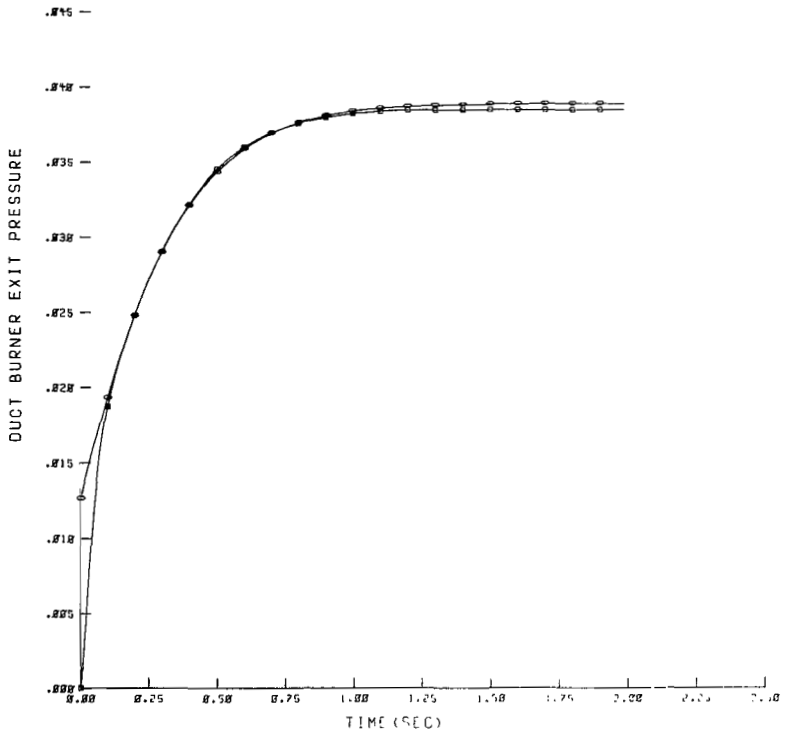


(a-1) With 3-percent step change in main fuel flow.

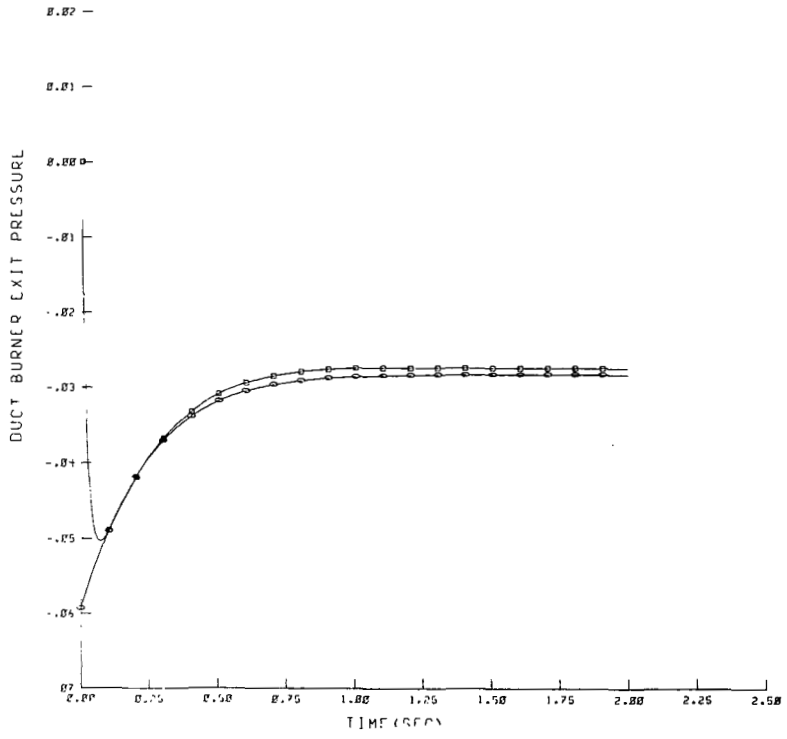


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



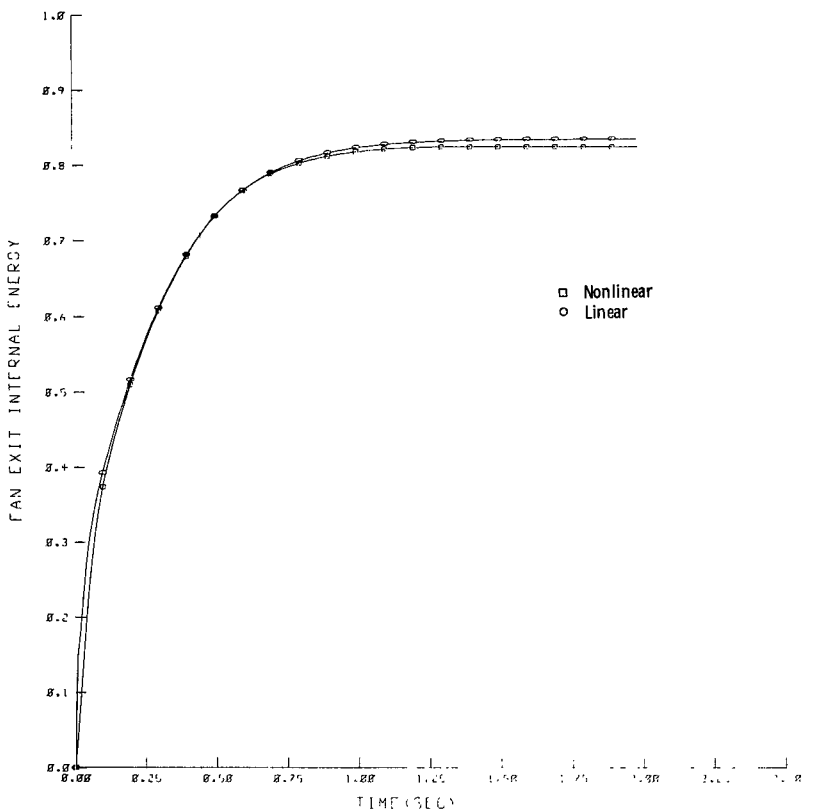
(b-1) With 3-percent step change in main fuel flow.



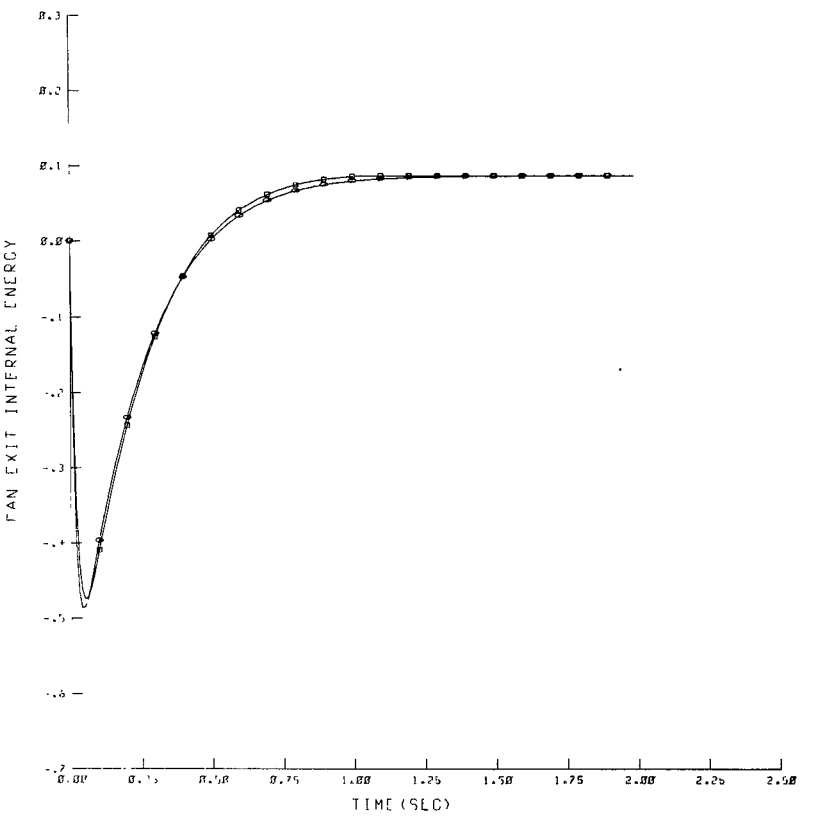
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 9. - Response of state 9 - duct-burner-exit pressure.

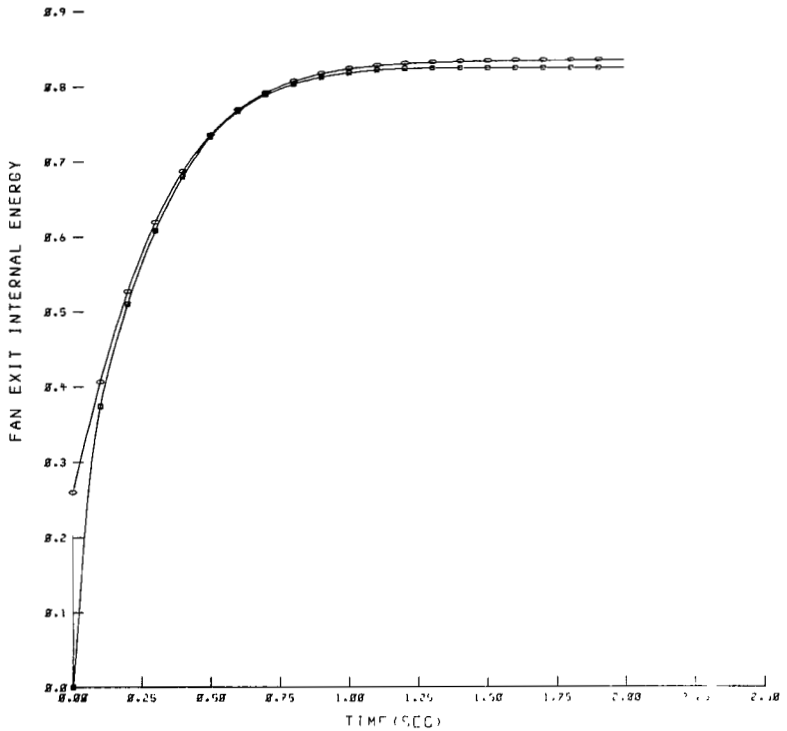


(a-1) With 3-percent step change in main fuel flow.

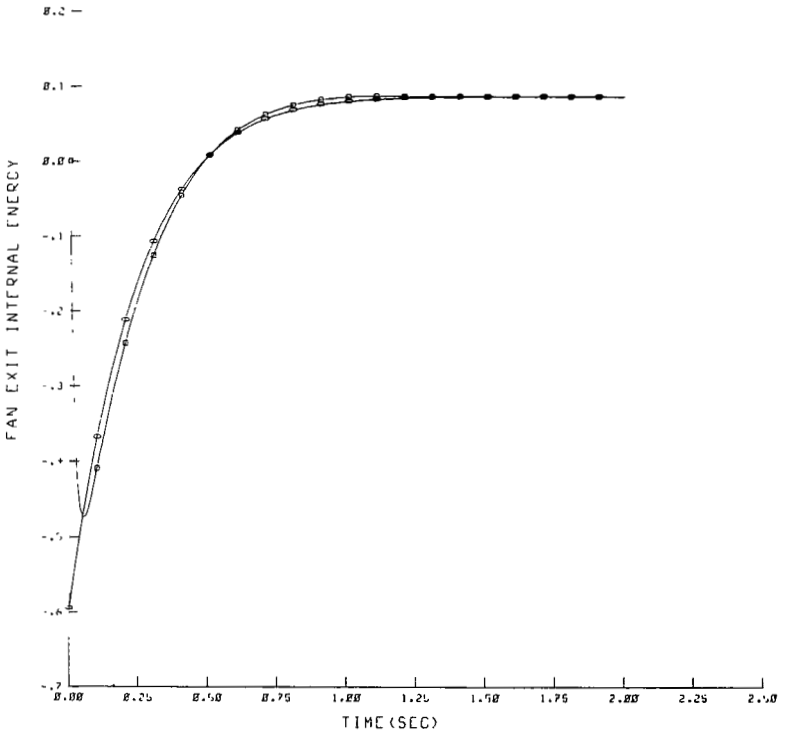


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



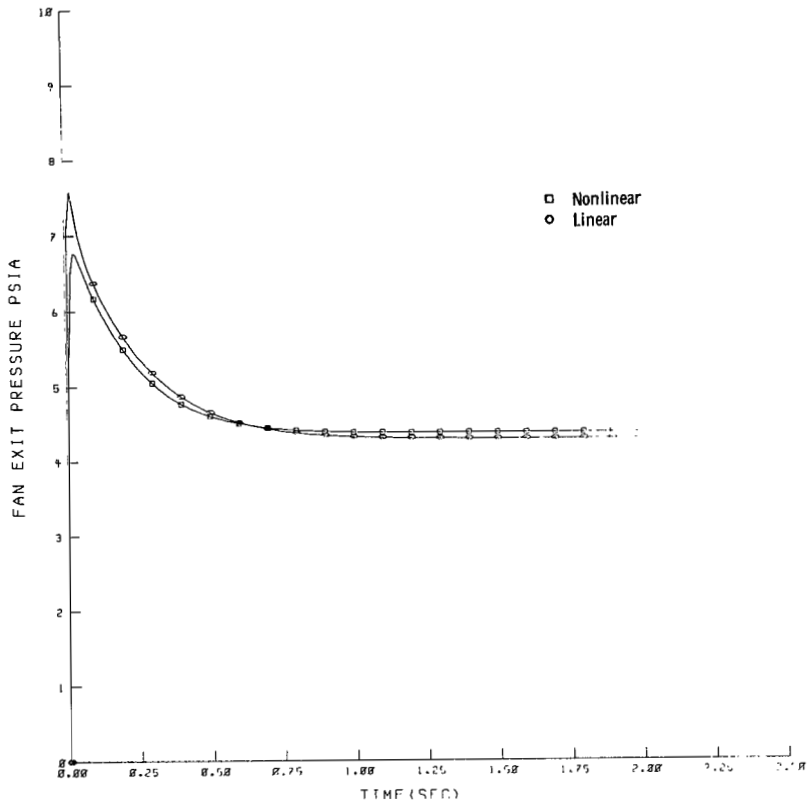
(b-1) With 3-percent step change in main fuel flow.



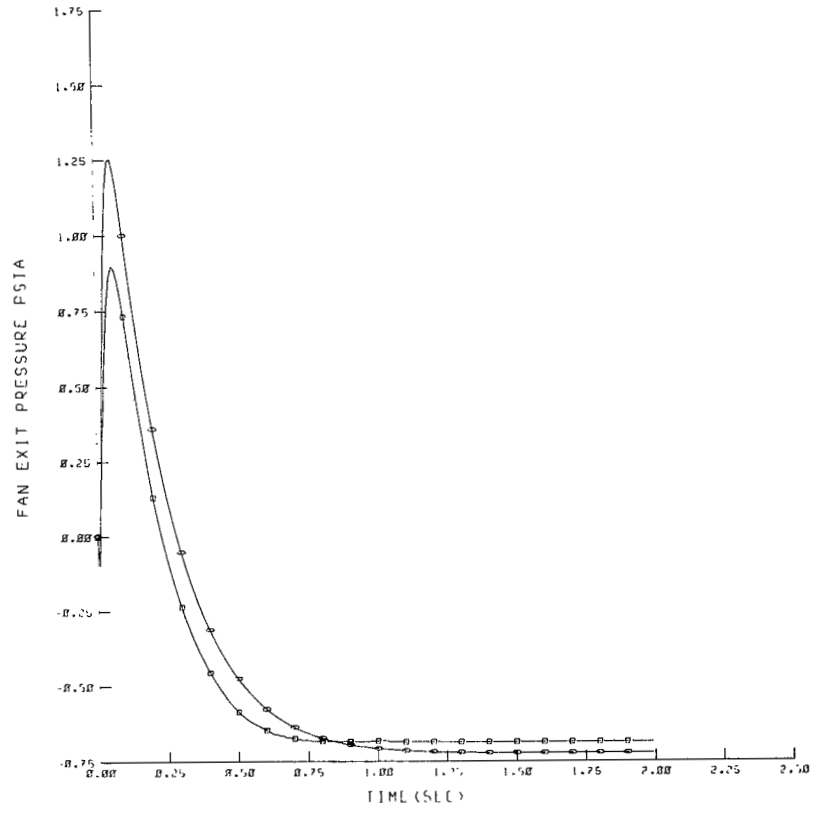
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 10. - Response of state 10 - fan-exit internal energy.

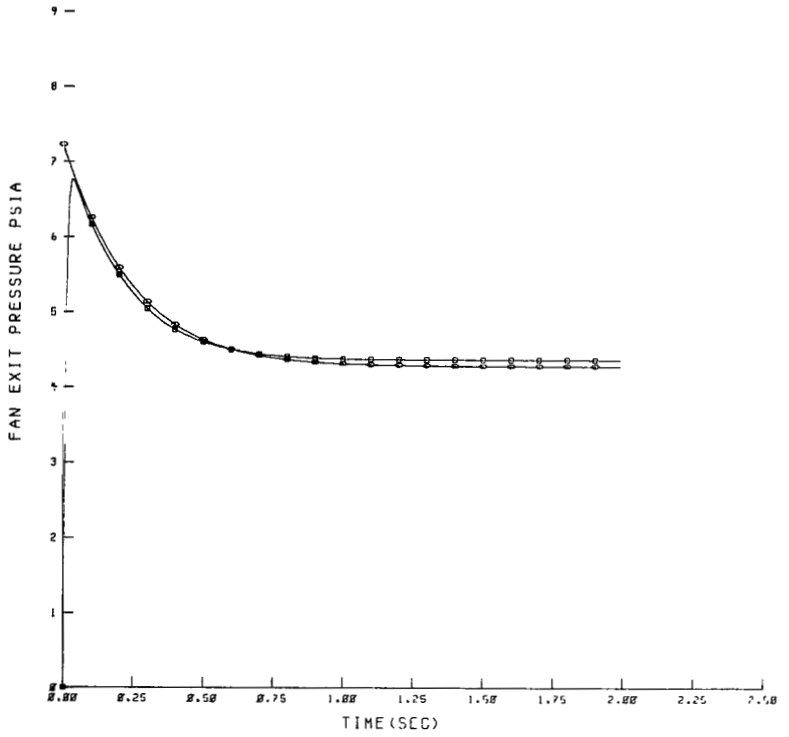


(a-1) With 3-percent step change in main fuel flow.

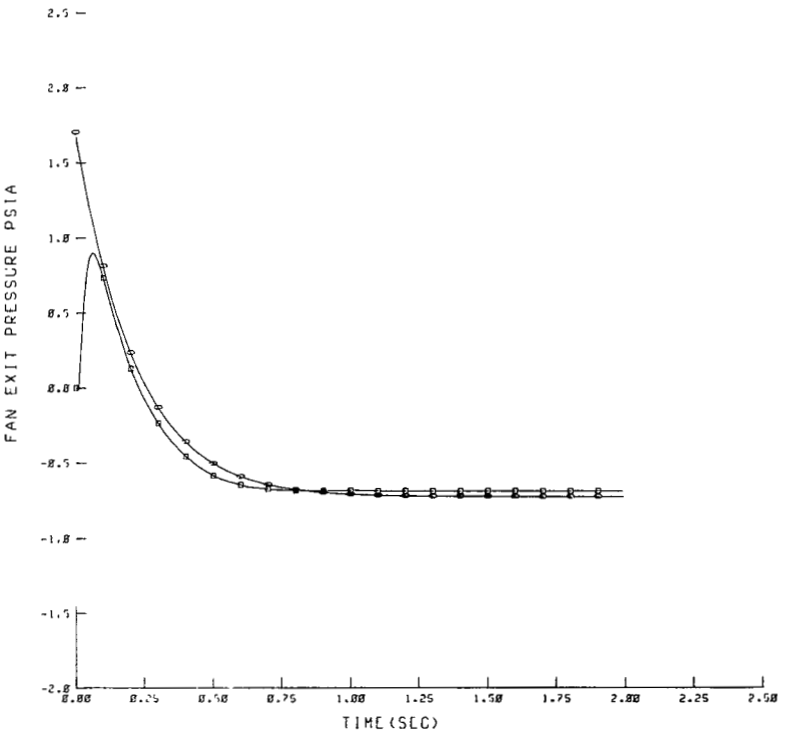


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



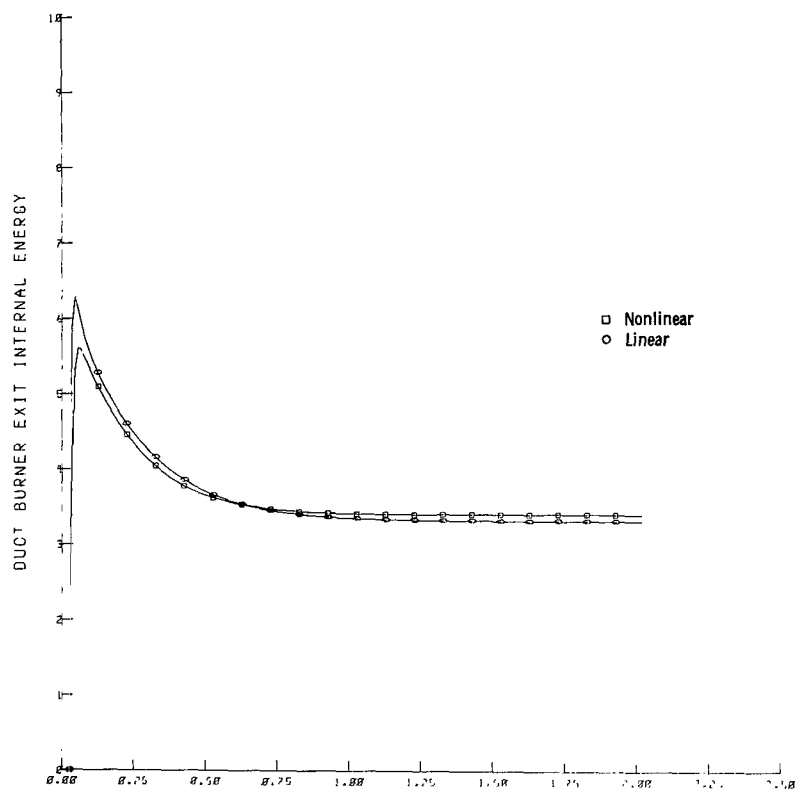
(b-1) With 3-percent step change in main fuel flow.



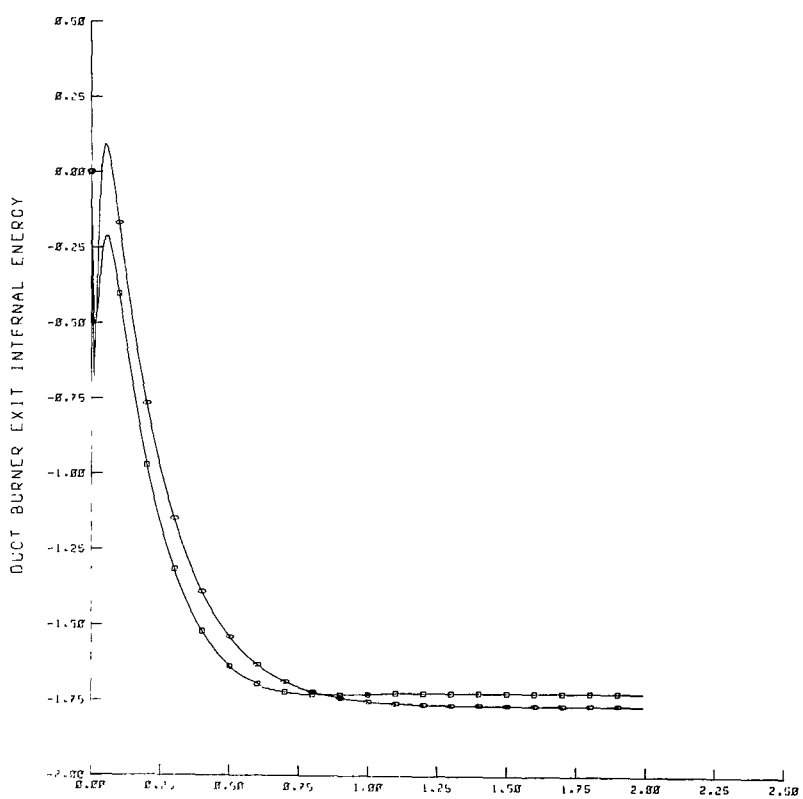
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 11. - Response of state 11 - fan-exit pressure.

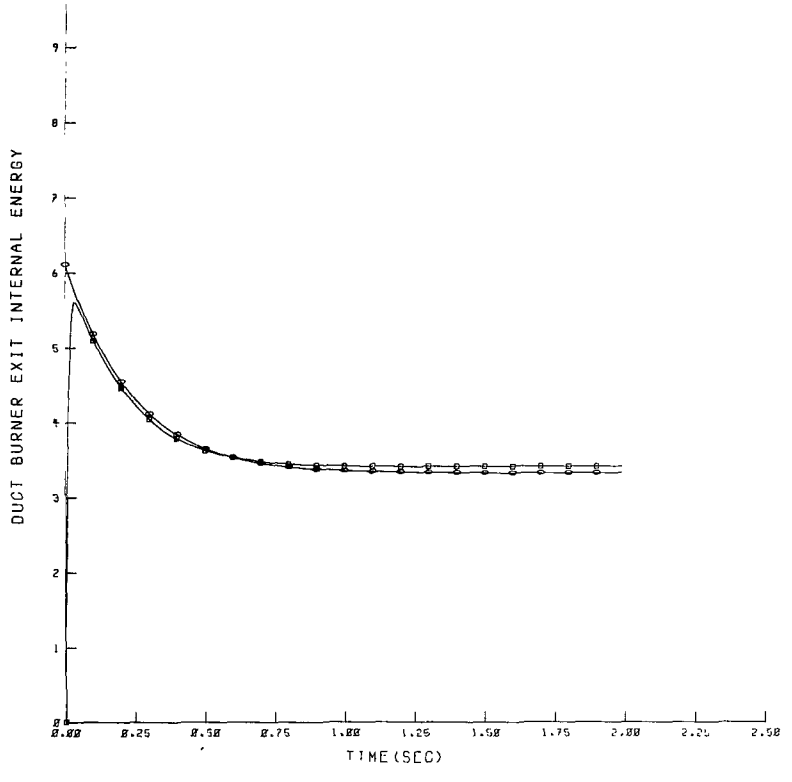


(a-1) With 3-percent step change in main fuel flow.

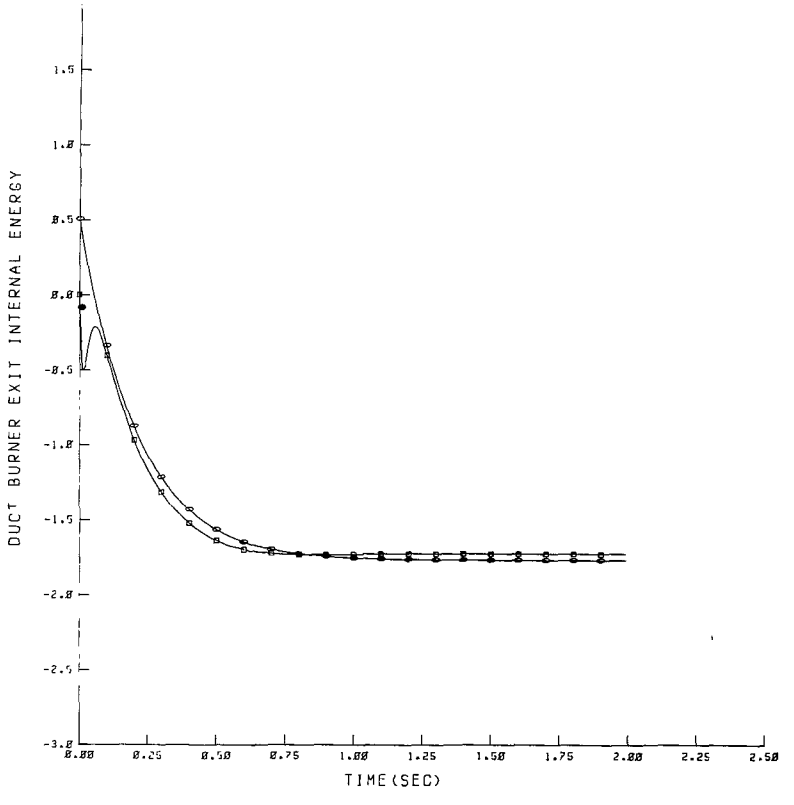


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of 11-order linear and nonlinear runs.



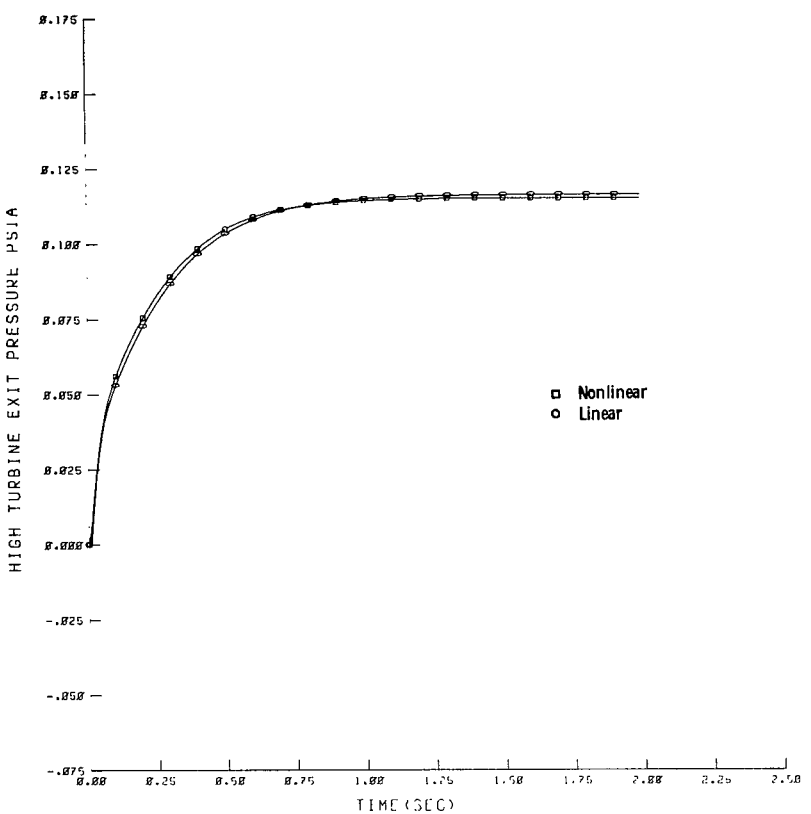
(b-1) With 3-percent step change in main fuel flow.



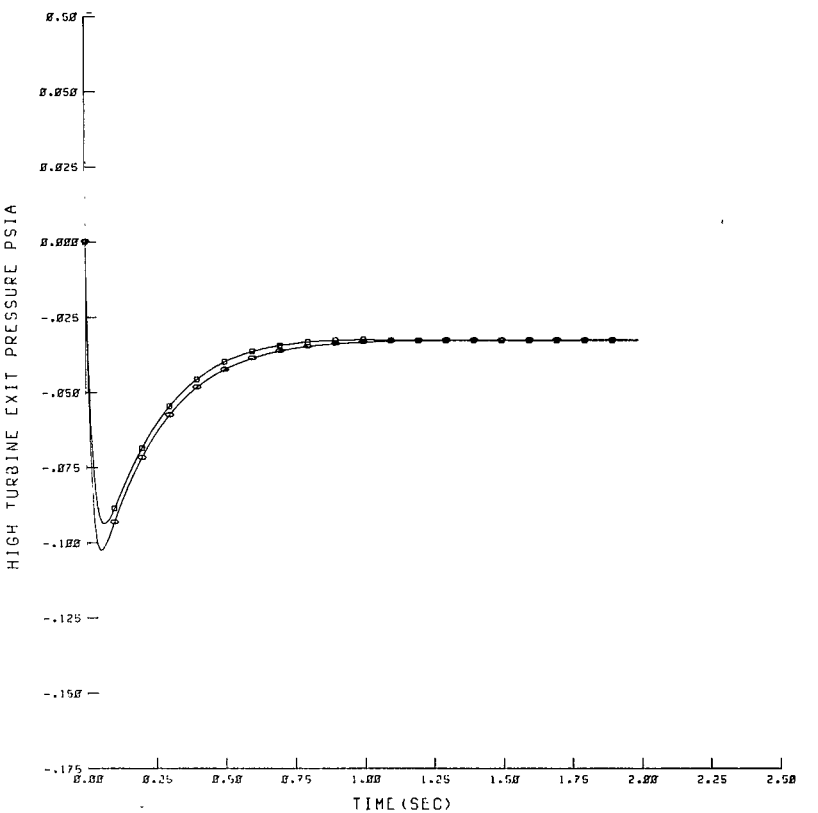
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

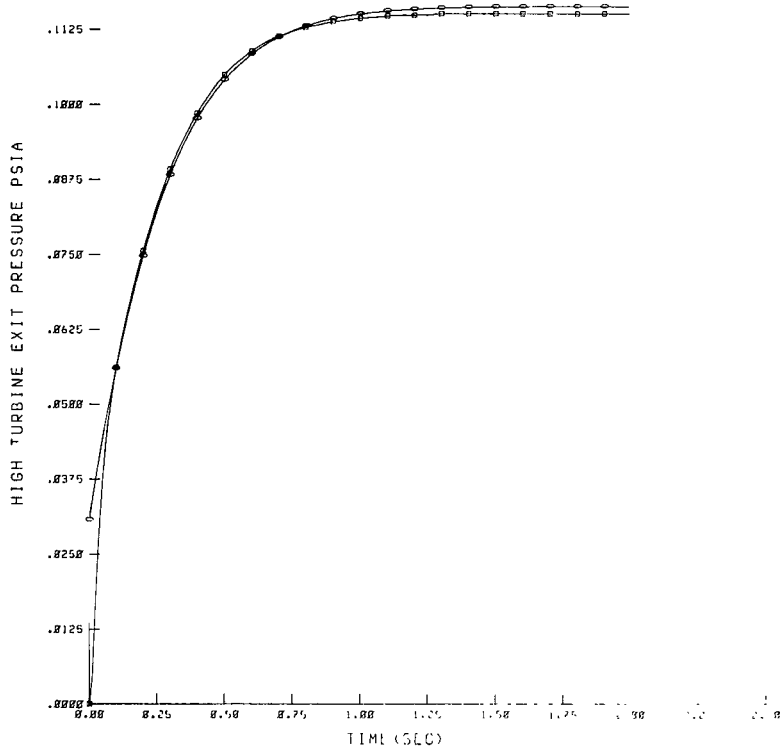
Figure 12. - Response of state 12 - duct-burner-exit internal energy.



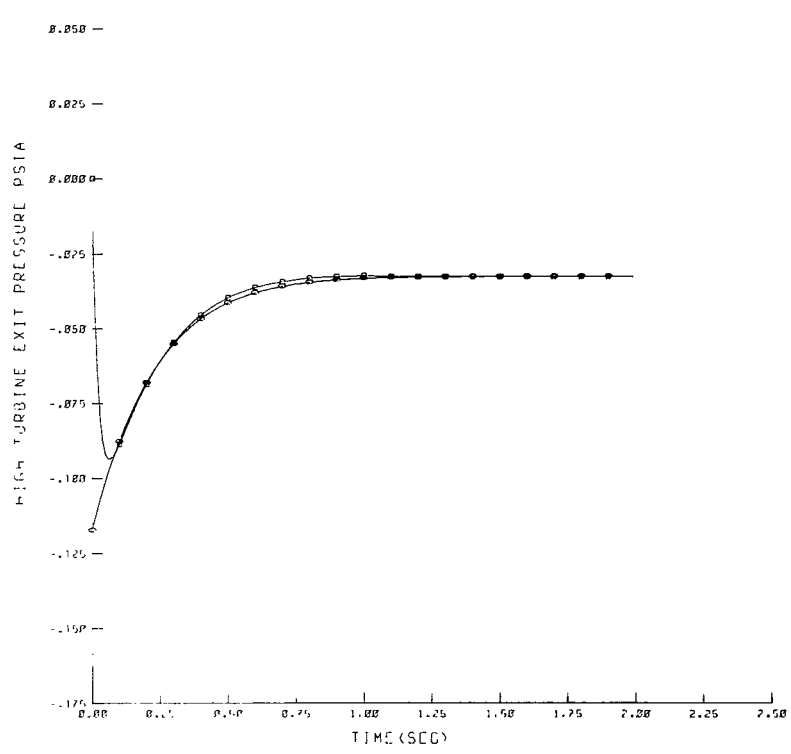
(a-1) With 5-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



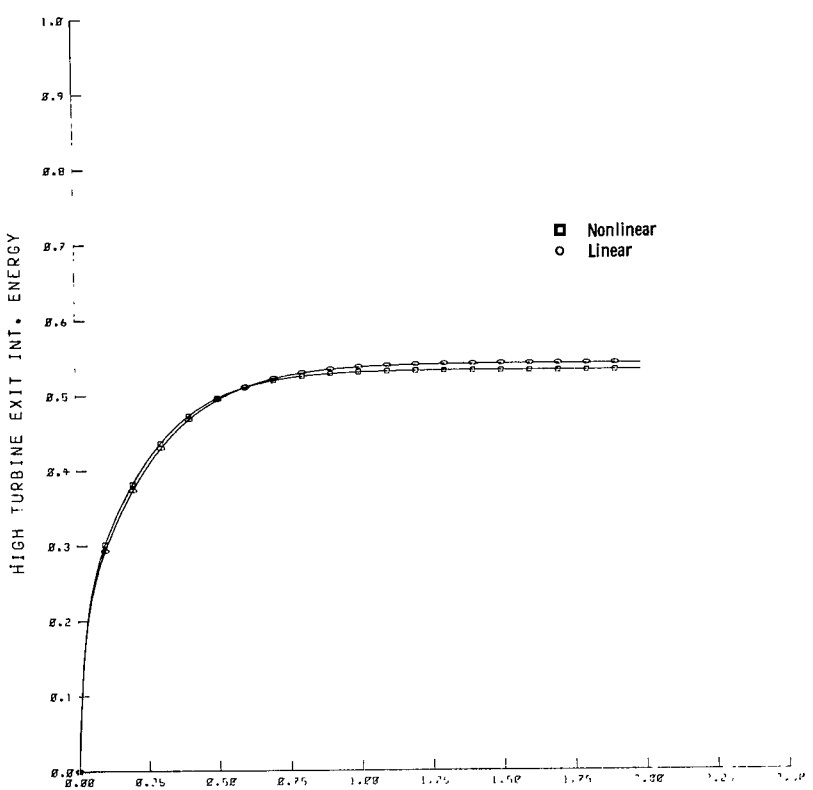
(b-1) With 3-percent step change in main fuel flow.



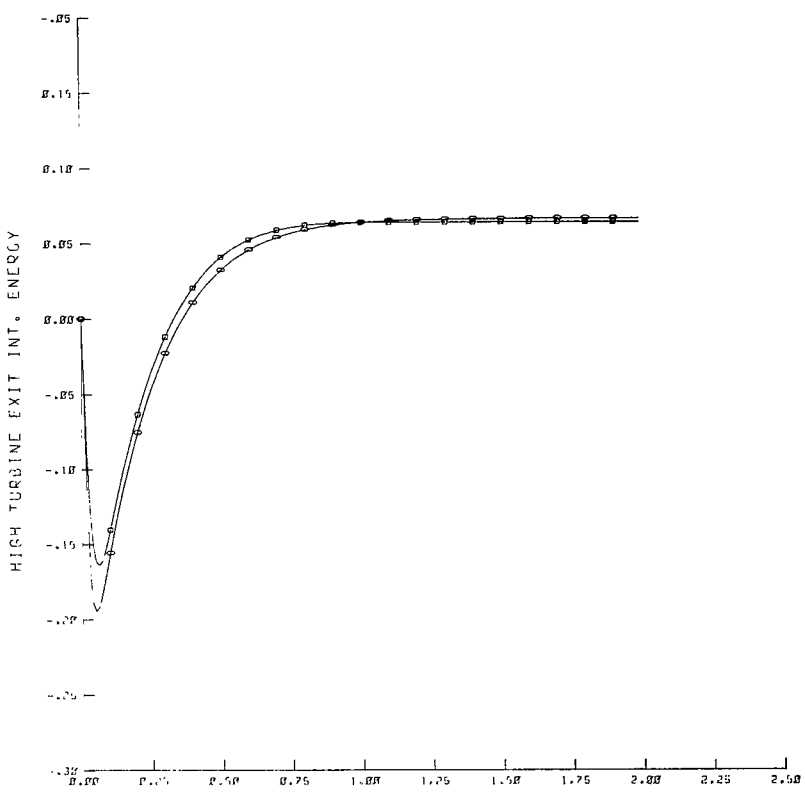
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

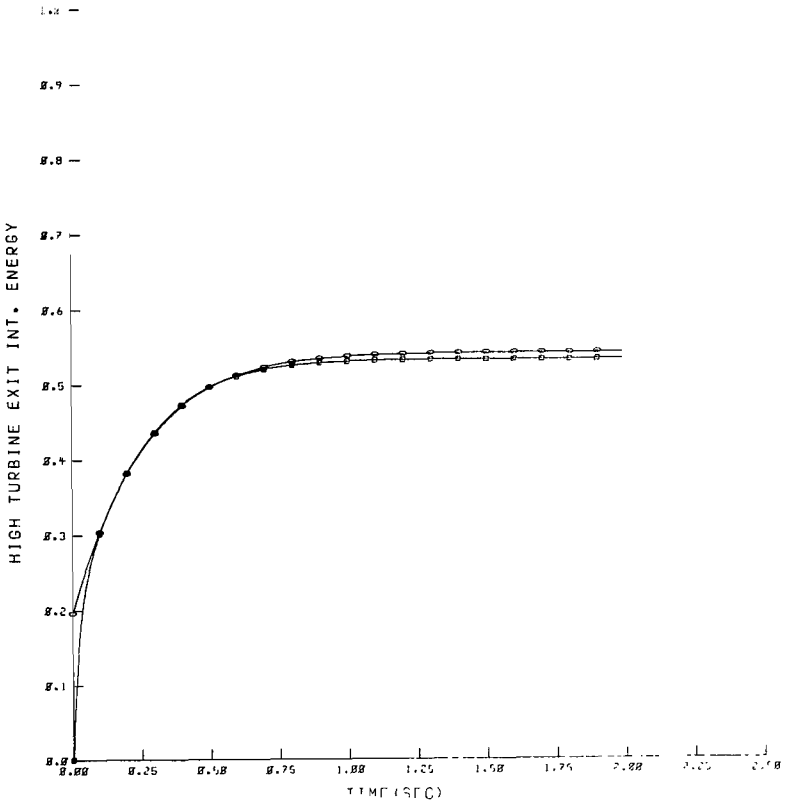
Figure 13. - Response of state 13 - high turbine-exit pressure.



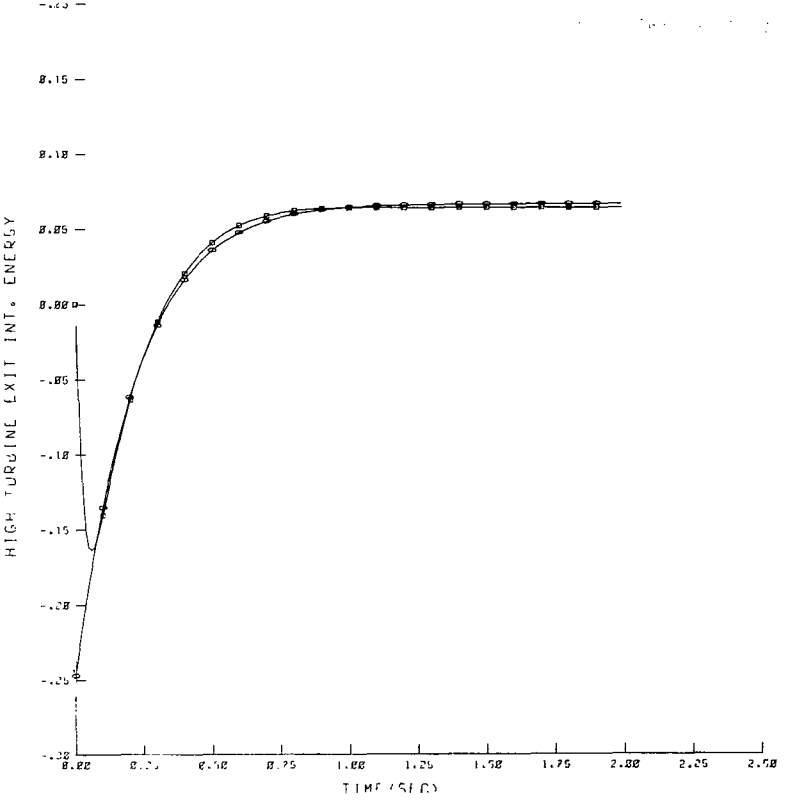
(a-1) With 3-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



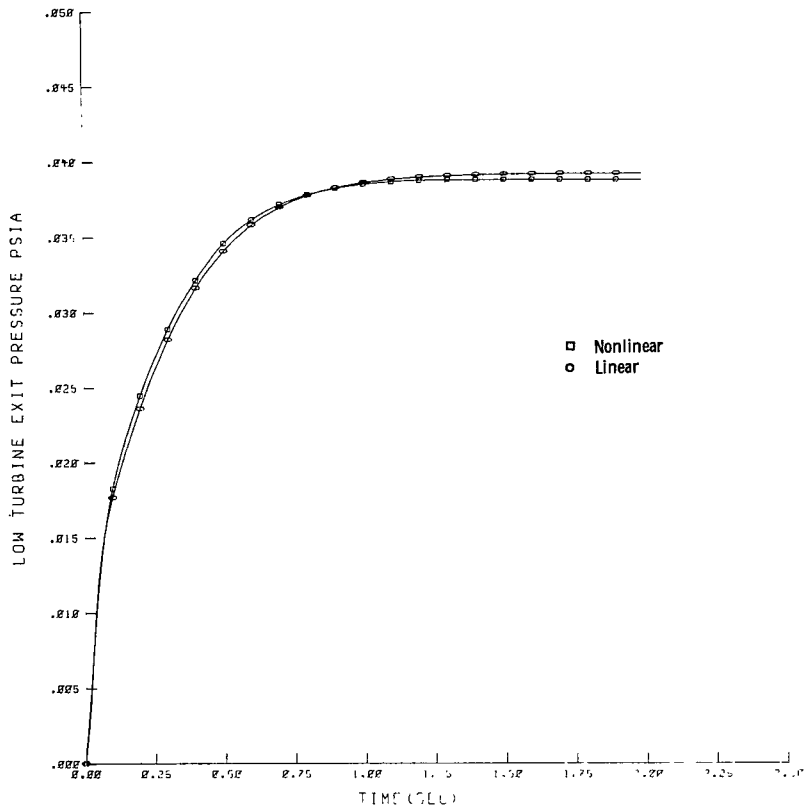
(b-1) With 3-percent step change in main fuel flow.



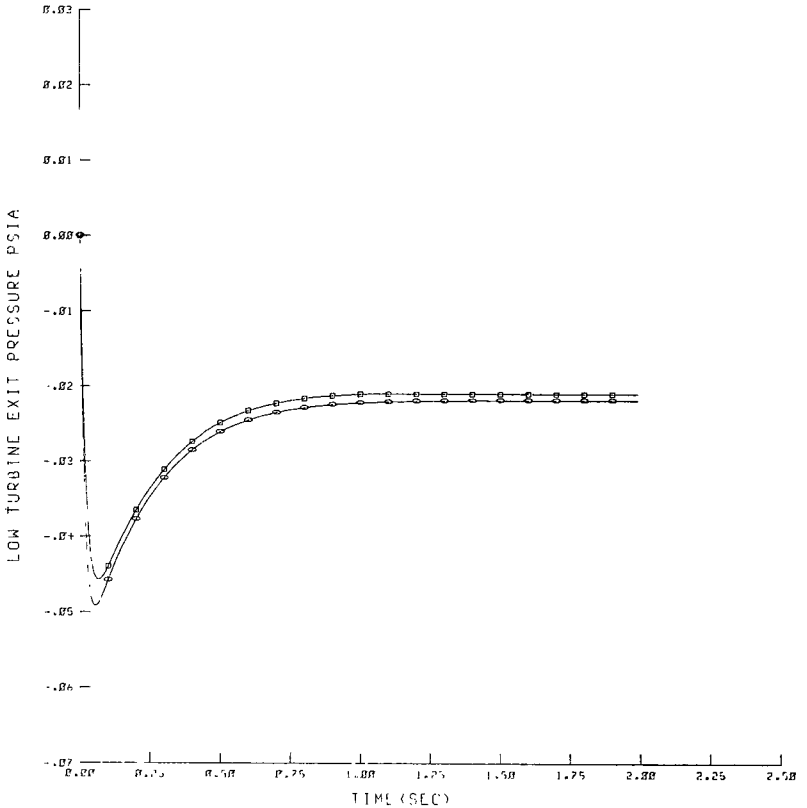
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 14. - Response of state 14 - high turbine-exit internal energy.

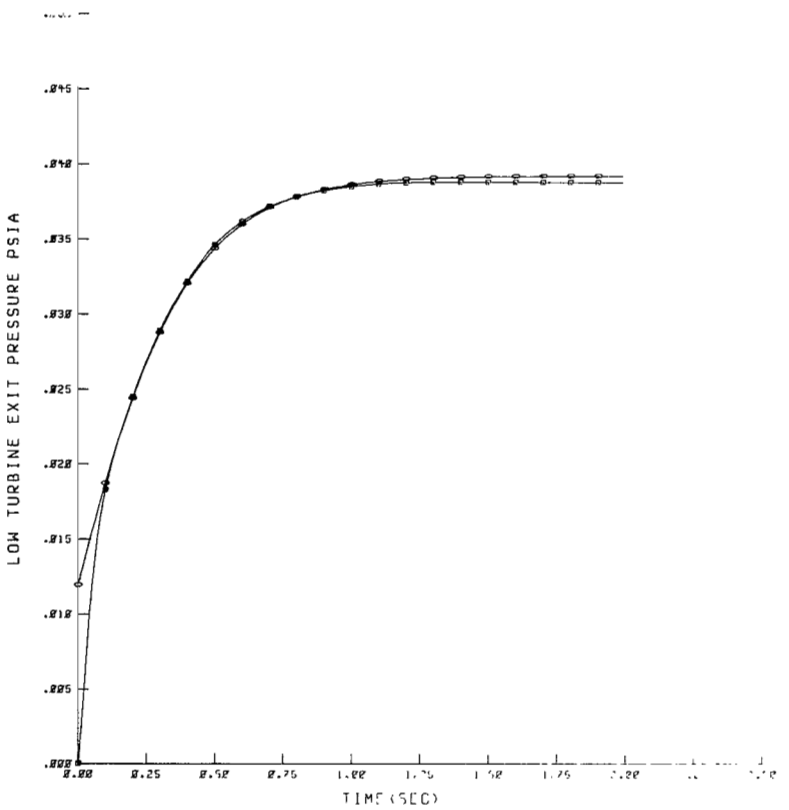


(a-1) With 3-percent step change in main fuel flow.

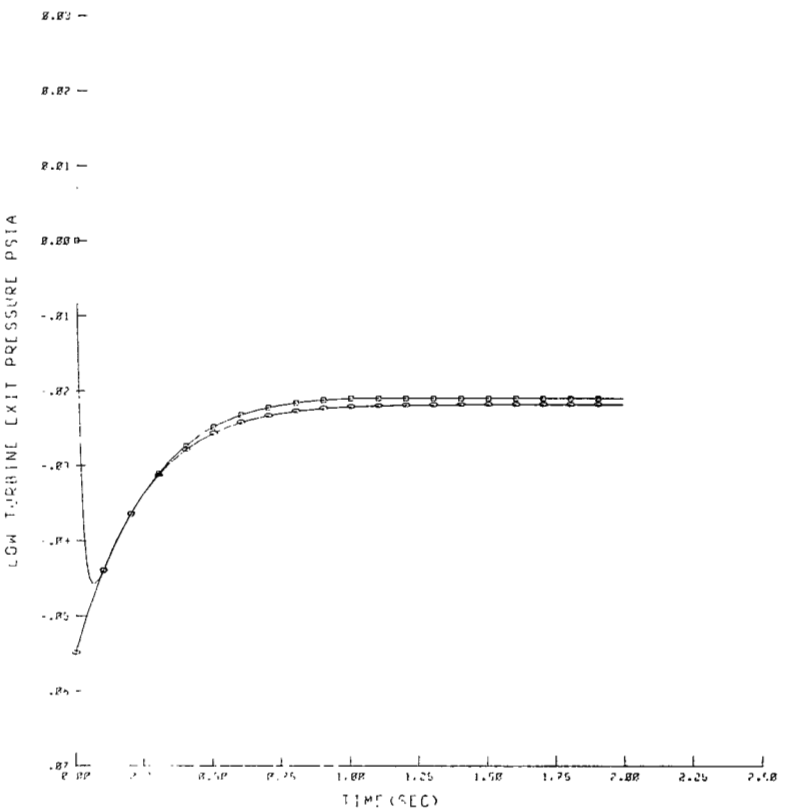


(a-2) With 3-percent step change in nozzle area.

(b) Comparison of linear responses and nonlinear runs.



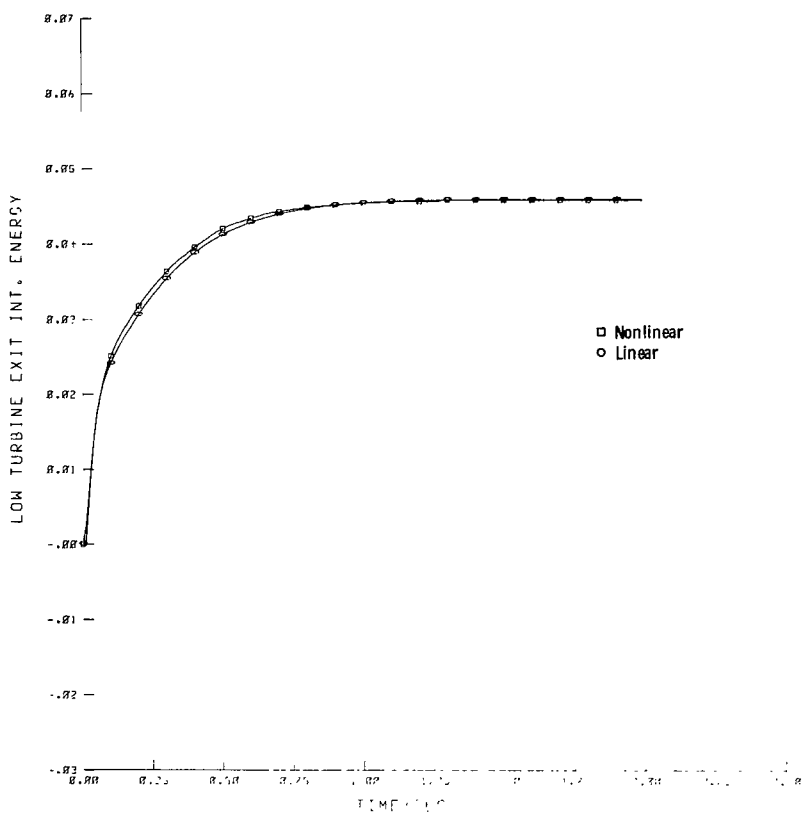
(b-1) With 3-percent step change in main fuel flow.



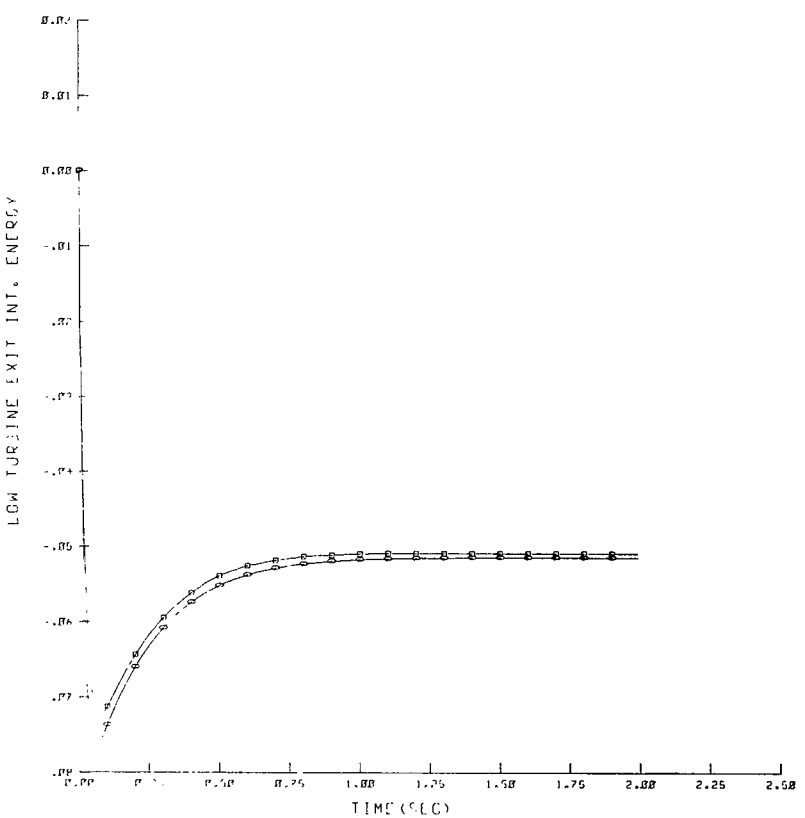
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

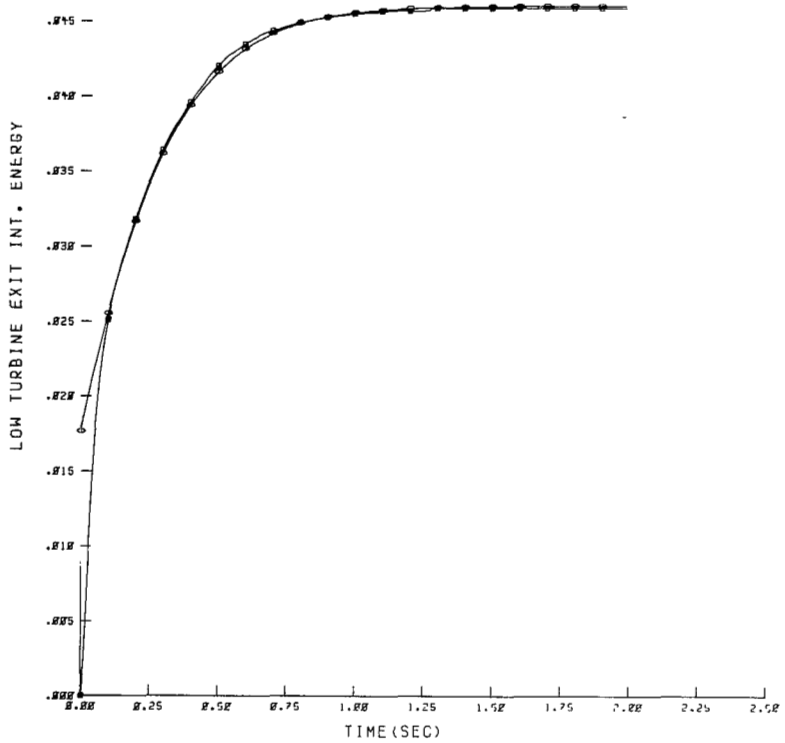
Figure 15. - Response of state 15 - low turbine-exit pressure.



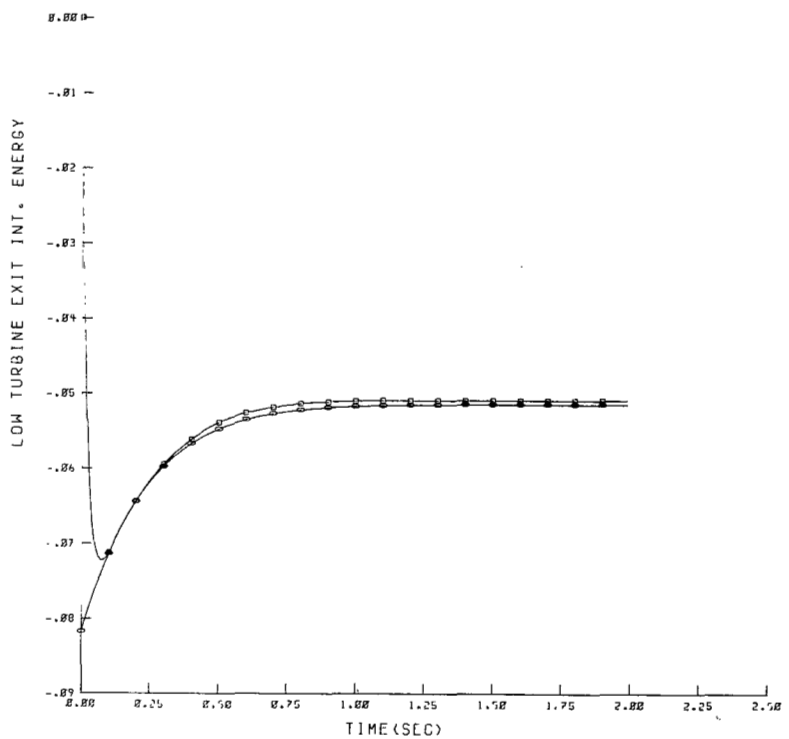
(a-1) With 3-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



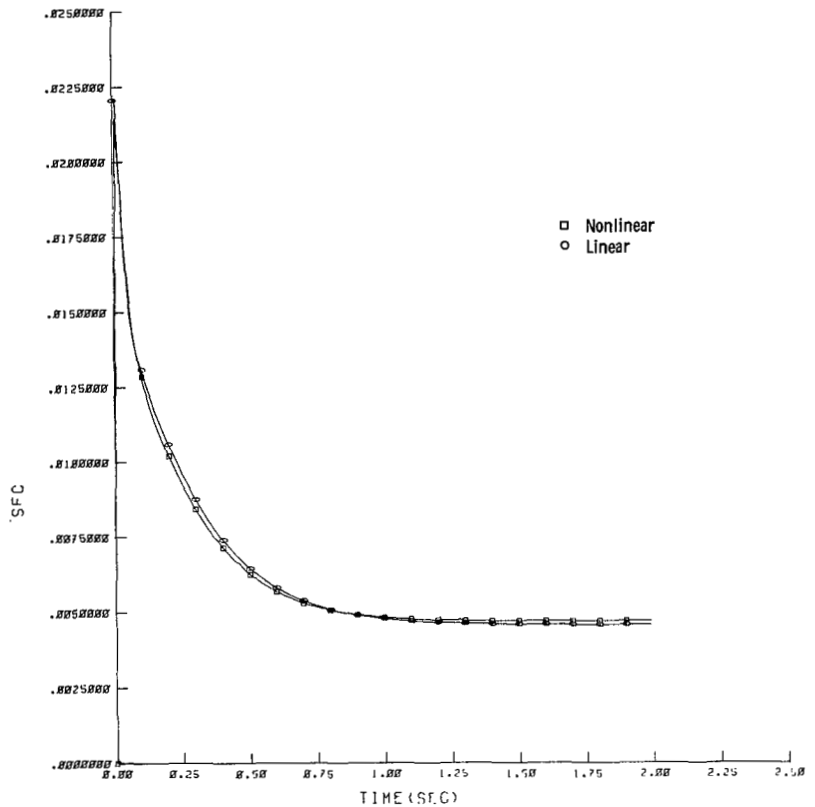
(b-1) With 3-percent step change in main fuel flow.



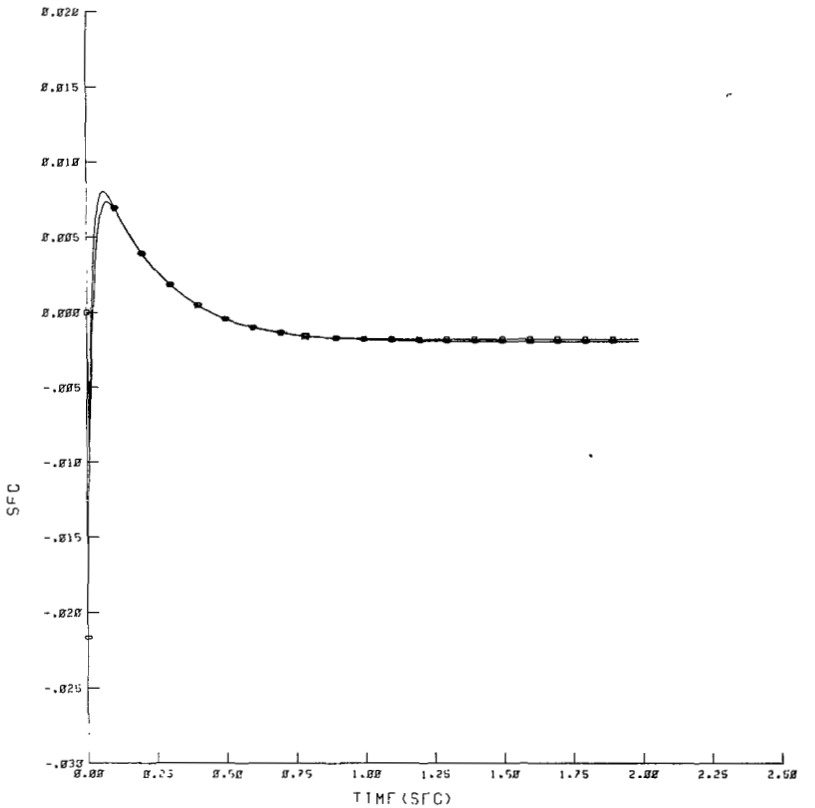
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

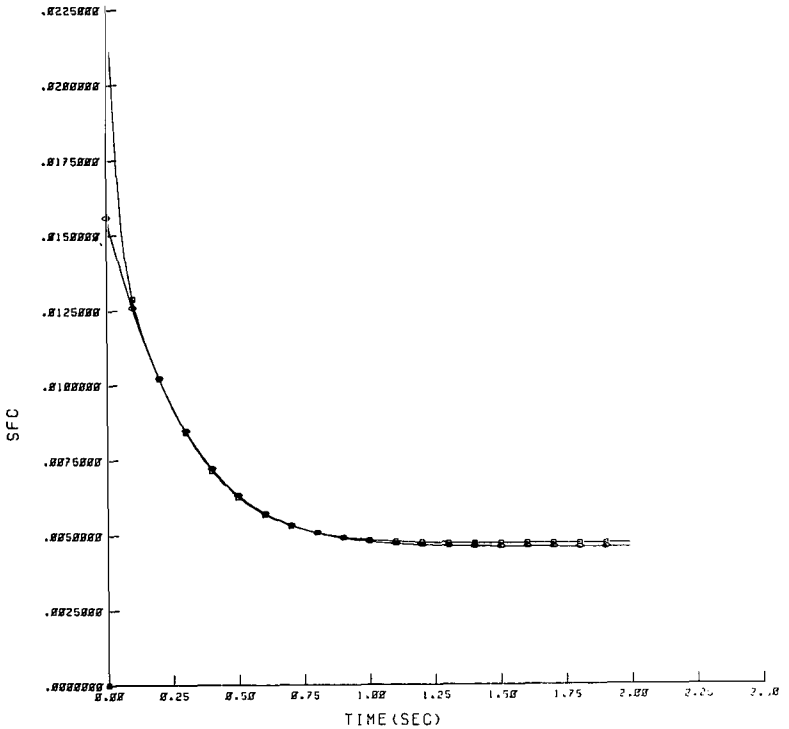
Figure 16. - Response of state 16 - low turbine-exit internal energy.



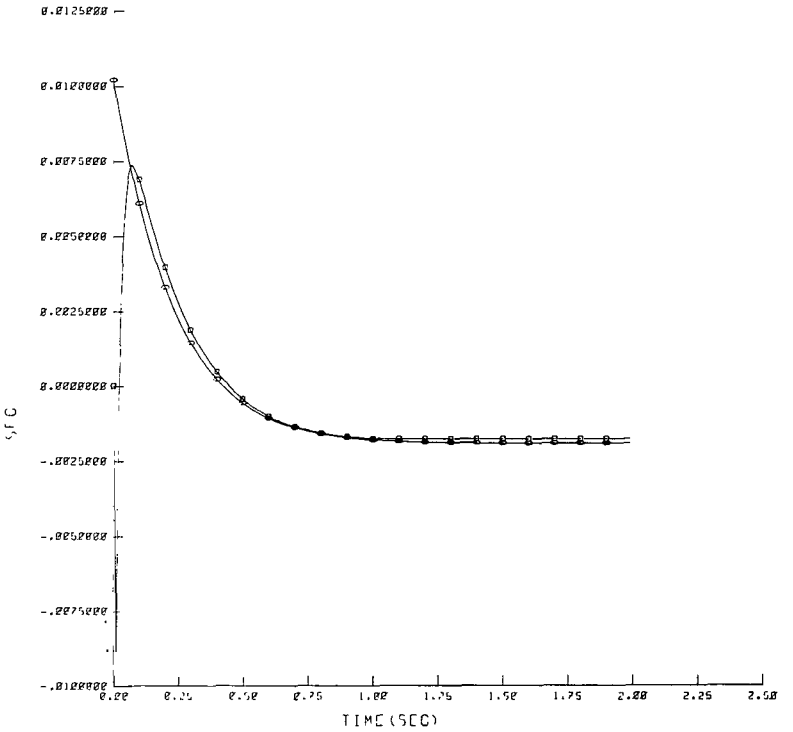
(a-1) With 3-percent step change in main fuel flow.



(a-2) With 3-percent step change in nozzle area.



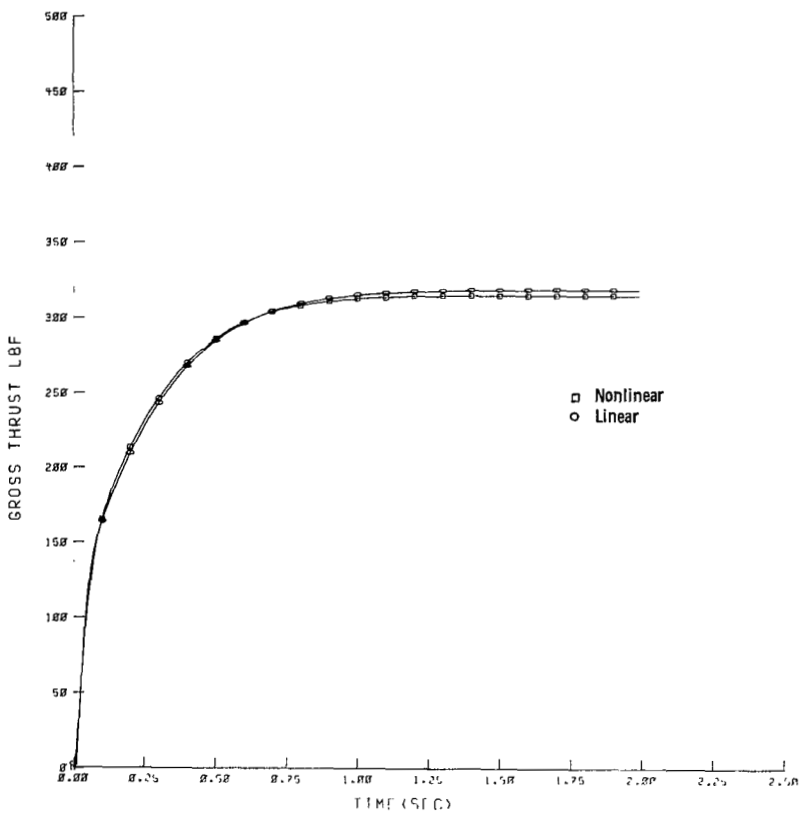
(b-1) With 3-percent step change in main fuel flow.



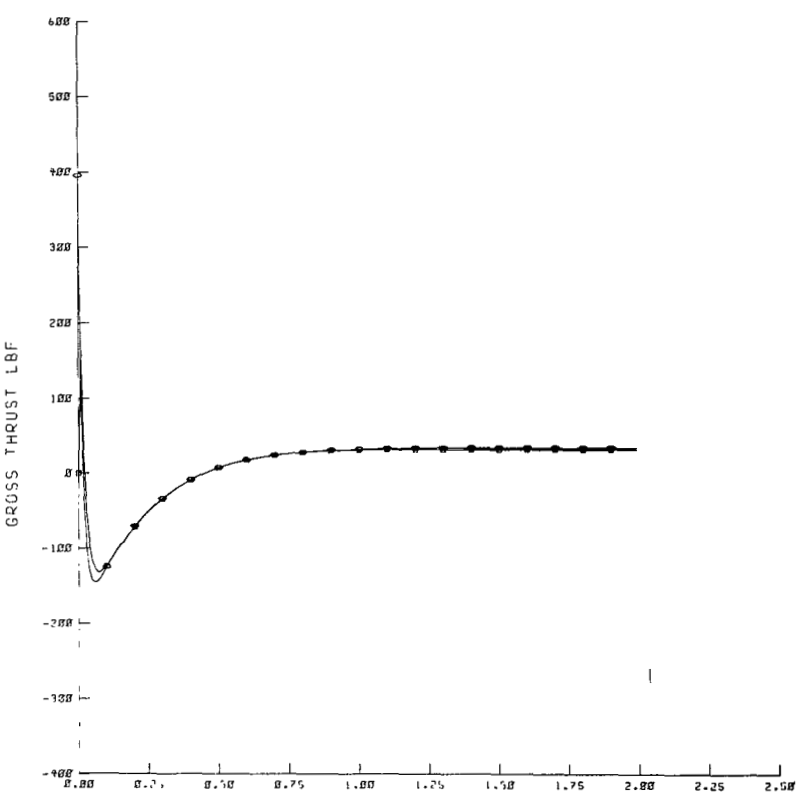
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 17. - Response of output 1 - specific fuel consumption.

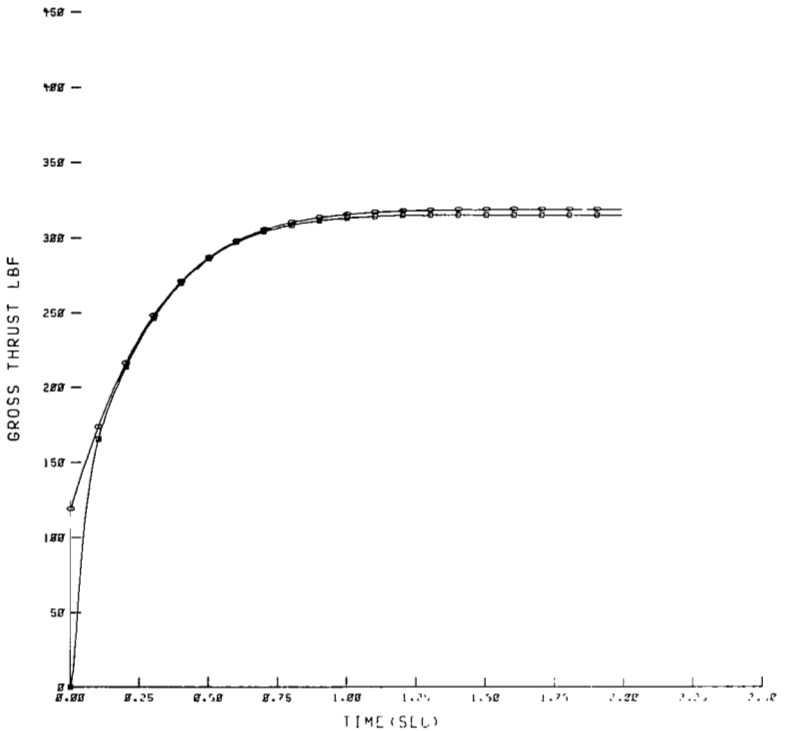


(a-1) With 3-percent step change in main fuel flow.

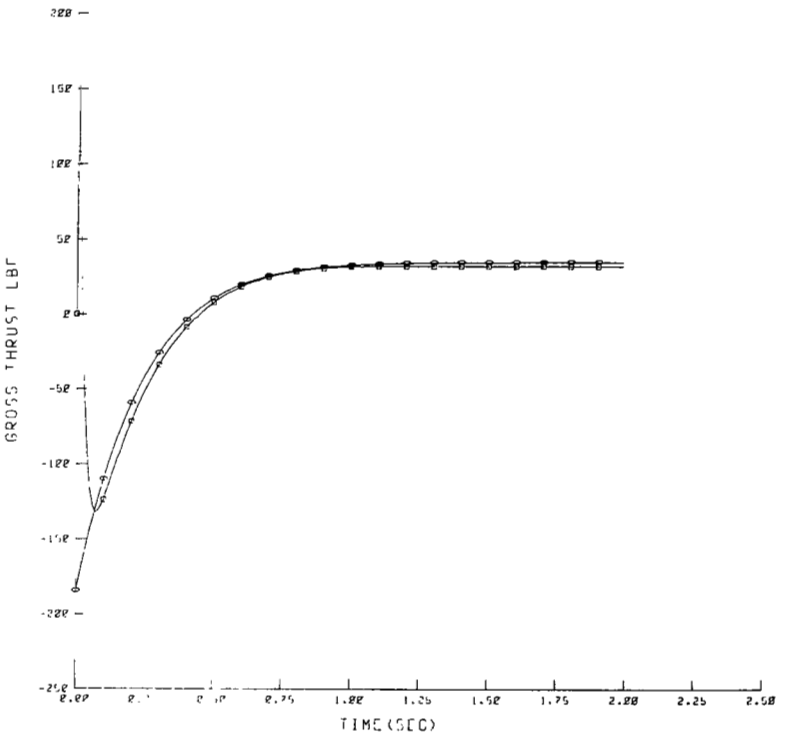


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



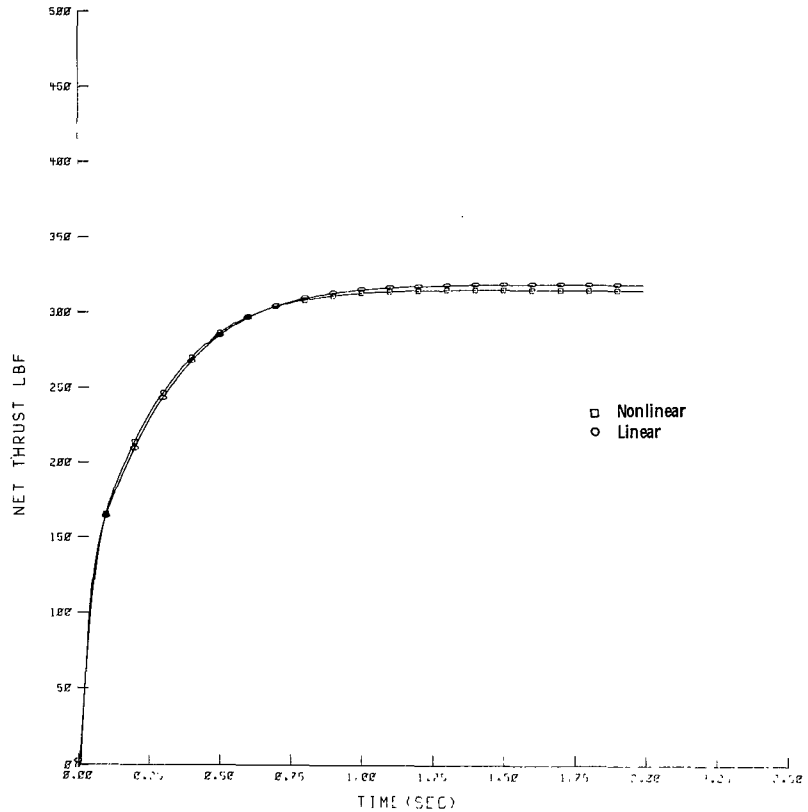
(b-1) With 3-percent step change in main fuel flow.



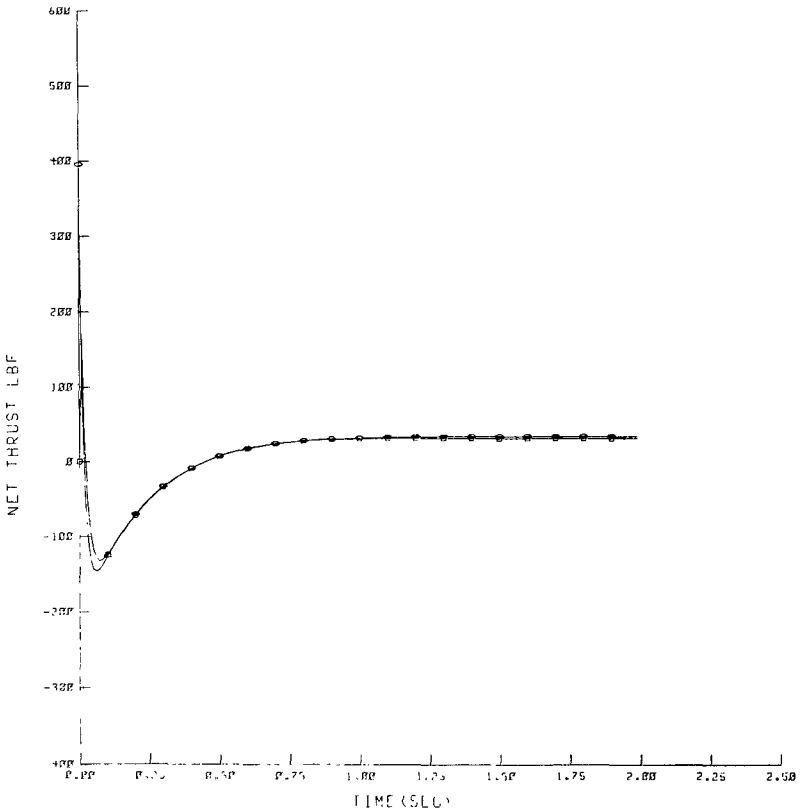
(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 18. - Response of output 2 - gross thrust.

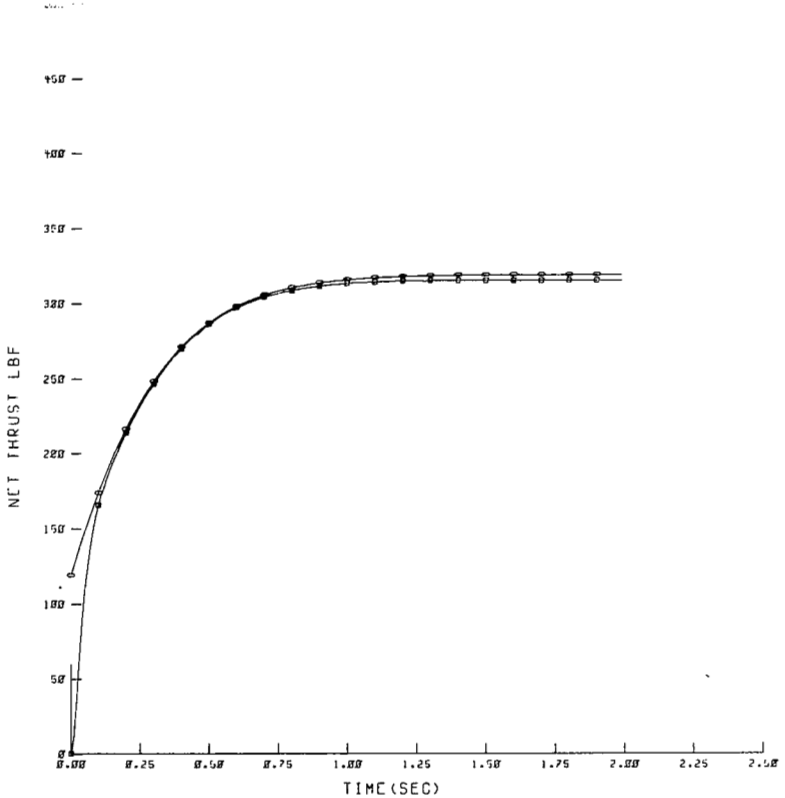


(a-1) With 3-percent step change in main fuel flow.

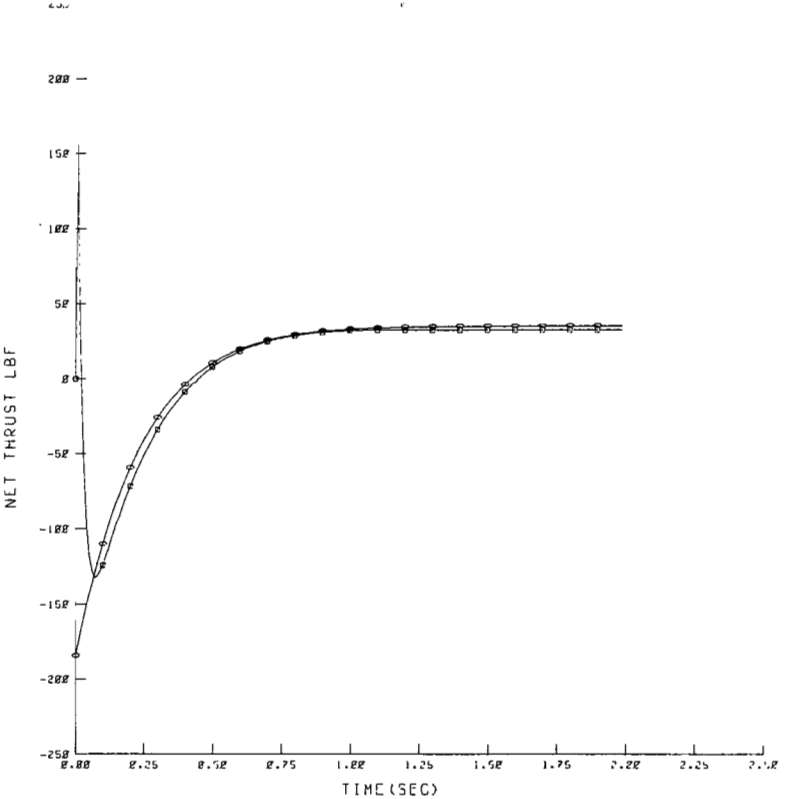


(a-2) With 3-percent step change in nozzle area.

(a) Comparison of full-order linear and nonlinear runs.



(b-1) With 3-percent step change in main fuel flow.



(b-2) With 3-percent step change in nozzle area.

(b) Comparison of reduced-order linear and nonlinear runs.

Figure 19. - Response of output 3 - net thrust.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TP-1295		

4. Title and Subtitle	5. Report Date
DYGABCD - A PROGRAM FOR CALCULATING LINEAR A, B, C, AND D MATRICES FROM A NONLINEAR DYNAMIC ENGINE SIMULATION	September 1978
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Lucille C. Geyser	E-9464
	10. Work Unit No. 505-05

9. Performing Organization Name and Address	11. Contract or Grant No.
National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135	

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Washington, D.C. 20546	Technical Paper
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A digital computer program, DYGABCD, has been developed that will generate linearized, dynamic models of simulated turbofan and turbojet engines. DYGABCD is based on an earlier computer program, DYNGEN, that is capable of calculating simulated nonlinear steady-state and transient performance of one- and two-spool turbojet engines or two- and three-spool turbofan engines. Most control design techniques require linear system descriptions. For multiple-input/multiple-output systems such as turbine engines, state space matrix descriptions of the system are often desirable. DYGABCD computes the state space matrices - commonly referred to as the A, B, C, and D matrices - required for a linear system description. The report discusses the analytical approach and provides a users manual, FORTRAN listings, and a sample case. NASA TN D-7901, describing DYNGEN, is a necessary adjunct to this report.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Dynamic engine simulation; Linear engine simulation; Nonlinear engine simulation; Reduced-order linear engine simulation	Unclassified - unlimited STAR Category 07

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	201	A09

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

NASA-Langley, 1978

14 1 1U,A, 082678 S00903DS
DEPT OF THE AIR FORCE
AF WEAPONS LABORATORY
ATTN: TECHNICAL LIBRARY (SUL)
KIRTLAND AFB NM 87117