NASA CONTRACTOR REPORT

NASA CR-710

NASA CR-710

N67 18397
(ACCESSION NUMBER)
234
(PAGES)
CR-710
(NASA CR OR TMX OR AD NUMBER)
(THRU)
1
(CODE)
03
(CATEGORY)
FACILITY FORM 602

ANALYTICAL PROCEDURE AND COMPUTER PROGRAM FOR DETERMINING THE OFF-DESIGN PERFORMANCE OF AXIAL FLOW TURBINES

by E. E. Flagg

Prepared by
GENERAL ELECTRIC
Flight Propulsion Division
Cincinnati, Ohio
for Lewis Research Center

GPO PRICE	$
CFSTI PRICE(S)	$ 3.00
Hard copy (HC)	
Microfiche (MF)	.65

ff 653 July 65

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • FEBRUARY 1967

134

NASA CR-710

ANALYTICAL PROCEDURE AND COMPUTER PROGRAM FOR DETERMINING THE OFF-DESIGN PERFORMANCE OF AXIAL FLOW TURBINES

By E. E. Flagg

Prepared under Contract No. NAS 3-7262 by
GENERAL ELECTRIC
Flight Propulsion Division
Cincinnati, Ohio

for Lewis Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151 – Price $3.75

FOREWORD

The research described herein, which was conducted by the General Electric Flight Propulsion Division, was performed under NASA Contract NAS 3-7262. The work was done under the technical management of Mr. Edward L. Warren, Airbreathing Engines Division, NASA-Lewis Research Center, with Mr. Arthur J. Glassman, Fluid System Components Division, NASA-Lewis Research Center, as research consultant. The report was originally issued as General Electric Document R66FPD258, March 1966.

PRECEDING PAGE BLANK NOT FILMED.

CONTENTS

APPENDIX

ANALYTICAL PROCEDURE AND COMPUTER PROGRAM FOR DETERMINING THE OFF-DESIGN PERFORMANCE OF AXIAL FLOW TURBINES

by E. E. Flagg
General Electric Company

1.0 SUMMARY

A method of analyzing the off-design performance of multi-stage axial-flow turbines is presented with appropriate analytical procedures and computer techniques. Use of the method is demonstrated by calculating a performance map on a turbine design of known performance provided by NASA Lewis Research Center.

The analysis is applicable to turbines having any number of stages up through eight (8). It allows for a change in mean-section radius between blade rows, and includes provisions for radial variation in loss and flow conditions. The radial variation in loss and flow conditions is calculated at up to six (6) radial positions at the radial center of fixed area sectors. Each sector is a quasi-one-dimensional element and the radial centers are joined utilizing simple radial equilibrium at the stator exit and the rotor exit. Semi-perfect gas properties are assumed with variable specific heat at constant pressure and variable specific heat ratio. Gas properties may be input, and in addition, provision is made to incorporate gas properties as a function of temperature. Geometry may be input as passage distributed area or vector flow angle. Two options to incorporate loss are provided in an expansion kinetic energy coefficient-inlet recovery coefficient method, and a total pressure loss coefficient method. The analysis is applicable from zero (0) speed to high speed, and the work done may be negative to the maximum work condition limited by discharge annulus area choking.

A complete listing of the computer program written in Fortran IV language is included with a discussion of all numerical techniques used in the analysis, a detailed logic flow diagram for the program, and a symbol list equivalencing the analysis equation variables to the Fortran variables. The computer program was demonstrated by calculating the performance map of a turbine of known performance.

Loss definition parameters and optimum incidence angle were selected to produce a calculated performance map with 94 percent peak rating efficiency at 120 percent design equivalent rotor speed. Blade row flow coefficients were selected to produce an equivalent flow function reduction with speed at high over-all rating total pressure ratios. This map is presented along with sample input and output.

2.0 INTRODUCTION

Investigations of advanced air breathing propulsion engines have indicated the need for wide thrust modulating capability with minimum SFC over a broad range of subsonic and supersonic flight conditions. This capability will allow mission cruise operation at reduced power settings and, by virtue of increased maximum thrus will reduce the number or size of engines required.

Current propulsion systems are limited by their inherent inability to adequately adjust the engine cycle to the desired thrust level while maintaining minimum SFC. Propulsion systems with variable geometry, however, can obtain the desired cycle flexibility through variation in the basic aerodynamics of the engine. Turbine stator area variation appears to offer very good potential. With the abilit to vary turbine stator area, it is possible to adjust the cycle pressure ratio, air flow and/or turbine inlet temperature to approach more optimum conditions. The achievement of optimum aerodynamic design of the turbine will depend upon a knowledge of the change in overall and internal performance as a function of the requirement variations. Therefore, a turbine performance prediction method is needed to assess the penalties associated with off-design performance and evolve design modifications to minimize these penalties.

The General Electric Company has initiated and developed a number of computer programs for predicting the off-design performance of multi-stage axial flow turbine The Large Steam Turbine Department, the Gas Turbine Department, and the Flight Propulsion Division have contributed to the present state-of-the-art in turbine off-design performance prediction. Programs have been written and successfully operate on the IBM 701, 650, 704, 7090, and 7094 computers as well as the GE 625 computer. Programs have been written in machine language, SOAP, in CAGE (Compiler and Assembl Program, General Electric), in Fortran II and Fortran IV language as modified by General Electric computer operational techniques. This program takes maximum advantage of the knowledge and experience previously acquired to develop an analytical method and tool using a sophisticated and complex computer program system to advance turbine technology for future airbreathing propulsion systems as well as other gas turbine systems.

3.0 DEVELOPMENT OF ANALYSIS PROCEDURE

3.1 Objective - The turbine is one of the principal components of the aircraft jet engine. In the case of most turbomachinery, the turbine is designed for a single operating condition (one discrete speed, pressure ratio, etc.) called the "design point". When operating at this design point condition, the turbine (with its fixed dimensions, blade angles, number of stages, etc.) will deliver the design point performance specifications (power, efficiency, etc.).

The aero-thermodynamicist predicts the turbine's design point characteristics by computing overall performance (weight flow, efficiency, work output, etc.) as well as more detailed interstage performance characteristics (velocity diagram quantities, state conditions, flow angles, etc.).

Within an aircraft jet engine, however, the turbine is required to perform at many conditions other than "design point". Its performance can be varied by adjusting the speed and/or pressure ratio, variable geometry stators, and other parameters. Under these different running conditions, the turbine is said to be operating "off-design". To completely predict the turbine's characteristics, the aero-thermodynamic designer must not only compute the single design-point performance but also the off-design performance over a wide range of operating conditions. The off-design performance is computed by varying one parameter (at a time) and repeating the same overall and interstage calculations used for the design. Each off-design point is plotted, thus generating a complete off-design predicted performance map. In addition to the overall performance characteristics in terms of equivalent work, equivalent weight flow parameter, equivalent speed, pressure ratio and efficiency, the aero-thermodynamicist is also interested in interstage blading performance in terms of incidence angles, Mach numbers, diffusion parameters, reaction and leaving swirl. Since each calculation is basically a repetition of the others, the turbine off-design computation is ideally suited for solution on a high-speed, digital computer.

3.2 Assumptions

3.2.1 Proportional Area Distribution. - For a pitchline type analysis, or a one (1) sector analysis, all of the flow must pass through a given area and there is no further assumption of flow per area necessary. When the quasi-one-dimensional procedure is extended to six (6) quasi-one-dimensional sectors, a decision must be made to perform the analysis and calculation on either a flow-stream-tube basis or an area basis.

In calculating the performance of turbines away from the design point, the flow-stream-tubes will shift radially and adjust because of momentum, energy, continuity and equilibrium conditions in each of the flow-stream-tubes. In addition, the radial distribution of static pressure between each of the flow-stream-tubes will adjust to maintain equilibrium. Thus, the flow per unit area is not necessarily a constant with radius as shown in Figure 1.

If the decision is to use flow-stream-tubes as the basic calculation technique, an iterative procedure must be employed to adjust the area required to pass the flow in each stream-tube. General Electric experience with stream-tube procedure was that many difficult problems must be circumvented in the vicinity of Mach one or choked flow condition in addition to the normal problems associated with iteration on continuity, momentum and energy. One example is if a high initial value of static pressure (or low axial velocity) is used to start the iteration, a large area is required to pass the flow of the outermost stream-tubes, and the flow in the inner stream-tube requires an area at smaller diameters than the root flowpath. For low radius-ratio turbines, negative diameters may be indicated by the iteration procedure which is an impossibility. Another example is if a low initial value of static pressure (or high axial velocity) is used to start the iteration, for a given total energy or total temperature level, a negative static temperature may be indicated by the iteration procedure. An alternate technique is to base the basic calculation on a proportional area basis.

If proportional area distribution is used throughout the turbine, it can be seen in Figure 1 that a different quantity of flow may enter the upstream cross-section area as will leave the downstream cross-section because of the flow that enters and leaves the conic surface area between the calculation stations. Since the equations of motion as applied to turbomachinery are, strictly speaking, valid along axi-symmetric stream-tubes, the assumption of fixed proportional area sectors introduces a small error in the momentum equation and the energy equation. To test the assumption of fixed area sectors, three test cases were calculated as five (5) one-dimensional sectors as shown in Figure 2.

One test case had equal sector heights at the stator exit station and the rotor exit station. One case had a large stator tip sector (small stator root sector), and the rotor exit had equal heights. Thus streamlines enter the rotor at a lower radius and leave at a higher radius, and the tip sector has high weight flow and the root sector has low weight flow. The alternate test case had a large stator root sector height (small stator tip sector height), streamlines enter the rotor at a high radius, leave at a lower radius, the tip sector has lower weight flow, and the root sector has higher weight flow. The stator angles were adjusted as a function of radius to simulate a free vortex design distribution. An important result of these three test cases is shown in Figure 3. As the stator sectors were adjusted radially 2% in station height ($\pm$ 10% in tip and root sector height), no change was observed in the energy function $\Delta h/T$ as stage pressure ratio was varied.

In order to evaluate the procedure for a non-free vortex design distribution, two additional test cases were completed with the stator pitchline flow angle held constant as the sectors were adjusted radially. The result shown in Figure 4 was a change in work output. A 2% shift in station height ($\pm$10% in tip and root sector height) produced a 2% change in energy function $\Delta h/T$ as stage pressure ratio was varied. From the manner the calculation was performed, the test cases were for a very non-free-vortex type distribution.

3.2.2 Radial Height Center. - Once the framework is set up on area sectors, it must be determined whether to calculate at the area centers of the proportional area sectors, at the root mean square diameter of the area sectors, or at the radial height center of the area sectors.

The use of equal area centers has long been a practice of installing instrumentation. It is based on an approximation to equal flow tubes. Therefore, an approximation to a mass flow weighted condition is an area weighted condition which is a simple average of data taken at equal area centers. In many flow situations the radial distribution of flow per unit area is a significant variable, particularly at the turbine stator exit and generally true in compressors. Since the mass flow is calculated in each of the six (6) area sectors, the difference between area centers or centers of equal station height should be small and only a matter of convenience in setting up the calculation procedure. The diameter of the calculation sector areas used as the radial height center are:

for i = 1

$$Dp_{1,k} = Dr_k + .5 * PCNH_1 * (Dt_k - Dr_k)$$

for i = 2, 3, 4, 5, 6

$$Dp_{i,k} = Dp_{i-1,k} + (.5 * PCNH_{i-1} + .5 * PCNH_i) * (Dt_k - Dr_k)$$

3.2.3 Radial Variation Method. - Several methods are available to obtain a quasi-one-dimensional solution to the turbine interstage flow conditions. The primary consideration of this program was to get an analytical expression to be iterated and integrated and avoid step-by-step calculations in radius with numerical integration methods. The most simple procedure is to assume a simple one-dimensional compressible flow in each of the sectors. That is to say that all total conditions as well as static conditions and velocities are constant throughout the sector. Then the sectors can be joined radially utilizing simple radial equilibrium to determine the radial pressure distribution at the sector centers. Many test data reduction systems have used this procedure. If ultimate objectives were to reduce test data with one-dimensional flow assumed in each sector, in order to determine the test loss definition parameters, then the off-design prediction procedure should be a compatible method. The error involved will tend to be smaller as the number of sectors is increased from one (1) to six (6). In addition, many times the actual test results are significantly different from the predicted performance due to deviation angle differences or flow coefficient differences due to local separation or tip-clearance boundary-layer interaction or secondary-flow boundary-layer centrifugation.

An extension to one-dimensional area segments are free-vortex area segments. It is then assumed that total pressure, total temperature and axial velocity are constant throughout each area sector, and the static conditions and tangential component of velocity vary as a free vortex. The six (6) radial sectors are then joined utilizing simple radial equilibrium. For a free-vortex turbine design, (of which there are many), the off-design performance result at the design operation condition is identical to the design case.

A third method which was evaluated was previously applied to a compressor analysis. A radial derivative form of the dynamics relations was obtained as a function of known geometry and the axial component of Mach number. This non-linear function was then calculated at three (3) radial positions and curve fit as a polynomial in radius. The radial derivative form of the dynamics relations was then known as a polynomial in radius and integrated in closed form to yield the radial distribution in static pressure. In addition, the energy equation and geometry provide a relationship between static pressure, total pressure, Mach number and axial component of Mach number such that at the selected axial component of Mach number, and geometry, the energy equation would yield a static pressure which is different from the static pressure that satisfies the momentum equation. The Mach number was then adjusted to find a static pressure that was common to both. The result was an analytic continuously varying total pressure profile, total temperature profile and velocity profile.

It was assumed that total pressure, total temperature and axial velocity are constant throughout each area sector; and for energy and momentum considerations, the static conditions and tangential velocity vary as a free-vortex. The six (6) radial area sectors are then joined utilizing simple radial equilibrium.

$$\frac{dPs}{dr} = \frac{\rho_s}{g}\frac{Vu^2}{r}$$

3.2.4 Continuity Integration. - In conjunction with a radial distribution of total temperature and total pressure to satisfy the energy and momentum conditions, the integrated total flow in each of the segments must add up to the total flow. Again the primary consideration is to get an analytic expression, avoid step-by-step numerical integration methods, and be compatible with the energy and momentum equations. The most simple procedure is to assume a simple one-dimensional compressible flow in each of the sectors. Then the specific mass flow calculated at the center of each of the sectors is considered to be constant throughout each of the sectors.

An extension to the simple one-dimensional area segments is free-vortex area segments. Although the axial component of velocity is constant, the density varies with radius, and the continuity integration with radius is longer than the simple one-dimensional flow in each sector.

A third system evaluated was used in conjunction with the curve fit of the dynamics equations with radius. The continuity equation was derived as a function of axial Mach number, Mach number and radius. The integrand was evaluated at three radial locations and curve fit as a polynomial in radius. The flow in any sector was then analytically determined as a function of an estimated axial Mach number, and the total flow of all the sectors determined. It was the purpose of the continuity convergence procedure to estimate the axial Mach number level, determine the radial distribution with the dynamics curve fit, satisfy the energy equation, integrate the mass flow with a continuity curve fit, then adjust the initial axial Mach number and repeat until the mass flow error is less than some tolerance.

It was assumed that total pressure, total temperature and axial velocity are constant throughout each area sector; and for continuity considerations, the specific mass flow was considered to be constant throughout each of the sectors, even though the dynamics was assumed to be free vortex in each radial segment. This small error introduced in continuity is often ignored even when applied to the entire flowpath. It is normally smaller than the uncertainty of the boundary layer displacement thickness, and the streamtube slope and curvature effects on flow distributions. When applied to the sectors of the flowpath, the error is reduced.

3.2.5 Loss Definition. - In the development of an analysis procedure for the off-design performance of an axial flow turbine, the loss assumption method is usually associated with the aero-thermodynamicists' experience and the test evidence to support the loss method. The objective of the various loss definition methods is to provide for a loss in total pressure as normalized by some suitable normalizing procedure which correlates well with the experimental evidence.

The use of blade row energy efficiency coefficient for the expansion process has long been a standard practice of turbine designers.

By definition,

Stator -

$$\eta_s = \frac{\text{Actual exit kinetic energy}}{\text{Theoretical exit kinetic energy}}$$

$$= \frac{Ho - h_1}{Ho - h_{1_i}} = \frac{(V_1)^2/2Ho}{1 - h_{1_i}/Ho}$$

$$= \frac{(V_1)^2/2Ho}{1 - (Ps_1/Pt_o)^{\frac{\gamma-1}{\gamma}}}$$

(See Figure 1 for station designation)

In a similar manner, the efficiency of the blade profiles for the expansion process in a rotating row is defined as:

Rotor -

$$\eta_R = \frac{\text{Relative exit kinetic energy}}{\text{Theoretical exit relative kinetic energy}}$$

$$= \frac{Hr_{1A} - h_2}{Hr_{1A} - h_{2i}} = \frac{(R_2)^2/2Hr_{1A}}{1 - h_{2i}/Hr_{1A}}$$

$$= \frac{(R_2)^2/2Hr_{1A}}{1 - (Ps_2/Ptr_{1a})^{\frac{\gamma-1}{\gamma}}}$$

(NOTE: The above values are a function of radial sector and held constant during off-design càlculation.)

The incidence loss definition can be handled in several different manners. One such method employs an inlet total pressure recovery factor specified as an analytic function of incidence angle, defined by:

Stator -

$$\eta_{SR} = \left[\left(\frac{Pto}{Pso}\right)^{\frac{\gamma-1}{\gamma}} - 1 \Big/ \left(\frac{\gamma-1}{2}\right)M_o^{\ 2}\right]$$

Rotor -

$$\eta_{RR} = \left[\left(\frac{Ptr_{1A}}{Ps_{1A}}\right)^{\frac{\gamma-1}{\gamma}} - 1 \Big/ \left(\frac{\gamma-1}{2}\right) M_{1A}^{\ 2}\right]$$

(NOTE: The above values are a function of radial sector.)

In some instances it has been assumed that for off-incidence angle calculations, the flow velocity is composed of two components of velocity; one component in the direction of the inlet angle at the design condition and another component which is normal to the inlet angle at the design condition. It is then further assumed that the velocity component oriented at the design condition passes through the turbine at the design loss level, and some fraction of the component of velocity normal to the design condition is lost. This type of design assumption leads to a definition of inlet recovery factor which is proportional to the cosine of the incidence angle raised to an exponent.

Stator -

$$\eta_{SR} = \eta_{SR}opt * \left[\cos(I_s)\right]^{exp}$$

Rotor -

$$\eta_{RR} = \eta_{RR}opt * \left[\cos(I_r)\right]^{exp}$$

From theoretical assumptions, if all the component of velocity normal to the design condition is lost, the exponent should be 2.

In conjunction with the expansion coefficients and the inlet recovery factor, a stage test factor has sometimes been employed in turbine design practice.

Stage Test Factor -

$$TF = \frac{\text{Output energy}}{\text{Vector diagram energy}}$$

The stage test factor has been used to represent the non-uniform work extraction due to blade end effects, or the average stage total enthalpy drop as compared with the pitchline vector diagram total enthalpy drop.

Extensive volumes of turbine aerodynamic cascade test results from transonic cascade wind tunnels have not appeared in the open literature. Significant differences still exist between British wind tunnel data, Australian wind tunnel data and the United States NASA wind tunnel results. The largest set of information in the United States has appeared in NACA TN 3802 "Investigation of a Related Series of Turbine Blade Profiles in Cascade" by Dunavant and Erwin. This data is presented in terms of a wake drag coefficient which is the total pressure loss as normalized by the upstream dynamic pressure. At high reaction conditions, a large difference between the low entering velocity on which the coefficients are based and the high leaving velocity causes high drag and lift coefficients; but the ratio of lift divided by drag is relatively unaffected.

Perhaps a more convenient loss coefficient method is the loss in total pressure as normalized by the theoretical downstream dynamic pressure:

$$\bar{\omega}_i = \frac{\text{Pt loss}}{\text{Theoretical exit dynamic pressure}}$$

Stator -

$$\bar{\omega}_{Si} \doteq \frac{Pto - Pt_1}{Pto - Ps_1}$$

Rotor -

$$\bar{\omega}_{Ri} = \frac{Ptr_{1A} - Ptr_2}{Ptr_{1A} - Ps_2}$$

This type of loss coefficient incorporates the expansion loss as well as the off-incidence angle loss and as such is described as an analytic function of off-incidence position.

Stator - for positive incidence

$$\bar{\omega}_{Si} = \bar{\omega}_{Sopt} \left[1 + A_1 \left(\frac{Is}{Is_{range}}\right)^2 + A_2 \left(\frac{Is}{Is_{range}}\right)^3 + A_3 \left(\frac{Is}{Is_{range}}\right)^4 \right]$$

Stator - for negative incidence

$$\bar{\omega}_{Si} = \bar{\omega}_{Sopt} \left[1 + A_4 \left(\frac{Is}{Is_{range}}\right)^2 + A_5 \left(\frac{Is}{Is_{range}}\right)^3 + A_6 \left(\frac{Is}{Is_{range}}\right)^4 \right]$$

Rotor - for positive incidence

$$\bar{\omega}_{Ri} = \bar{\omega}_{Ropt} \left[1 + B_1 \left(\frac{Ir}{Ir_{range}}\right)^2 + B_2 \left(\frac{Ir}{Ir_{range}}\right)^3 + B_3 \left(\frac{Ir}{Ir_{range}}\right)^4 \right]$$

Rotor - for negative incidence

$$\bar{\omega}_{Ri} = \bar{\omega}_{Ropt} \left[1 + B_4 \left(\frac{Ir}{Ir_{range}}\right)^2 + B_5 \left(\frac{Ir}{Ir_{range}}\right)^3 + B_6 \left(\frac{Ir}{Ir_{range}}\right)^4 \right]$$

3.2.6 Choked Flow - In order to evaluate the choked flow criteria, and perform the radial equilibrium calculation and continuity integration, six test cases were thoroughly analyzed. A list of the example test cases follows:

Test Case 1. - Small stator area, equal sector height.

Test Case 2. - Small stator area, free vortex stator sector angle, large stator tip sector height (small stator root sector height).

Test Case 3. - Small stator area, free vortex stator sector angle, large stator root sector height (small stator tip sector height).

Test Case 4. - Small stator area, constant stator sector angle, large stator tip sector height (small stator root sector height).

Test Case 5. - Small stator area, constant stator sector angle, large stator root sector height (small stator tip sector height).

Test Case 6. - Large stator area, equal sector height.

Shown in Figure 5 are the results of the flow calculation in each sector as a function of the sector pressure ratio for Test Case 1. The solid lines indicate the flow through the respective sectors with two branches in the supersonic region: one branch for choked flow at the critical value, and the other branch for a supersonic flow decrease. The dashed lines indicate the radial distribution of sector pitchline pressure ratio. The oontinuity integration was carried out along the dashed lines.

Shown in Figure 6 are the results of the continuity integration across the five sectors using two different assumptions as compared with the simple one-dimensional flow procedure. For the simple one-dimensional flow calculation, the maximum flow through the selected test case was Wg = 190.11 #/sec at a station pitchline pressure ratio of $Pt_0/Ps_1 = 1.89$. This condition was a reference for comparison of the five-sector calculation methods.

For the first calculation method, it was assumed that the respective sectors reached a critical flow value which was then held constant as the sector pressure ratio increased, and the flow angle was over-expanded to increase the area. When the station pitchline pressure ratio was 1.89, the pitch sector was slightly critical at $(Pt_0/Ps_1)_{3,1} \sim 1.89$, the root sector was over-critical at $(Pt_0/Ps_1)_{1,1} = 2.26$, and the tip sector had not reached critical at a value of $(Pt_0/Ps_1)_{5,1} = 1.60$. The integrated flow at this condition was 190.1 #/sec. Thus, when the station pitchline pressure ratio was at a critical value, the integrated flow using the constant flow with an over-expanded area method was 0.00% higher than the simple one-dimensional flow calculation. The maximum flow was not reached until the tip sector was critical. For this point the station pitchline pressure ratio and pitchline sector pressure ratio was critical at $(Pt_0/Ps_1)_{3,1}$= 2.1, and the the root sector was well over critical at $(Pt_0/Ps_1)_{1,1}$ = 2.7. The integrated flow at this maximum flow condition was 191.02 #/sec which was 0.48% higher than the simple one-dimensional flow calculation.

In an alternate calculation method, it was assumed that the respective sector flows decreased as the sector pressure ratio exceeded the critical value. When the station pitchline pressure ratio was 1.6, the root sector reached the critical value. As the station pitchline pressure ratio is increased to 1.89, the root and root-pitch sector flows are decreasing whereas the pitch, tip-pitch, and tip sector flows are increasing. The integrated flow at this condition was 189.15 #/sec which was .51% lower than the simple one-dimensional flow calculation. The maximum flow was calculated at a condition before the tip sector was critical where the decrease in flows in the root, root-pitch, and pitch sectors offset the increase in flows of the tip-pitch, and tip sectors. This maximum value was 189.3 #/sec which was .43% lower than the simple one-dimensional calculation.

In comparing the two calculation methods at the stator exit station, the maximum flows as calculated from either method are ± 0.5% from a simple one-dimensional flow method. Thus, proof of validity of a calculation procedure based on test evidence of measured flow will be difficult. The unknown flow coefficient could be in error by 1/2% to offset the difference in calculation methods.

Shown in Figure 7 and Figure 8 are the rotor exit flow calculations for Test Case 1. Shown in Figure 7 are the results of the rotor flow calculation in each sector as a function of the sector relative inlet total pressure to static pressure ratio. The solid lines indicate the flow through the respective sectors. Only one branch is shown because for the example selected the stator chokes first and limits the flow. The dashed lines indicate the radial distribution of sector pitchline pressure ratio. Figure 8 indicates the total stage flow as a function of stage inlet total pressure to static pressure ratio. To test the effect of the assumption of fixed area sectors on the continuity integration, the three test cases as previously indicated in Figure 2 were evaluated from a flow continuity viewpoint. Results similar to Figures 5 through 8 were obtained, however, no appreciable effect was observed on the stator station weight flow or the stage weight flow.

The selected method for station choke determination was that sectors with pressure ratio greater than the pitchline sector pressure ratio have supersonic flow decrease up to the maximum pitchline sector flow function. Thereafter the pitch sector and sectors with pressure ratio greater than the pitch sector have a constant flow function with effective area increase. Subsonic sectors have a weight flow increase after the pitch sector pressure ratio for maximum pitch sector flow function up to local sector maximum flow function. Because of the sector efficiency term η s or η r, the maximum flow function occurs at a pressure ratio slightly less than the sonic pressure ratio.

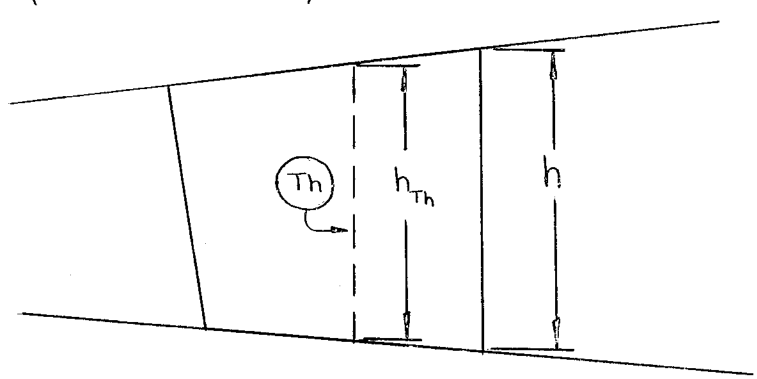
<u>3.2.7 Effective Area Relationship</u> - The basic calculation procedure is based on the flow continuity calculation stations placed between the blade rows at the trailing edge. For subsonic pressure ratios, the effective flow angle is held constant at the input radial distribution, or the effective flow angle is calculated from the passage area with a flow coefficient applied to the throat and an annulus area change between the throat station and the exit station. The equation relationship between the passage area and the effective angle is:

<u>Stators:</u>

$$\alpha 1_{i,k} = \cos^{-1}\left((ndos_{i,k} * Cfs_{i,k}) \Big/ \left[(Se/Sth)_k * \pi * Dp1_{i,k} * \sqrt{\eta s_{i,k}}\right]\right)$$

<u>Rotors:</u>

$$\beta 2_{i,k} = \cos^{-1}\left((ndor_{i,k} * Cfr_{i,k}) \Big/ \left[(Re/Rth)_k * \pi * Dp2_{i,k} * \sqrt{\eta r_{i,k}}\right]\right)$$



$$n * do * Cf * hth = \pi * Dp * h * \cos \alpha * \sqrt{\eta}$$

Thus with the use of a flow coefficient and a velocity coefficient, it is assumed that the flow is isentropic to the throat, and the expansion loss occurs between the throat and the exit. There is no provision for an annulus blockage.

3.2.8 Semi-Perfect Gas - The equations of motion, as utilized in the program, are based on a constant molecular weight. The energy absorbed does not account for any heat released due to composition change. Thus the gas constant RG is not a function of temperature or pressure. Provision is made to input the gas constant, or an option is provided to calculate the gas constant for mixtures of air, JP4 fuel and water vapor.

$$RG = 53.35045 + (.658 * F_{air} + 32.433 * W_{air}) \Big/ (1. + F_{air} + W_{air})$$

Thus for $F_{air} = 0.$ and $W_{air} = 0.$, the standard value for dry air is obtained.

The specific heat at constant pressure for a mixture of products of combustion of JP4 fuel in air can be determined as follows

$$Cpg = (Cpa + Cpf * F_{air} + Cpw * W_{air}) \Big/ (1. + F_{air} + W_{air})$$

A polynomial curve fit versus temperature is provided for the specific heats as shown in the following table.

T(°R)	Cpa Value	Cpa Curve Fit	Cpf Value	Cpf Curve Fit	Cpw Value	Cpw Curve Fit
100	.2392	.2404				
200	.2392	.2391				
300	.2392	.2385				
400	.2393	.2386	.3873	.3877	.4421	.4418
500	.2396	.2393	.4330	.4334	.4439	.4439
600	.2403	.2405	.4728	.4722	.4473	.4477
700	.2416	.2421	.5064	.5054	.4527	.4529
800	.2434	.2442	.5344	.5341	.4592	.4593
900	.2458	.2465	.5591	.5596	.4667	.4666
1000	.2486	.2491	.5819	.5825	.4448	.4745
1100	.2516	.2518	.6031	.6035	.4833	.4829
1200	.2547	.2547	.6230	.6234	.4920	.4918
1300	.2579	.2577	.6422	.6424	.5008	.5008
1400	.2611	.2606	.6608	.6608	.5099	.5101
1500	.2642	.2636	.6791	.6788	.5191	.5194
1600	.2671	.2665	.6969	.6966	.5285	.5288
1700	.2698	.2693	.7142	.7140	.5380	.5382
1800	.2725	.2720	.7313	.7311	.5476	.5476
1900	.2750	.2746	.7480	.7477	.5570	.5569
2000	.2773	.2771	.7640	.7638	.5663	.5661
2100	.2794	.2794	.7790	.7793	.5754	.5752
2200	.2813	.2815	.7937	.7941	.5843	.5842
2300	.2831	.2835	.8077	.8081	.5929	.5929
2400	.2848	.2853	.8211	.8214	.6013	.6014
2600	.2878	.2885	.8459	.8457	.6172	.6173
2800	.2905	.2912	.8678	.8674	.6318	.6317
3000	.2929	.2934	.8867	.8869	.6450	.6450
3200	.2950	.2952				
3400	.2969	.2967				
3600	.2986	.2982				
3800	.3001	.2996				
4000	.3015	.3009				

T(°R)	Cpa Value	Cpa Curve Fit
4200	.3029	.3023
4400	.3041	.3038
4600	.3052	.3052
4800	.3063	.3066
5000	.3072	.3078
5200	.3081	.3088
5400	.3090	.3095
5600	.3098	.3099
5800	.3106	.3102
6000	.3114	.3106
6200	.3121	.3115
6400	.3128	.3136

The specific heat ratio for the mixture can be determined as follows:

$$\gamma g = (Cpg/(Cpg - RG))$$

3.2.9 Conservation of Angular Momentum. - The basic calculation for the equations of momentum, continuity and energy are performed at the stator exit station and the rotor exit station. The calculation at the following rotor inlet station and stator inlet station is primarily to find the incidence angle and inlet Mach number for the loss calculation. Because of a possible shift in sector pitchline diameter due to the flow path slope between the blade row exit and the following blade row inlet, the tangential component of velocity is adjusted inversely proportional to diameter to conserve angular momentum.

$$Vu_{1A} = Vu_1 * Dp_1/Dp_{1A}$$

$$Vu_{2A} = Vu_2 * Dp_2/Dp_{2A}$$

3.2.10 Continuity Adjustment. - The axial component of velocity is adjusted for the annulus area change and density change between the blade row exit and the following blade row inlet as well as any weight flow injected between the stations. The continuity adjustment is the only allowance made for the injected weight flow. There is no provision for injection losses, temperature depression due to mixing, or energy extraction of the injected weight flow.

$$Vz_{1A} = Vz_1 * \rho_{s_1}/\rho_{s_{1A}} * A_1/A_{1A} * Rwg_{1A}/Rwg_1$$

$$Vz_{2A} = Vz_2 * \rho_{s_2}/\rho_{s_{2A}} * A_2/A_{2A} * Rwg_{2A}/Rwg_2$$

3.2.11 Conservation of Relative Total Conditions. - As the gas flows from the rotor inlet to the rotor exit station, there may be a radial shift in pitchline diameter, and in the relative coordinate system an additional work term appears due to the radial outflow or inflow. From the energy equation, the relative total temperature is adjusted proportional to the difference in wheel speed squared.

$$Ttr_2 = Ttr_{1A} + \left[(U_2)^2 - (U_{1A})^2\right]/(2gJCp_{1A})$$

The relative total pressure is isentropically adjusted for the relative total temperature change.

$$Ptr_2 = Ptr_{1A} * (Ttr_2/Ttr_{1A})^{\frac{\gamma 1A}{\gamma 1A - 1.}}$$

Thus, there is no loss associated with the relative total pressure change across the rotor due to radial flow shifts.

3.2.12 Profile Averaging - In order to provide an index of stage performance with a radial profile of total temperature and total pressure, stage exit average conditions $\bar{P}$ and $\bar{T}$ are determined based on a weighted average enthalpy and entropy from the radial sectors. The next stage inlet conditions may be taken as uniform at the average value, the radial sector profiles may be used, or a third option which keeps the total temperature profile and "smooths" the total pressure profile may be used. The next stage stator entrance loss is calculated from the previous swirl angle profile in all cases.

The average total temperature at each stage exit is determined by a mass weighted average of each sector total temperature.

$$\bar{T}_t = T_{t_{pitch}} \sum \frac{m(i)}{M} \frac{T_t(i)}{T_{t_{pitch}}}$$

The average total pressure at each stage exit is determined from a mass weighted average entropy of each sector.

$$\bar{P}_t = P_{t_{pitch}} * e^{(power)}$$

where

$$\text{Power} = \sum \frac{m(i)}{M} * \ln \left[\frac{P_{t(i)}}{P_{t_{pitch}}} \right] + \frac{\gamma}{\gamma-1} \left(\sum \left[\frac{m(i)}{M} * \frac{T_{t(i)}}{T_{t_{pitch}}} \right] - \sum \left[\frac{m(i)}{M} * \ln \frac{T_{t(i)}}{T_{t_{pitch}}} \right] \right)$$

For the third option which keeps the total temperature profile and "smooths" the total pressure profile, the pressure profile is obtained by setting the sector entropy equal to the average entropy.

$$P_{t(i)} = \bar{P}_t * (T_{t(i)}/\bar{T}_t)^{\frac{\gamma}{\gamma-1}}$$

4.0 PREPARATION OF COMPUTER PROGRAM

4.1 Objective - The digital computer has provided a valuable engineering analysis tool to determine the internal and overall performance of axial flow turbines. The specific objectives of Task II of this program are to prepare a digital computer program to be written in Fortran IV computer language compatible with the Lewis Research Center IBM directly coupled 7094-44 computer. The analysis equations, flow charts of the subroutines, and a listing of the Fortran IV source program statements are included in this report as Appendices 1, 2, and 3.

4.2 Computer Program Organization - The analysis equations, flow charts of the program, and a listing of the Fortran IV statements are presented in Appendices 1, 2, and 3. The program has been written as twenty-three (23) subroutines called in sequence by a main control program. The functions of the various subroutines are as follows:

NTCP - Main Calling Program

This routine controls the flow of the program calling the input routine, computing stations, and output routines as needed. It increments the stage number and tests for the last stage of the turbine. When it finds the end of the turbine it calls for overall performance output and interstage performance output, tests for axial Mach number limit, pressure ratio increment, previous choke, modifies the proper upstream pressure ratio and resets the stage number, blade row number, etc., to start over.

INIT - Initialization

The initialization routine calls the input routine, tests the stage loss indicator to store stage one (1) loss input data in following stages, sets up sector height, pitch diameter, annulus area, wheel speed, tests for angle input, converts angles to radians, and sets the index registers and forks.

INPUT - Read Input

The INPUT routine reads in basic input and stage input by "NAMELIST". Stage input is assigned to its proper location by comparison against the dummy word, BLANKS, to see if the element is input.

STA01 - Station 0-1 (See Figure 1 for station designation.)

This routine performs the calculations associated with the first stator entrance loss and establishes the turbine weight flow. FLOW1 is called for each sector flow.

FLOW1 - Flow 1

The FLOW1 routine calculates the flow in a stator exit sector. It is called by STA01 or STA1 (depending on whether this is the first stage or not). Supersonic weight flow decrease for supercritical sectors up to pitch sector critical pressure ratio and constant supersonic weight flow function with $\cos\alpha_{1E}$ correction after the pitch sector is critical are handled in this routine.

LOSS1 - Loss 1

The LOSS1 routine computes stator efficiency from stator loss coefficient as a quadratic polynomial function of stator incidence angle.

R - Gas Constant

A number of thermodynamic subroutines are called by those routines already described: Subroutine R is a simple routine for the gas constant, for the products of combustion of JP4 fuel, and/or water vapor in standard air.

GAMMA - Specific Heat Ratio

A simple routine for the specific heat ratio, for the products of combustion of JP4 fuel, and/or water vapor in standard air.

CPA - Constant Pressure Specific Heat, Air

A simple seventh order polynomial curve fit of specific heat at constant pressure for air at low pressures. Limits ($100 < °R < 6400$).

CPF - Constant Pressure Specific Heat, Fuel

A simple seventh order polynomial curve fit of specific heat at constant pressure for burned JP4 fuel. Limits ($400 < °R < 3000$).

CPW - Constant Pressure Specific Heat, Water - A simple seventh order polynomial curve fit of specific heat at constant pressure for water vapor at low pressures. Limits ($400 < °R < 3000$).

PRATIO - Pressure Ratio

A routine to find the subsonic pressure ratio consistent with a flow function per unit area. A successive iteration technique is employed to solve the transcendental equation within a tolerance "Prtol".

CHECK - Error Check

A routine to test the sense lights and set Prever = .True. if a light was on.

STA1A - Station 1A

Routine STA1A computes the inlet flow conditions relative to the rotor and determines the rotor inlet recovery as a function of rotor incidence angle. The tangential component of velocity is adjusted to conserve angular momentum and the axial momentum of velocity is adjusted for weight flow change, area change and density change from STA1.

STA2 - Station 2

This routine performs the rotor exit flow continuity integration, corrects total conditions for diameter change and checks for a choking condition. The check is performed by calling the LOOP subroutine. If no choke exists and the calculation is not on a choke iteration, the program proceeds to the next stage. If the wanted flow is greater than the critical flow, the return from LOOP transfers control back to the main program to perform the necessary iteration.

FLOW2 - Flow 2

A routine similar to FLOW1 to calculate the flow in a rotor exit sector for STA2.

LOSS2 - Loss 2

A routine similar to LOSS1 to compute rotor efficiency from rotor loss coefficient as a quadratic polynomial function of rotor incidence angle.

LOOP - Loop Iteration

Because of the complex logic for iteration to obtain the exact choke point and reducing the interval when a multiple choke occurs, a separate routine has been established to handle the bookkeeping. Routine LOOP is called from the rotor exit stations and the following stator exit stations. Eight possible conditions may arise during the calculation: (1) underflow, (2) no choke, (3) initial choke detection, (4) choke iteration now sub-critical, (5) choke iteration super-critical again, (6) multiple choke, (7) choke iteration complete, (8) supersonic.

STA2A - Station 2A

This routine computes the inlet flow conditions to all stators after the first and computes stator inlet recovery as a function of stator incidence angle. Stage exit average conditions $\overline{Tt}$ and $\overline{Pt}$ are determined based on an average weighted entropy. Three options for an entrance profile are available. Conservation of angular momentum and continuity are similar to STA1A.

STA1 - Station 1

For other than the first stator, this routine is used to compute stator exit flow continuity and check for a choking condition. Operation is similar to STA2.

OVRALL - Overall Performance

A routine to compute overall performance output and some key hub and casing values.

INSTG - Interstage Performance

A routine to compute each blade row interstage performance output. For three sectors or less, hub and casing values are also computed based on free-vortex distribution.

WOUT - Write-out

A routine to write out the interstage performance data from INSTG.

DIAGT - Diagnostic

A routine to diagnose some key parameters when an error in encountered.

PHIM - Phi Maximum

A routine to find the pressure ratio at the maximum flow function per unit area.

4.3 Input - The NAME heading card and TITLE heading card are transmitted as a BCD record by the general I/O statement:

READ (i,n)list

where i = 5

n = Format number

list = 60 character alphameric field

The two heading cards are read for each case.

The stage input is read in via:

NAMELIST/DATAIN/

which is referenced by a READ statement. The input data must meet the specifications of the NAMELIST statement format.

One block of memory to hold one stage of data is set to the dummy word, BLANKS, before each stage is read. Each stage is headed by $ DATAIN, starting in column (2). After each stage is read in, the elements are compared against BLANKS to see if the element is input. If the element is BLANKS, the element is set to 0.0 for input printout. If the element is not BLANKS, the proper number stage value is set to the element via an EQUIVALENCE system. Data input in a previous case remains in memory for succeeding change cases until it is altered with a new input element.

4.3.1 Options

A. Type Case - Two type input cases are available.

Basic input case, Stgch = 1.0

Change type case, Stgch = 0.0

If an input error occurs in a basic type case, change type cases will be skipped until the next basic type case is reached.

B. Gas Properties - The gas constant RG and the specific heat ratio GAMG may be input or calculated by the program. RG is tested to set the fork γF.

C. Flow Area - Geometry may be input as a passage distributed area or the vector flow angle. SDEA (1) of stage 1 is tested for the stator and RDEA (1) of stage 1 is tested for the rotor.

D. Loss Method - Two options to incorporate loss are provided in an expansion kinetic energy coefficient-inlet recovery coefficient method, and a total pressure loss coefficient method. STPLC (1) of stage 1 is tested for the stator and RTPLC (1) of stage 1 is tested for the rotor.

E. Stage Loss Indicator - An indicator SLI is tested to store stage one (1) efficiency, recovery, flow coefficient, and test factor in following stages if they are to be constant throughout the turbine.

F. Profile Averaging - A fork PAF is tested to set the next stage inlet value as follows:

0 - uniform average profile
1 - use exit leaving profile
2 - smooth pressure profile

4.3.2 Input Sheet.

```
                    TURBINE COMPUTER PROGRAM
                        STANDARD OPTION
                          INPUT SHEET

START ALL INPUT CARDS IN COLUMN 2

NAME 1
TITLE 1
$DATAIN  STAGE=   ,
STGCH=          ,
TTIN=           ,PTIN=          ,WAIR=          ,FAIR=          ,
PTPS=           ,DELC=          ,DELL=          ,DELA=          ,
STG=            ,SECT=          ,EXPN=          ,EXPP=          ,
RG=             ,PAF=           ,SLI            ,AACS=          ,
RPM=            ,VCTD=          ,ENDJOB=        ,ENDSTG=        ,

                      INLET RADIAL PROFILE
PCNH=           ,          ,          ,          ,          ,

                         AXIAL STATIONS
        STA. 0     STA. 1     STA.1A    STA. 2     STA. 2A

GAMG=           ,          ,          ,          ,          ,
DR=             ,          ,          ,          ,          ,
DT=             ,          ,          ,          ,          ,
RWG=            ,          ,          ,          ,          ,

                   STATOR RADIAL DISTRIBUTIONS
          ROOT         PITCH       TIP         (FOR THREE SECTORS)

SDIA=           ,          ,          ,          ,          ,
SDEA=           ,          ,          ,          ,          ,
SREC=           ,          ,          ,          ,          ,
SETA=           ,          ,          ,          ,          ,
SCF=            ,          ,          ,          ,          ,
SPA=            ,          ,          ,          ,          ,
SESTH=          ,

                   ROTOR RADIAL DISTRIBUTIONS
          ROOT         PITCH       TIP         (FOR THREE SECTORS)

RDIA=           ,          ,          ,          ,          ,
RDEA=           ,          ,          ,          ,          ,
RREC=           ,          ,          ,          ,          ,
RETA=           ,          ,          ,          ,          ,
RCF=            ,          ,          ,          ,          ,
RPA=            ,          ,          ,          ,          ,
RTF=            ,          ,          ,          ,          ,
RERTH=          ,

ENDJOB=         ,       ENDJOB=1.0 IF LAST CASE
ENDSTG=         $       ENDSTG=1.0 IF LAST STAGE
```

TURBINE COMPUTER PROGRAM
LOSS COEFFICIENT OPTION

USE THIS SHEET TO SUPPLEMENT NORMAL STAGE INPUT SHEET

START ALL INPUT CARDS IN COLUMN 2

REPLACE SREC AND SETA CARDS WITH FOLLOWING 9 CARDS

	ROOT	PITCH	TIP	(FOR THREE	SECTORS)
STPLC=	,	,	,	,	,
SINR=	,	,	,	,	,
SINMX=	,	,	,	,	,
SCPS=	,	,	,	,	,
SCPC=	,	,	,	,	,
SCPQ=	,	,	,	,	,
SCNS=	,	,	,	,	,
SCNC=	,	,	,	,	,
SCNQ=	,	,	,	,	,

REPLACE RREC AND RETA CARDS WITH FOLLOWING 9 CARDS

	ROOT	PITCH	TIP	(FOR THREE	SECTORS)
RTPLC=	,	,	,	,	,
RINR=	,	,	,	,	,
RINMX=	,	,	,	,	,
RCPS=	,	,	,	,	,
RCPC=	,	,	,	,	,
RCPQ=	,	,	,	,	,
RCNS=	,	,	,	,	,
RCNC=	,	,	,	,	,
RCNQ=	,	,	,	,	,

4.3.3 - Input Nomenclature

NAME ①	60 character name and date card
TITLE ①	60 character title card
$DATAIN	NAMELIST name
STAGE	Stage identification number
STGCH	Stage change (1-basic; 0-change case)
TTIN	Inlet total temperature (°R)
PTIN	Inlet total pressure (psia)
WAIR	Water/air ratio* (Omit with RG & GAMG input)
FAIR	Fuel/air ratio* (Omit with RG & GAMG input)
PTPS	Starting pitchline pressure ratio (Pt_0/Ps_1).
DELC	Increment to first blade row choke
DELL	Increment from first to last blade row choke
DELA	Increment to annulus choke
STG	Number of stages(8 maximum)
SECT	Number of sectors (6 maximum)
EXPN	Cosine exponent, negative incidence
EXPP	Cosine exponent, positive incidence
RG	Gas constant,(ft.-lbs./lb.°R)
PAF	Profile averaging fork (0-uniform; 1-profile; 2-smooth pressure)
SLI	Stage loss indicator (0-stage data input; 1-all stages equal)
AACS	Annulus area choke stop
RPM	Speed (rpm)
VCTD	Vector diagram interstage output
ENDJOB	End of job
ENDSTG	End of stages for this case

PCNH	Percent station height distribution
GAMG	Specific heat ratio (γg)
DR	Diameter root (in.)
DT	Diameter tip (in.)
RWG	Ratio of flow to inlet flow
SDIA	Stator design inlet angle (αo) (°reference from axial)
SDEA	Stator design exit angle ($\alpha 1$) (°reference from axial)
SREC	Stator recovery coefficient (ηsr_{opt})
SETA	Stator efficiency coefficient (ηs)
SCF	Stator flow coefficient (Cfs)
SPA	Stator passage area per unit height (n"do", in.^{2}/in.)
SESTH	Stator exit stator throat height ratio.
RDIA	Rotor design inlet angle ($\beta 1A$) (°reference from axial)
RDEA	Rotor design exit angle ($\beta 2$) (°reference from axial)
RREC	Rotor recovery coefficient, (ηrr_{opt})
RETA	Rotor efficiency coefficient (ηr)
RCF	Rotor flow coefficient (Cfr)
RPA	Rotor passage area per unit height (n"do", in.^{2}/in.)
RTF	Rotor test factor
RERTH	Rotor exit rotor throat height ratio.
ENDJOB	End of job.
ENDSTG	End of stages for this case.

STPLC	Stator optimum total pressure loss coefficient
SINR	Stator incidence range
SINMX	Stator incidence maximum value
SCPS	Stator coefficient positive square term
SCPC	Stator coefficient positive cubic term
SCPQ	Stator coefficient positive quartic term
SCNS	Stator coefficient negative square term
SCNC	Stator coefficient negative cubic term
SCNQ	Stator coefficient negative quartic term

RTPLC	Rotor optimum total pressure loss coefficient
RINR	Rotor incidence range
RINMX	Rotor incidence maximum value
RCPS	Rotor coefficient positive square term
RCPC	Rotor coefficient positive cubic term
RCPQ	Rotor coefficient positive quartic term.
RCNS	Rotor coefficient negative square term
RCNC	Rotor coefficient negative cubic term
RCNQ	Rotor coefficient negative quartic term.

4.4 Overall Performance Output. The results of each case of data are precede by a listing of the input used for that case. For a change type case in which mos of the input data is a carry-over from the previous basic input case, only the inp for that case is printed. Carry-over input from the previous basic input appears as zero. The output is headed with a program title and two lines of NAME and TITL identification which were entered with the input. The output will be listed as fixed-point decimal with each line identified at the left. There will be as many columns of output as there are stages computed. The turbine overall values are in vidually identified at the bottom of the stage answers.

4.4.1 - Overall Performance Nomenclature

TURBINE COMPUTER PROGRAM

NAME

TITLE

CASE I. SUBCASE IS

STAGE PERFORMANCE

	STAGE 1	STAGE 2	STAGE 3	STAGE 4

TT 0	Stage inlet total temperature (°R).
PT 0	Stage inlet total pressure (psia)
WG 0	Stage inlet weight flow (lbs/sec.)
DEL H	Stage energy output (BTU/lbs)
WRT/P	Stage corrected weight flow function
DH/T	Stage energy function
N/RT	Stage corrected speed
ETA TT	Stage total to total efficiency

ETA TS	Stage total to static efficiency
ETA AT	Stage total to axial total efficiency
PTO/PS1	Stage stator pitchline pressure ratio
PTO/PT2	Stage total pitchline pressure ratio
PTO/PS2	Stage total-static pitchline pressure ratio
PTR1A/PS2	Stage rotor pitchline pressure ratio
TT2/TTO	Stage total pitchline temperature ratio
TTR1/TTO	Stage rotor inlet pitchline relative total temperature ratio
PS 1	Rotor inlet pitchline pressure (psia)
TTR 1	Rotor inlet pitchline relative total temperature (°R)
PTR 1	Rotor inlet pitchline relative total pressure (psia)
PS 2	Stage exit pitchline static pressure (psia)
TT 2	Stage exit pitchline total temperature (°R)
PT 2	Stage exit pitchline total pressure (psia)
UP/VI	Pitchline wheel speed to isentropic velocity ratio.
UR/VI	Root wheel speed to isentropic velocity ratio
PSI P	Pitchline kinetic energy loading parameter
PSI R	Root kinetic energy loading parameter
RX P	Pitchline reaction ratio
RX R	Root reaction ratio
ALPHA 0	Stator inlet gas angle
I STATOR	Stator inlet incidence angle
BETA 1A	Rotor inlet gas angle
I ROTOR	Rotor inlet incidence angle
ALPHA 2	Stage leaving swirl angle
DBETA R	Rotor root turning angle

M 1	Stator exit pitchline Mach number
M1 RT	Stator exit root Mach number
MR 1A	Rotor inlet pitchline relative Mach number
MR1A RT	Rotor inlet root relative Mach number
MR 2	Rotor exit pitchline relative Mach number
MR2 TIP	Rotor exit tip relative Mach number
E/TH CR	Stage equivalent energy, corrected to standard inlet critical velo
N/RTH CR	Stage equivalent speed, corrected to standard inlet critical veloci
WRTHCRE/D	Stage equivalent flow, corrected to standard inlet critical velocit

OVRALL PERFORMANCE

PSI P	Overall pitchline kinetic energy loading parameter
PSI R	Overall root kinetic energy loading parameter
DEL H	Overall energy output (BTU/lb.)
WRT/P	Inlet corrected weight flow function
N/RT	Inlet corrected speed
DELH/T	Overall energy function
PTO/PT2	Overall total pressure ratio
PTO/PS2	Overall total static pressure ratio
PT/PAT2	Overall total to axial total pressure ratio
ETA TT	Overall total to total efficiency
ETA TS	Overall total to static efficiency
ETA TAT	Overall total to axial total efficiency
WNE/60D	Inlet equivalent flow speed parameter
N/RTH CR	Inlet equivalent speed
E/TH CR	Overall equivalent energy

4.5 Interstage Performance Output. The interstage performance output is obtained when VCTD = 1.0. Because of the large volume of output, it is not generally recommended when DELC, DELL, and DELA are non-zero in the course of generating an entire turbine map. It is generally useful for obtaining detailed output at specific operating points. One page of output is obtained for each blade row in the turbine. Detailed radial distribution of vector diagram conditions as well as state properties and loading as expressed by incompressible forms of Zweifel parameter and pressure coefficient normalized by exit conditions.

4.5.1 - Interstage Performance Nomenclature.

TURBINE COMPUTER PROGRAM

NAME

TITLE

CASE I. SUBCASE IS

INTER-STAGE PERFORMANCE

STA 0	STATOR INLET
DIAM 0	Diameter (in)
TT 0	Total Temperature (°R)
PT 0	Total Pressure (psia)
ALPHA 0	Absolute Angle (°)
I STATOR	Incidence Angle (°)
V 0	Velocity (ft/sec)
VU 0	Tangential Velocity (ft/sec)
VZ 0	Axial Velocity (ft/sec)
TS 0	Static Temperature (°R)

PS 0	Static Pressure (psia)
DENS 0	Static Density (lb/ft3)
M 0	Mach Number
STA 1	STATOR EXIT
DIAM 1	
ALPHA 1	
DEL A	Turning Angle (°)
V 1	
V 1	
VU 1	
VZ 1	
TS 1	
PS 1	
DENS 1	
M 1	
ZWI INC	Zweifel Parameter, Incompressible
CP S	Pressure Coefficient, Incompressible

TURBINE COMPUTER PROGRAM

NAME

TITLE

CASE I. SUBCASE IS

INTER-STAGE PERFORMANCE

STA 1A	ROTOR INLET
DIAM 1A	
PTR 1A	Relative Total Pressure (psia)
TTR 1A	Relative Total Temperature (°R)
BETA 1A	Relative Angle (°)
I ROTOR	Incidence Angle (°)
R 1A	Relative Velocity (ft/sec)
RU 1A	Relative Tangential Velocity (ft/sec)
U 1A	Wheel Velocity (ft/sec)

STA 2	ROTOR EXIT
DIAM 2	
BETA 2	
DBETA	Turning Angle (°)
R 2	
RU 2	
MR 2	Relative Mach Number
U 2	

RX	Reaction
DELH	Energy (BTU/lb)
PSI P	Energy Loading Parameter
ETA TT	Efficiency Total - Total
ETA TS	Total - Static
ETA AT	Total - Axial Total
ZWI INC	
CP R	
PT 2A	
TT 2A	
V 2A	
VU 2A	
ALPHA 2A	
MF 2A	Axial Mach Number
VZ 2A	
TS 2A	
PS 2A	
DENS 2A	
M 2A	

4.6 Techniques

4.6.1 PRATIO - The subroutine PRATIO determines the pressure ratio which is consistent with a flow function per unit area. Since the relationship is a transcendental function of Mach number, a successive iteration is employed.

From the compressible flow properties:

$$Pt/Ps = \left(1. + \frac{\gamma - 1}{2.} M^2\right)^{\frac{\gamma}{\gamma - 1}} \qquad (1)$$

and

$$W\sqrt{Tt}/Pt_A = M \sqrt{\gamma g/R} \sqrt{Tt/Ts} \; (Pt/Ps)^{-1} \qquad (2)$$

solving (1) for M and substituting into (2)

$$W\sqrt{Tt}/Pt_A = \sqrt{2g\gamma/R\ (\gamma-1.)} \sqrt{(Pt/Ps)^{\frac{\gamma-1}{\gamma}} - 1.} \sqrt{(Pt/Ps)^{\frac{\gamma-1}{\gamma}}} \sqrt{(Pt/Ps)^{\frac{-2\gamma}{\gamma}}} \qquad (3)$$

or

$$W\sqrt{Tt}/Pt_A \sqrt{(R/2g)\left[(\gamma-1)/\gamma\right]} = f(M) = \sqrt{(Pt/Ps)^{-\frac{2}{\gamma}} - (Pt/Ps)^{-\frac{\gamma+1}{\gamma}}} \qquad (4)$$

Therefore:

The function of Mach number calculated from the fight side of Equation (4) minus the function of Mach number calculated from the left side of Equation (4) approaches zero error as Pt/Ps approaches the correct answer.

An upper limit is set at the critical value and a lower limit is set at unity. An average value is selected and Equation (4) is computed. If the error in the function is positive, Pt/Ps trial is too large, the upper limit is reduced to the previous trial value, a new average value is selected and Equation (4) is recomputed. If the error in the function is negative, Pt/Ps trial is too small, the lower limit is increased to the previous trial value, a new average value is selected and Equation (4) is recomputed. The iteration is complete when pressure ratio error $\left| Pt/Ps - Pt/Ps_{previous\ trial} \right|$ is less than tolerance Prtol.

4.6.2 PHIM - The subroutine PHIM determines the pressure ratio at which the flow function is a maximum including the efficiency of expansion $_S$ or $_R$. This critical pressure ratio is then used for the supersonic overexpansion criteria.

From the definitions:

$$\eta_S = \frac{V^2/2gJCpTt}{1 - (1/\emptyset)}$$

The flow function per unit area becomes

$$W\ \sqrt{Tt}/PtA = \sqrt{(2g/R)\ \gamma/(\gamma-1)}\ \ \sqrt{\eta\,(1 - 1/\emptyset)}\Big/ [\emptyset]^{\frac{\gamma}{\gamma-1}}\ \left(1 - \eta\left[1 - 1/\emptyset\right]\right)$$

with a substitution of variables.

Let

$$X = \eta(1 - 1/\emptyset)$$

the derivative of flow function with respect to X evaluated at zero yields

$$X = -B - \sqrt{B^2 - 4AC}\Big/2A$$

where

$$A = \left(\frac{\gamma}{\gamma-1} - \frac{1}{2}\right)$$

$$B = -\left(\frac{\gamma}{\gamma-1} + \frac{1-\eta}{2}\right)$$

$$C = \eta/2$$

4.6.3 Station Continuity - The continuity of weight flow at a stator exit station and a rotor exit station is handled by a summation of sector flows. A starting value of the pitch sector pressure ratio is the one-dimensional value obtained from subroutine PRATIO. Simple radial equilibrium is then used to determine the pitchline pressure ratio of the adjacent sectors until the hub and tip sector weight flows are complete. The calculated total weight flow is compared with the desired weight flow and the error in weight flow is compared with a weight flow tolerance. The iteration to reduce the error to less than the tolerance is handled similar to PRATIO. A lower limit is set at unity and an upper limit is computed on each pass to produce critical flow in the low pressure ratio sector, based on the radial pressure gradient of the previous pass:

$$\frac{\text{Pt/Ps pitchline upper limit}}{\text{Pt/Ps critical}} = \frac{\text{Pt/Ps previous pitchline}}{\text{Pt/Ps previous limiting sector}}$$

An average value is selected for the pitch sector pressure ratio and the total weight flow from all the sectors is computed. If the error in weight flow is positive, Pt/Ps trial is too large, the upper limit is reduced to the previous trial value, a new average value is selected and the total weight flow from all the sectors is recomputed. If the error in the weight flow is negative, Pt/Ps trial is too small, the lower limit is increased to the previous trial value, a new average value is selected and the total weight flow is recomputed. The iteration is complete when the error in weight flow |weight flow calculated - weight flow wanted| is less than tolerance Wtol.

4.6.4 Forks and Indicators

A. PRPC - Indicator for pressure ratio at pitchline, critical. Initialized at 0.0 in INIT, set at 1.0 in FLOW1 or FLOW2 when pitch sector first exceeds critical and is reset to critical, set at 2.0 in FLOW1 or FLOW2 after critical. Reset by STA1 or STA2 to 0.0 when pitch sector pressure ratio is less than critical. Main program saves PRPC in CS(K) or CR(K) and resets PRPC on next calculation.

B. PREVER - Logical constant for previous error. Set .FALSE. at beginning of MAIN program. Set to .TRUE. by CHECK when a sense light is on. Set to .TRUE. by subroutines at known errors. Tested by MAIN to end case calculation.

C. MFSTOP - Axial Mach number limit set in STA2A at Mf2A pitchline/AACS. Thus, when AACS < 1.0, calculation can be stopped before annulus area choke is reached.

D. JUMP - Fork set by LOOP to indicate whether subcritical path or supercritical path. Set at 0 in LOOP for no-choke or choke iteration complete. Set at 1 in LOOP for choke iteration. Tested in the MAIN program to go to next calculation station or go by on choke iteration.

E. LOPIN - Fork set in MAIN program at entry to each calculation station to indicate previous calculation was previous upstream station (LOPIN = 0), or a downstream station on a choke iteration (LOPIN = 1). Tested by STA1 and STA2 to bypass initial starting pressure ratio calculation and continuity iteration on station where choke previously occurred.

F. SCRIT - Indicator set by FLOW1 or FLOW2 that station critical flow has been reached when the last sector exceeds critical. Initialized to 0.0 at beginning of STA01, STA2, and STA1 on each entry. Tested by LOOP for subcritical or supercritical branch.

G. PTRN - Indicator that a negative sector pressure ratio has been reached due to the inlet total pressure profile and the exit static pressure profile.

H. WTOL - Weight flow error tolerance used in STA2 and STA1. When the change in pressure ratio produces a pressure ratio error which is less than a pressure ratio tolerance PRTOL, however, the weight flow error is still larger than the weight flow tolerance WTOL, a choked condition is assumed and a choke iteration is started.

I. RHOTOL - Density error tolerance used in STA1A and STA2A to correct axial velocity.

J. PRTOL - Pressure ratio tolerance used in PRATIO iteration and STA2 and STA1 continuity iteration. Tested by LOOP for accuracy of choke point calculation.

K. TRLOOP - Switch for an output trace of the LOOP iteration. The calculation station writes:

$$Wgt_2, Wgt_{2c}, Wg_2(i = 1, Isect), Pt/Ps_2(i = 1, Isect)$$

(2 is replaced by 1 for station 1)

Then LOOP writes:

IBRC, LBRC, ISORR, KN, LSTG, IPC, ISS, ICHOKE, JUMP, LBRCS, ISORRS, LSTGS
SPTPS, Pt_0/Ps_1, DELPR, DELL, SCRIT, LOPC

L. LSTG - Last stage completed.

M. LBRC - Last blade row completed.

N. IBRC - Index of present blade row counter.

O. ICHOKE - Iteration on choke (0 - no choke).

P. ISORR - Stator or rotor index (1 - stator, 2 - rotor).

R. PTOPSI(IP,K) - Computed memory location of pitchline pressure ratio. Since PTRS2 follows PTOPI1 in memory order, the rotor pressure ratio can be obtained by an index on the stator pressure.

S. TRDIAG - Switch for an output trace of an intermediate calculation diagnostic.

T. SC - Stator choke indicator. Set equal 1. when first stator is critical flow.

U. RC - Not used.

V. DELPR - Pressure ratio increment. Set to DELC by INIT, set to DELL by MAIN program after first stator has critical flow, set to DELL by LOOP when choke iteration is complete and DELA on last rotor. On initial choke detection and choke iteration:

$$DELPR = 1/2\ DELPR$$

On multiple choke:

$$DELPR = 1/4\ DELPR$$

W. PASS - Indicator that one calculation has been completed. Tested by LOOP to reduce pressure ratio by DELPR = DELC if initial pressure ratio is over critical.

X. IPC - Indicator for previous choke completed. Set by LOOP equal to IBRC on choke iteration completion. Tested by LOOP to determine multiple choke. Reset to zero by LOOP on multiple choke detection at reduced pressure ratio.

Y. LOPC - LOOP counter.

Z. ISS - Indicator for super-sonic pressure ratios. Set by LOOP to IBRC on choke iteration completion. Tested LOOP to find upstream station after flow is choked.

4.7 Program Operation - The program was written in FORTRAN IV primarily for the IBM 7090/7094 IBSYS operating system - version 12. The system and the object program do not exceed the capacity of core storage so that an "Overlay" is not required and no special tape units are called.

4.7.1 Tape Unit Assignment

A. Normal operation - no source.

Logical Unit	IBSYS Name	Function
01	SYSUT1	System Utility Unit 1
02	SYSUT2	System Utility Unit 2
03	SYSUT3	System Utility Unit 3
04	SYSUT4	System Utility Unit 4
05	SYSIN1	System Input Unit
06	SYSOU1	System Output Unit
	SYSLB1	ISBYS system and library

B. Compiler Operation - source, deck.

		Above units and additional
07	SYSPP1	System Peripheral Punch

4.7.2 Loader Deck Names - The deck names were selected as four or less alphameric characters, rather than use the six characters subroutine name. Since the identification punched by the compiler in Column 73-80 is the first four characters of the deck name, identification symbols were stripped from FLOW1, FLOW2, STA01, STA1A, and STA2A in the object deck when the subroutine name was used.

5.0 ANALYSIS OF NASA REFERENCE TURBINE

In order to demonstrate the use of the computer program the turbine performance map was calculated for the design of Reference 2. The turbine geometry as provided by Lewis Research Center are shown in the following table.

NASA REFERENCE TWO-STAGE TURBINE

TTIN = 700°R

$N/\sqrt{\theta}$ = 60 — 120 percent design (N = 5041 rpm)

PTIN = 34.9 in. Hg

STG = 2

SECT = 5

RG = 53.3

PCNH = .2, .2, .2, .2, .2

STAGE 1

GAMG 1 = 1.40, 1.40, 1.40, 1.40, 1.40

DR 1 = 19.110, 19.110, 18.969, 18.406, 18.265

DT 1 = 28.000, 28.000, 28.141, 28.704 28.845

RWG 1 = 1., 1., 1., 1., 1.

SDIA 1 = .0, .0, .0, .0, .0

SDEA 1 = ---------------------------

SREC 1 =

SETA 1 =

SCF 1 =

SPA 1 = 22.140, 26.035, 30.135, 34.194, 38.499

SESTH 1 = 1.00

RDIA 1 = 58.6, 52.9, 46.1, 38.2, 28.9

RDEA 1 = ---------------------------

RREC 1 =

RETA 1 =

RCF 1 =

RTF 1 =

RPA 1 = 33.408, 36.352, 38.976, 41.280, 43.008

RERTH 1 = 1.01

STAGE 2

GAMG 2 = 1.40, 1.40, 1.40, 1.40, 1.40

DR 2 = 18.265, 17.814, 17.673, 17.110, 17.110

DT 2 = 28.845, 29.296, 29.437, 30.000, 30,000

RWG 2 = 1., 1., 1., 1., 1.

SDIA 2 = 33.0, 30.4, 28.2, 26.3, 24.6

SDEA 2 = ---------------------------

SREC 2 =

SETA 2 =

SCF 2 =

SPA 2 = 30.420, 36.855, 43.485, 50.765, 58.240

SESTH 2 = 1.01

RDIA 2 = 44.6, 34.9, 24.1, 12.6, 1.3

RDEA 2 = ---------------------------

RREC 2 =

RETA 2 =

RCF 2 =

RTF 2 =

RPA 2 = 43.350, 48.150, 52.350, 55.750, 58.550

RERTH 2 = 1.01

A parametric variation of loss definition parameters and corrected speed was completed to establish the flow and efficiency characteristics to produce 92.5 percent overall rating efficiency at 100 percent design speed at a rating pressure ratio of 2.60. By comparison of calculated results with the known performance data it was observed that peak efficiency was calculated near 110 percent design speed whereas the test data peaked at 120 percent design speed. The drop-off in peak efficiency with speed was calculated to be less than the test data.

In order to obtain a better match with the test data it was assumed that optimum incidence angle occurred at the design value -8°. In addition, the cosine exponent for off-incidence angle recovery factor was increased from 2.0 to 3.0 to better match the drop-off in peak efficiency with speed, and a decreasing flow coefficient with blade row number was assumed to produce a reduction equivalent flow with speed.

5.1 Input Sheets - Shown in Appendix 4 are the input sheets for NASA Reference Two-Stage Turbine. The input deck is listed in Appendix 6.

5.2 Output Listing - Shown in Appendix 5 are the output listing near the design point for NASA Reference Two-Stage Turbine.

5.3 Results - Shown in Figures 10 and 9 are the performance maps for NASA Reference Two-Stage Turbine in terms of equivalent weight-flow versus overall rating total-pressure ratio for lines of equivalent speed, and equivalent shaft work versus equivalent weight-flow-speed parameter with contours of rating pressure ratio, equivalent speed and rating efficiency. A table of variation of significant parameters along the peak efficiency ridge is given in the following table:

$\%N/\sqrt{\theta cr}$ design	60	80	100	120
$WN\epsilon/60\delta$	1470.7	2361.1	3142.1	3804.3
$\Delta h/\theta cr$	8.14	15.65	24.69	33.08
Pto/Pat_2	1.298	1.671	2.308	3.192
η tat	.915	.923	.935	.942
$Ir,_1$	10.09	10.37	8.23	1.92
$Is,_2$	-2.01	.83	.13	-6.33
$Ir,_2$	-11.72	- 5.2	-.73	.06
$\alpha_{2,2}$	-25.48	-17.61	-9.3	-.25
$Mf,_2$	.15	.23	.34	.49

It can be seen that the efficiency peak occurs at 120 percent design speed. The drop-off in peak efficiency with speed characteristics is similar to the test data.

An evaluation of significant parameters along a constant 100 percent speed line with varying pressure ratio is given in the following table:

%N/ $\overline{\theta cr}$ design	100		
$WN\epsilon/60\delta$	2684.4	3142.1	3198.3
$\Delta h/\theta cr$	11.21	24.69	34.28
Pto/Pat_2	1.523	2.308	3.597
ηtat	.801	.935	.899
$Ir,_1$	-9.36	8.23	11.01
$Is,_2$	-33.88	.13	7.59
$Ir,_2$	-41.96	-.73	13.89
$\alpha_{2,2}$	-46.22	-9.30	18.64
$Mf,_2$	.19	.34	.58

The drop-off in efficiency with pressure ratio is similar to the test data, however, at positive incidence angles the calculated result was lower in efficiency than the reported test data.

6.0 REFERENCES

1. Dunavant, J. and Erwin, J.: Investigation of a Related Series of Turbine Blade Profiles in Cascade. NACA Tech Note 3802, 1956.
2. Schum, H., Petrash, D. and Nunamaker, R.: Experimental Investigation of Two-Stage Air-Cooled Turbine Suitable For Flight at Mach 2.5. NASA Tech Memo X148, 1959.
3. Wu,C.H.: General Theory of Three-Dimensional Flow in Subsonic and Supersonic Turbomachines of Axial, Radial, and Mixed Flow Types. NACA Tech Note 2604, 1952.
4. Zweifel, O.: The Spacing of Turbo-Machine Blading Especially With Large Angular Deflection. Brown Boveri Review 32, 1945.

7.0 NOMENCLATURE FOR ANALYSIS

A	-	Coefficient of polynomial
Ann	-	Annulus area (in^2)
B	-	Coefficient of polynomial
Cf	-	Flow coefficient
Cp	-	Specific heat at constant pressure (ft/°R)
Dp	-	Diameter, pitch (in)
Dr	-	Diameter, root (in)
Dt	-	Diameter, tip (in)
FF	-	Flow function ($in^2 T^{1/2}$/sec)
FF/A	-	Flow function per unit area ($T^{1/2}$/sec)
FA1A	-	Fuel/air ratio
g	-	Gravity constant (ft/sec^2)
h	-	Sector height (in)
I	-	Incidence angle (°)
J	-	Mechanical equivalent of heat (ft lb/BTU)
M	-	Mach Number
Mf	-	Axial Mach Number
n do	-	Passage distributed area (in)
Prtol	-	Pressure ratio tolerance
Pre	-	Pressure ratio error
Ps	-	Static pressure (psia)
Pt	-	Total pressure (psia)
Ptp	-	Total pressure profile (psia)
Ptr	-	Total pressure ratio
Ptrmo	-	Previous total pressure ratio
r	-	Ratio
R	-	Ratio
R	-	Relative velocity (ft/sec)
Rg	-	Gas constant (ft/°R)
rpm	-	Speed (rev/min)
Rx	-	Reaction
Sh	-	Station height (in)
Scrit	-	Station critical

TF	-	Test factor
Ta	-	Average temperature (°R)
Ts	-	Static temperature (°R)
Tt	-	Total temperature (°R)
U	-	Wheel velocity (ft/sec)
V	-	Velocity (ft/sec)
Wair	-	Water/air ratio
Wg	-	Weight flow (lb/sec)
Wgt	-	Total weight flow (lb/sec)
Wr	-	Flow ratio
X	-	Array
Y	-	Array
Z	-	Zweifel parameter
α	-	Absolute flow angle (°reference from axial)
αF	-	Alpha fork
β	-	Relative flow angle (°reference from axial)
βF	-	Beta fork
γ	-	Specific heat ratio
γF	-	Gamma fork
Δ	-	Incremental change
δ	-	Standard pressure correction
d	-	Differential operator
ϵ	-	Standard gamma correction
η	-	Efficiency
θ	-	Standard temperature correction
π	-	Circle circumference/diameter
ρe	-	Density error
ρs	-	Static density (lb/ft^3)
ρtol	-	Density tolerance
$\emptyset$	-	Temperature ratio
ψ	-	Loading parameter
ω	-	Total pressure loss coefficient
%	-	Percent

SUBSCRIPTS

0	-	Station designation
1	-	Station designation
1A	-	Station designation
2	-	Station designation
2A	-	Station designation
at	-	Axial total
c	-	Calculated
c	-	Cubic coefficient
c	-	Critical
cr	-	Critical
crit	-	Critical
e	-	Effective
h	-	Hub
i	-	Sector counter
I	-	Isentropic
in	-	Inlet
inc	-	Incompressible
Ip	-	Pitchline sector
Isect	-	Number of sectors
isen	-	Isentropic
j	-	Index counter (general)
k	-	Stage counter
Kstg	-	Number of stages
ℓ	-	Index counter (general)
m	-	Multiplier
mn	-	Maximum negative limit
mp	-	Maximum positive limit
o	-	Overall
p	-	Pitchline
q	-	Quartic coefficient
r	-	Root
r	-	Rotor
r	-	Relative
r	-	Range

rr	-	Rotor recovery
s	-	Static
s	-	Stator
s	-	Stage
s	-	Square coefficient
sl	-	Standard sea level
sr	-	Stator recovery
t	-	Total
t	-	Tip
th	-	Throat
tr	-	Trial
u	-	Tangential component
vd	-	Vector diagram
z	-	Axial component

<u>SUPERSCRIPTS</u>

o	-	Degrees
*	-	Blade angle

FIGURE 1

FLOW-STREAM-TUBE VERSUS PROPORTIONAL AREA

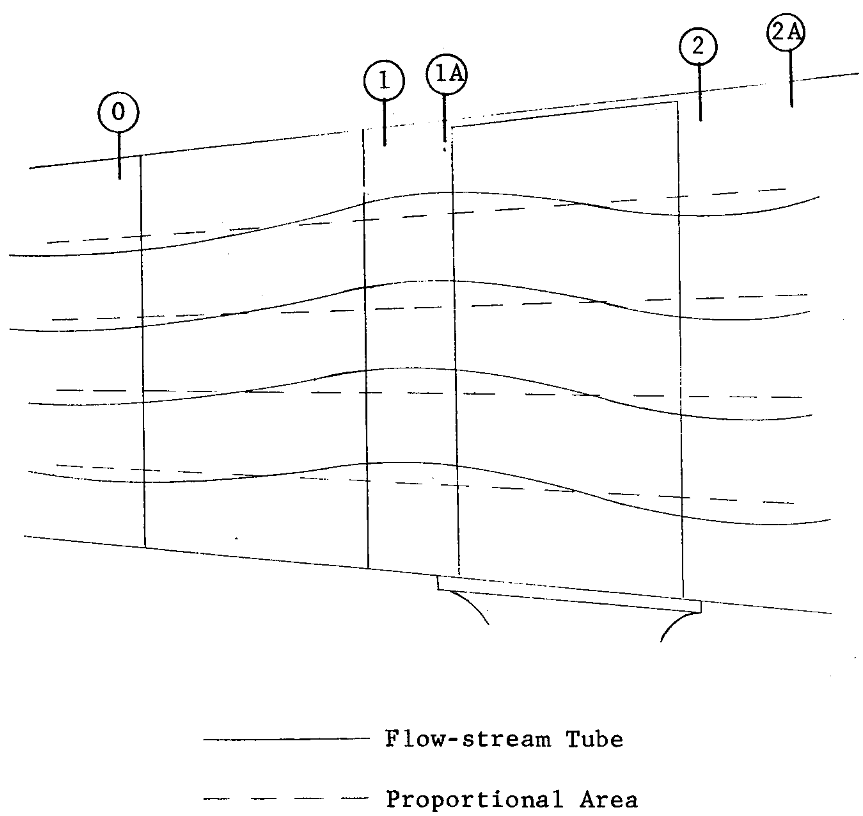


FIGURE 2

THREE TEST CASES

ERROR EVALUATION FOR PROPORTIONAL AREA SECTORS

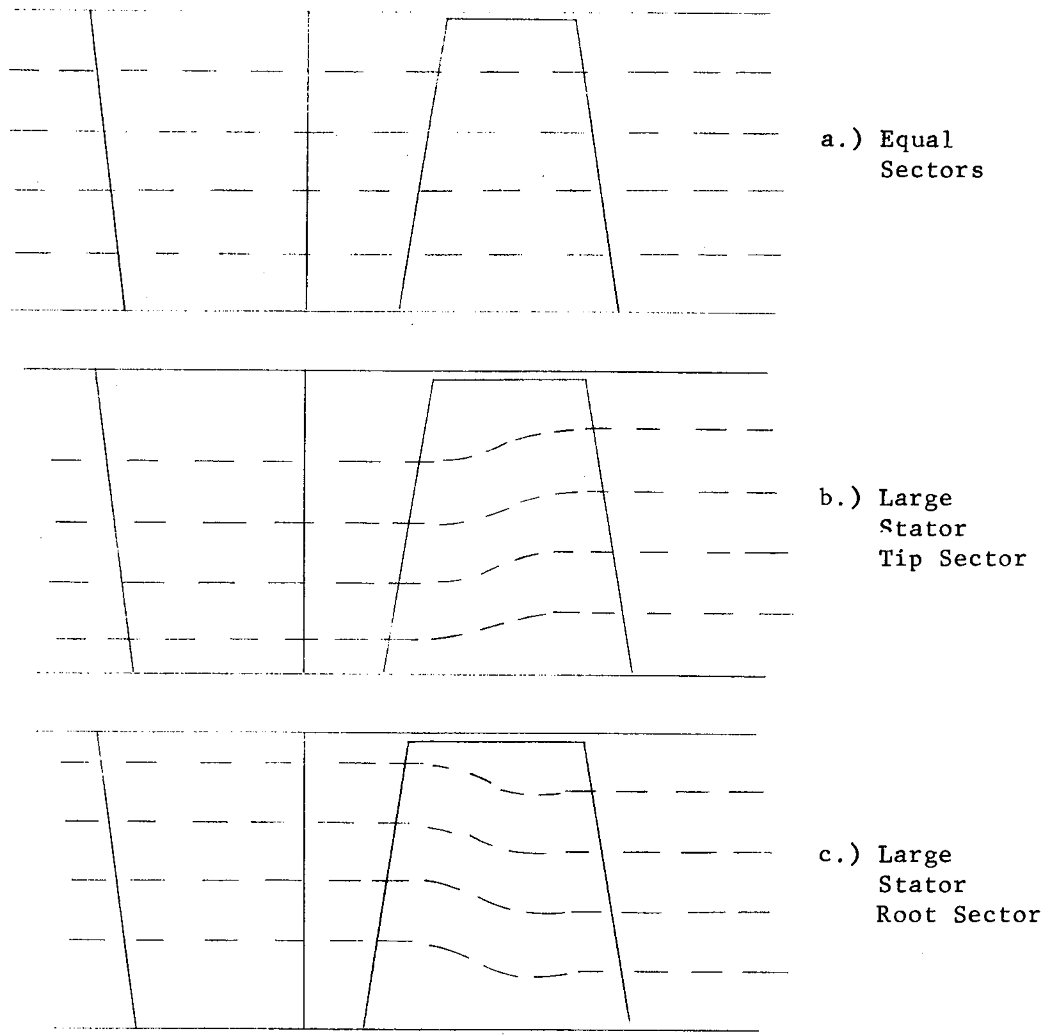


FIGURE 3

MOMENTUM AND ENERGY TRANSPORT APPROXIMATION

FREE VORTEX STATOR ANGLE DISTRIBUTION

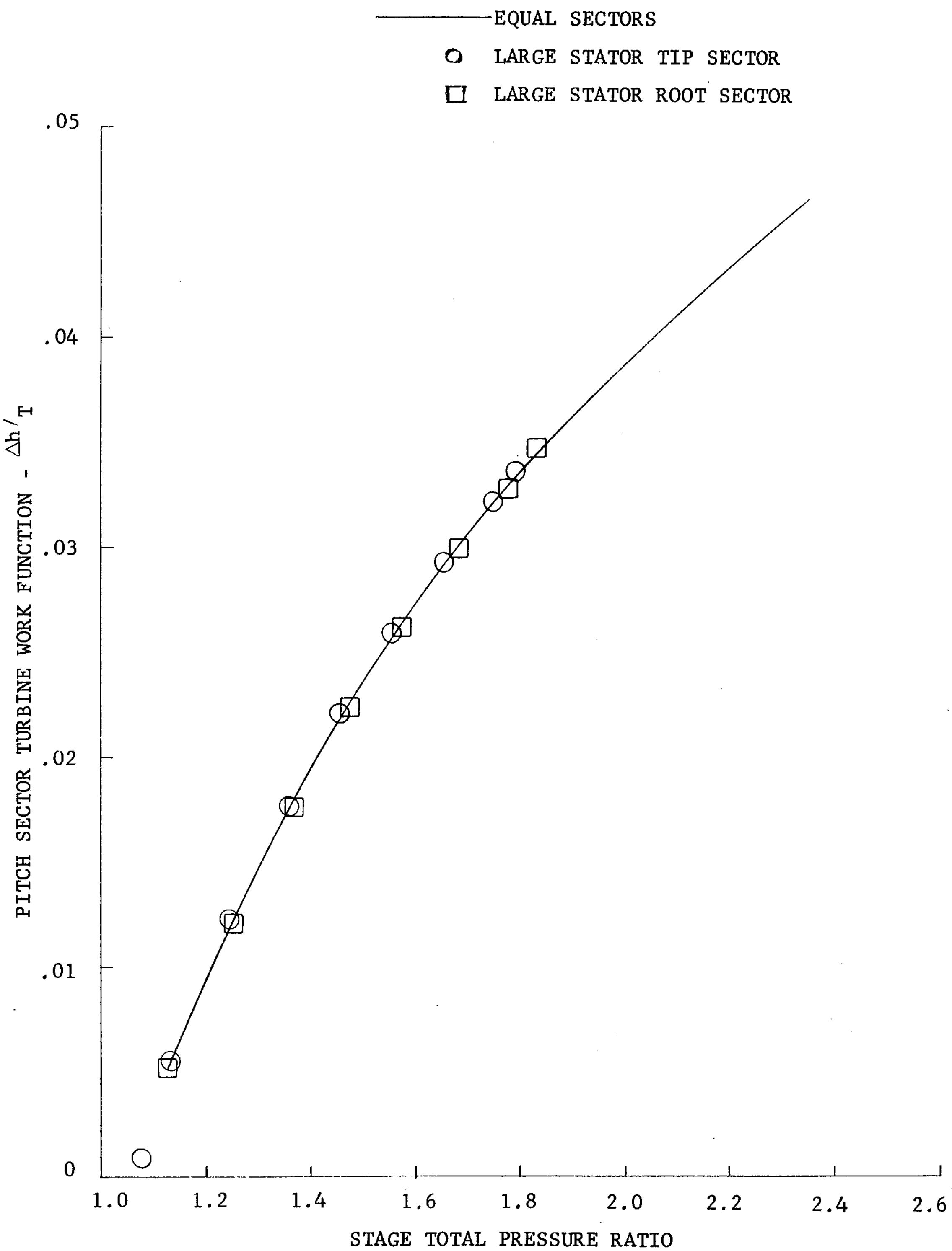


FIGURE 4

MOMENTUM AND ENERGY TRANSPORT APPROXIMATION

CONSTANT STATOR ANGLE

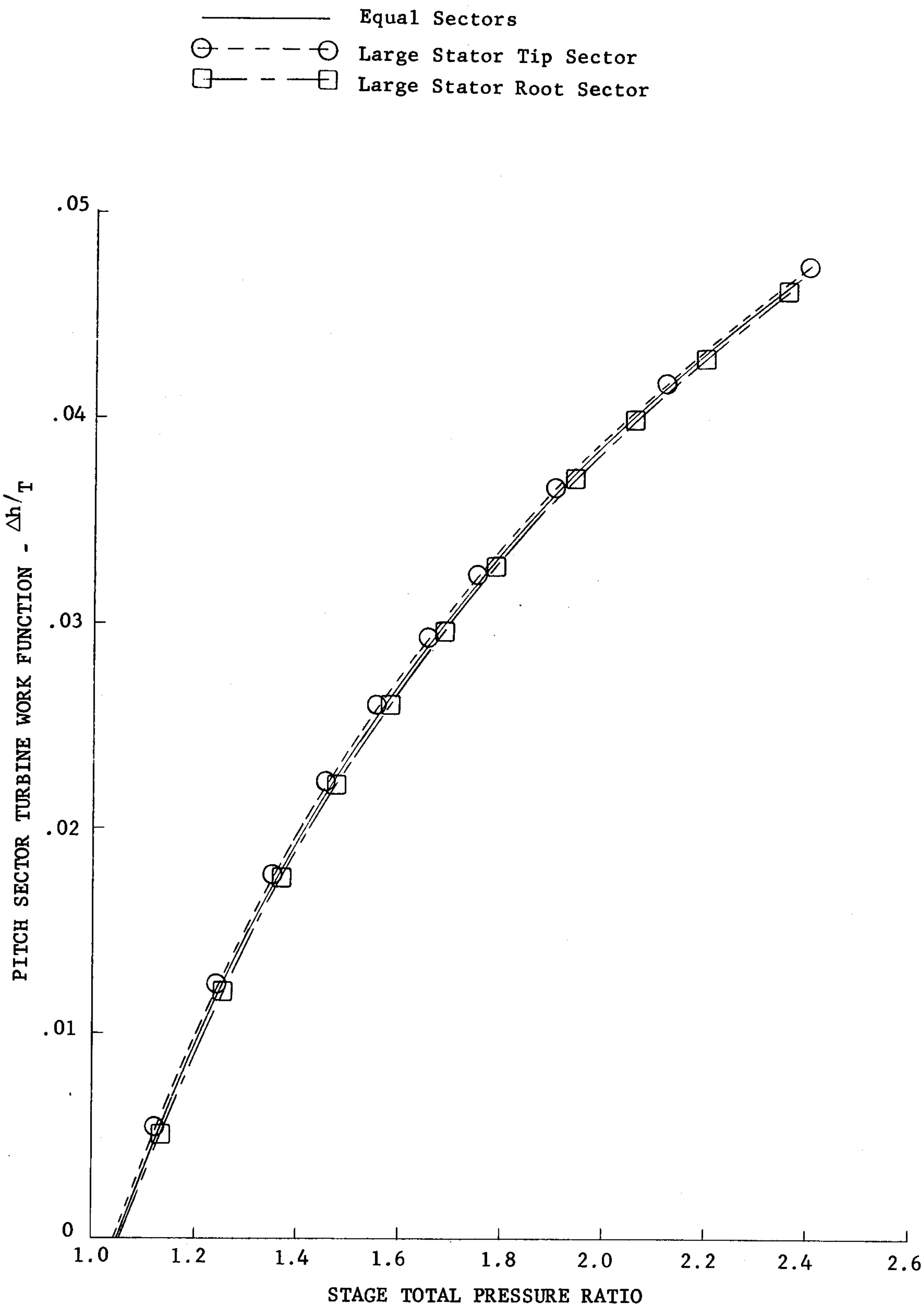


FIGURE 5
TEST CASE 1.
STATOR SECTOR WEIGHT FLOW VERSUS SECTOR PRESSURE RATIO

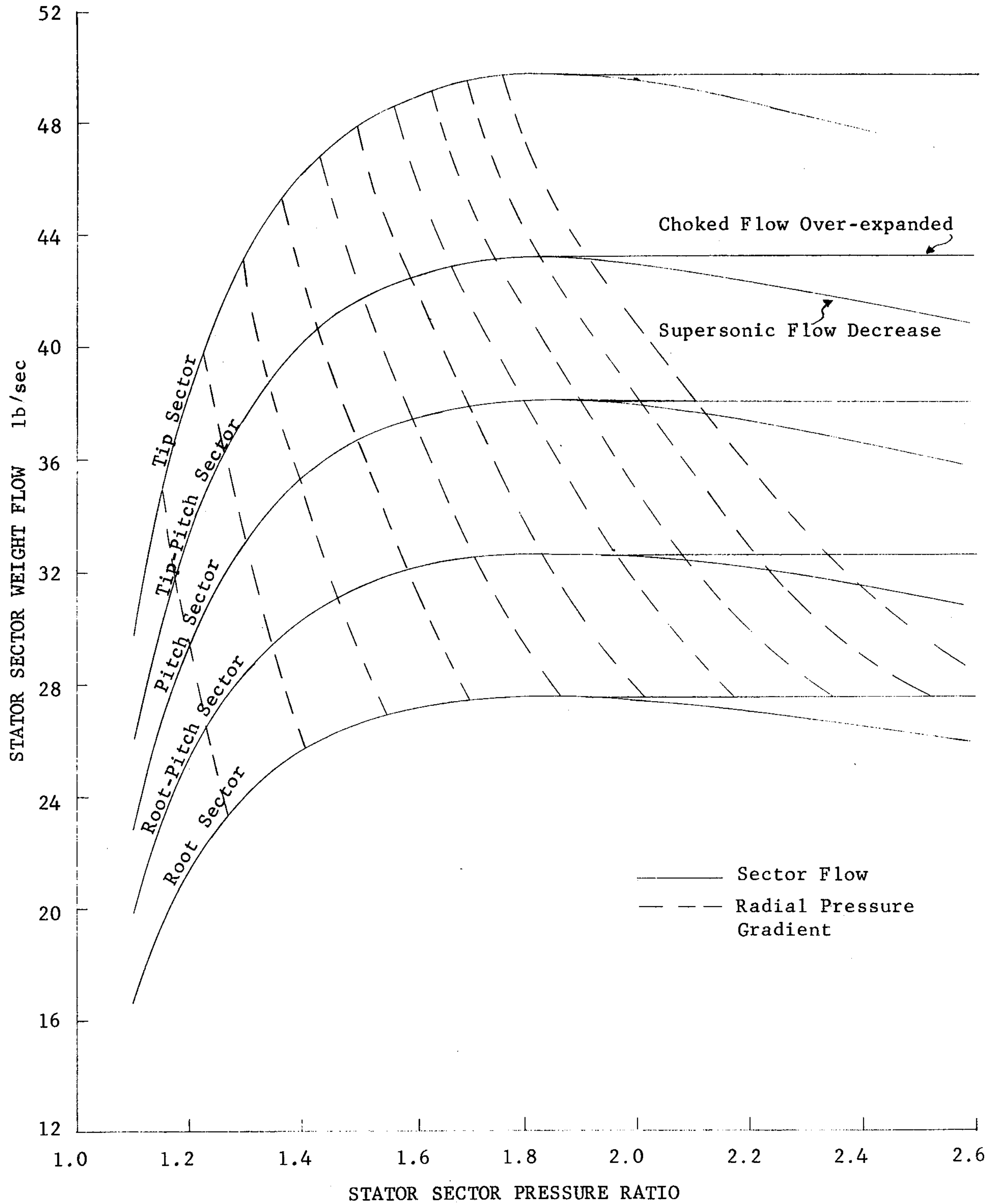


FIGURE 6

TEST CASE 1.

STATOR STATION WEIGHT FLOW VERSUS STATOR PRESSURE RATIO

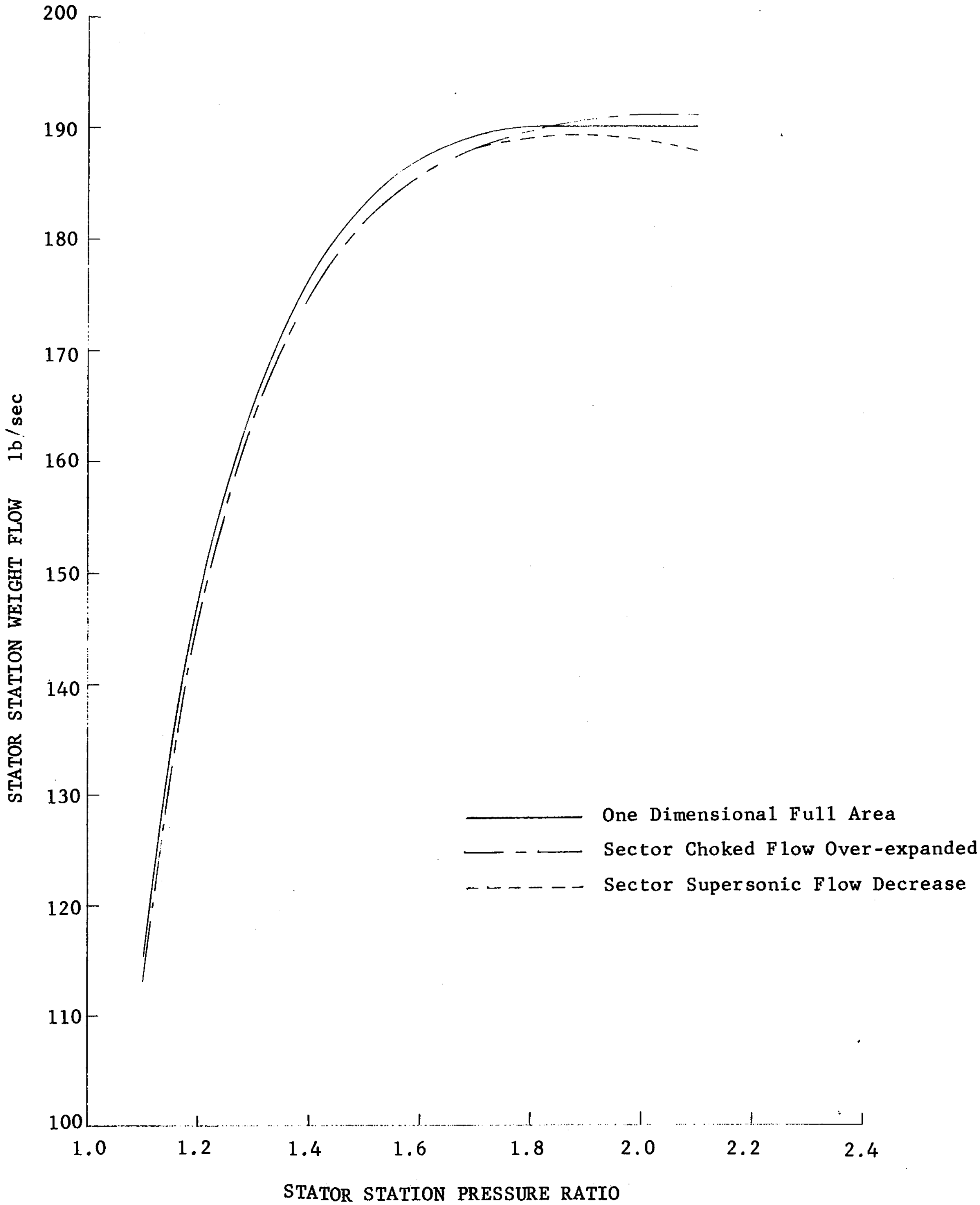


FIGURE 7

TEST CASE 1.

ROTOR SECTOR WEIGHT FLOW VERSUS SECTOR PRESSURE RATIO

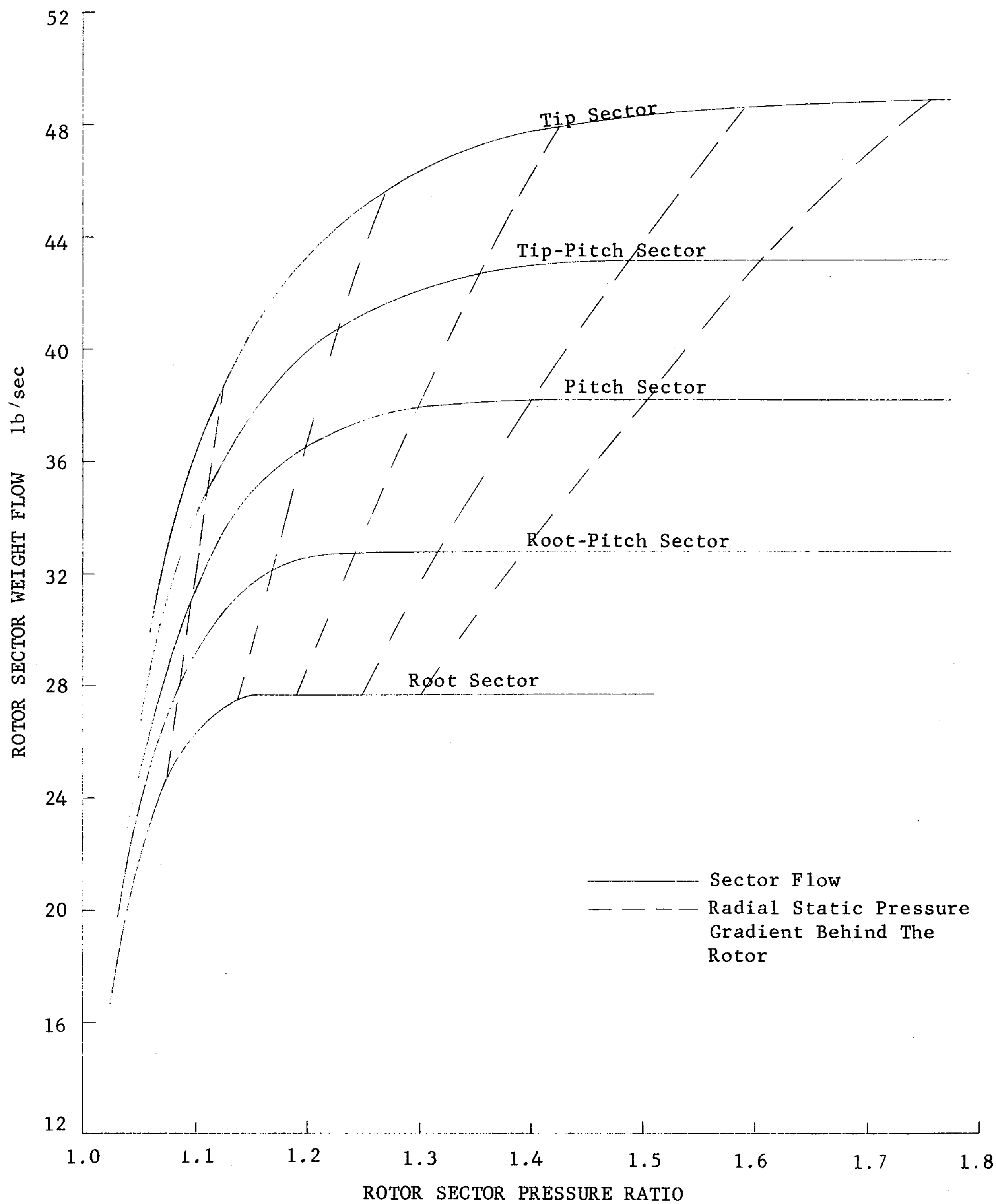


FIGURE 8

TEST CASE 1.

STAGE TOTAL WEIGHT FLOW VERSUS STAGE PRESSURE RATIO

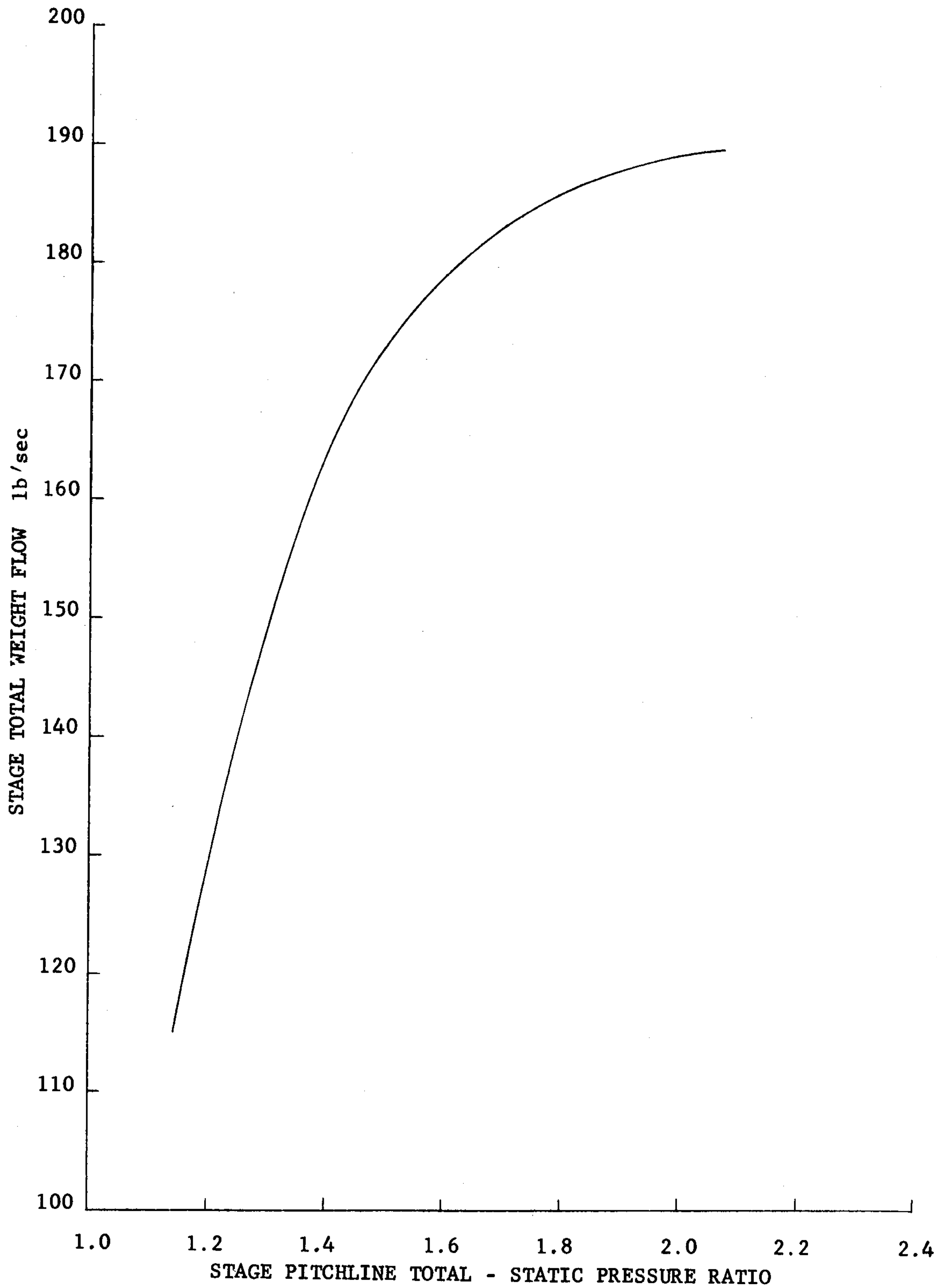


FIGURE 9

NASA TWO STAGE TURBINE

EQUIVALENT ENERGY VERSUS EQUIVALENT FLOW-SPEED PARAMETER

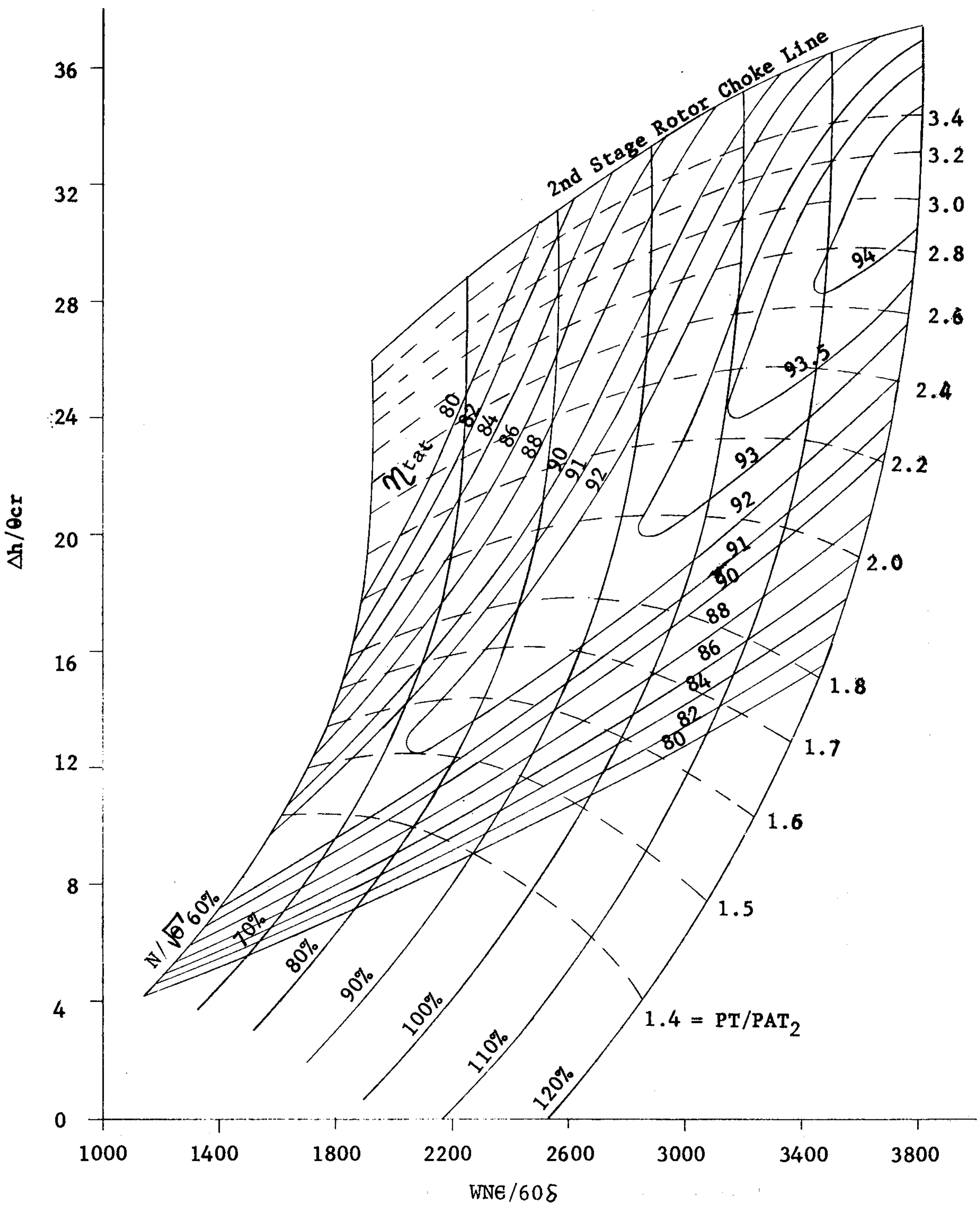
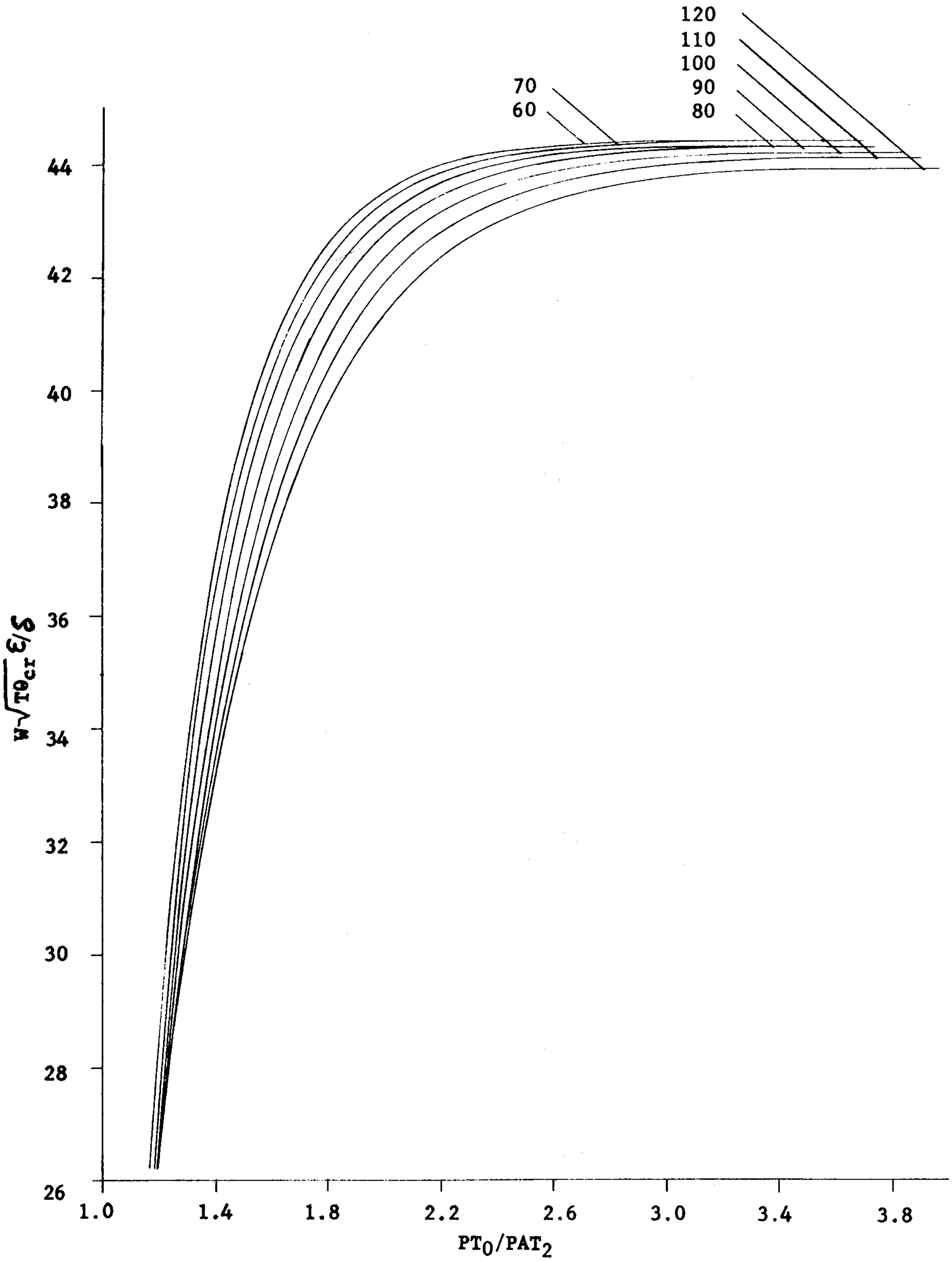


FIGURE 10

NASA TWO-STAGE TURBINE

EQUIVALENT FLOW VERSUS RATING PRESSURE RATIO



APPENDIX 1A.

NTCP - MAIN CALLING PROGRAM

NTCP   Main Program

Turn off sense lights

Set standard values

WAIR	=	0.0
FAIR	=	0.0
PTPS	=	1.02
DELC	=	0.0
DELL	=	0.0
DELA	=	0.0
EXPN	=	2.0
EXPP	=	2.0
EXPRE	=	0.0
RG	=	0.0
PAF	=	0.0
SLI	=	0.0
AACS	=	1.0
SECT	=	1.0
VCTD	=	0.0
WTOL	=	1.E10-4
RHOTOL	=	1.E10-4
PRTOL	=	1.E10-6
PCNH(1)	=	1.0
GAM(1,1)	=	0.0
RWG(1,1)	=	1.0
ETAS(1,1)	=	0.0
ALPHA1(1,1)	=	0.0
ETAR(1,1)	=	0.0
BETA2(1,1)	=	0.0
TRDIAG	=	0.0
TRLOOP	=	0.0

G = 32.17405

AJ = 778.161

Set case number to zero

Program return for next case

1. Set previous error to "false."

CALL INIT

Set point number to zero

Test for previous error "true."

If error occurred, GO TO 1.

If no error occurred

Set

CS(I) = 0. for I = 1,8

CR(I) = 0.

PASS = 0.

2. Set PRPC to CS(KN)

CALL STA01

Set CS(KN) to PRPC

Test for SCRIT 1., set DELPR to DELL

Test for previous error, GO TO 40.

3. CALL STA1A

Test for previous error, GO TO 40.

4. LOPIN = 0

JUMP = 0

Set PRPC = CR(KN)

CALL STA2

Set CR(KN) = PRPC

Test for previous error, GO TO 40.

Test M_{F2} root $\left[1.0 - MF_2 (1,k)\right]$

If MF_2 root $>$ 1.0

Skip to end of case (24)

If MF_2 root $<$ 1.0

5. Test JUMP

If JUMP = 1.0

Skip to set index registers (20)

If JUMP = 0.0

6. CALL STA 2A

Test for previous error, GO TO 40.

Test for last stage

If $KSTG > KN$

7. Increase stage number

$LOPIN = 0.0$

8. $JUMP = 0.0$

set $PRPC = CS(KN)$

CALL STA1

set $CS(KN) = PRPC$

Test for previous error, GO TO 40.

Test JUMP

If $JUMP = 0.0$

GO TO CALL STA 1A (3)

If $JUMP = 1.0$

Skip to set index registers (20)

If $KN = KSTG$

9. CALL OVRALL

Test VCTD

If $VCTD > 0.0$

Calculate interstage vector diagrams

10. CALL INTER

If $VCTD = 0.0$

11. $PASS = 1.0$

Test TRDIAG

If $TRDIAG > 0.0$

12. CALL DIAG

If $TRDIAG = 0.0$

13. Test MFSTOP $[1. - MFSTOP]$

If $MFSTOP > 1.0$

Skip to end of case (24)

If $MFSTOP < 1.0$

14. Test DELC

If $DELC = 0.0$

Skip to end of case (24)

If DELC $>$ 0.0

15. Test DELL

If DELL = 0.0

Skip to test for choke (17)

If DELL $>$ 0.0

16. Test DELPR

If DELPR = 0.0

Skip to end of case (24)

If DELPR $>$ 0.0

Skip to increase CASE number (18)

17. Test CHOKE

If CHOKE = 0.0

18. Increase CASE Number

Increase proper P/P

19. JL = (ISORR-1) * 8 + LSTG

$$Pto/Ps_1(IP,JL) = Pto/Ps_1(IP,JL) + DELPR$$

Set index registers

20. LOPIN = 1

KN = LSTG

IBRC = LBRC

IPC = 0

Test for first stage

If first stage, KN = 1

21. Test for stator or rotor

If stator, ISORR = 1

GO TO CALL STA01 (2)

If rotor, ISORR = 2

GO TO CALL STA2 (4)

All other stages, KN $>$ 1

22. Test for stator or rotor

If Stator, ISORR = 1

GO TO CALL STA1 (8)

If Rotor, ISORR = 2

GO TO CALL STA2 (4)

40. Write previous case has error,

Test for MFSTOP $\equiv$ 2., GO TO next case (24)

GO TO next point (18)

24. Test for last case

If not last case, ENDJOB $\neq$ 1.

GO TO next case (1)

If last case ENDJOB $\equiv$ 1.

GO TO CALL EXIT

23. CALL EXIT.

APPENDIX 1B.

INIT - INITIALIZATION

Subroutine Init

The purpose of subroutine INIT is to check for input errors, skip change cases if the basic case has an input error, test stage loss indicator to store stage one (1) data in following stages, test for equal sectors, set-up sector height, sector pitch diameter, sector annulus area, pitchline wheel speed, define pitchline index, test for angle input to calculate exit angle in radians, inlet angle in radians, and initialize index registers and forks.

3. CALL INPUT

   Increase case number counter

   Test for change case

If change type case, STGCH = 0.0

4. Set error fork = 1

If basic input case, STGCH = 1.0

5. Test for input error

If error occurred, L = 1

6. Write tape 6, case number has an error

   Test for change case

If basic input case, STGCH = 1.0

   Read in next case (3).

If change type case, STGCH = 0.0

   Increase error fork to 2.

   Read in next case (3).

If no error occurred, L = 2

8. Test error fork - 2

If error fork = 2

   Read in next case (3).

If error fork = 1,

   no error

9. Set sector and stage counters

$$ISECT = SECT + .0001$$

$$KSTG = STG + .0001$$

Initialize index registers

$$\begin{aligned} L\Theta PC &= 0 \\ CHOKE &= 0 \\ ICHOKE &= 0 \\ ISORR &= 1 \\ KN &= 1 \\ LSTG &= 1 \\ IBRC &= 1 \\ LBRC &= 1 \\ DELPR &= DELC \\ SC &= 0. \\ RC &= 0. \\ PRPC &= 0. \\ IPC &= 0 \\ ISS &= 0 \\ PTRN &= 0. \end{aligned}$$

Test stage loss indicator, normally zero.

If $SLI = 1.0,$

store stage 1 parameters in following stages

11. $i = 1, ISECT; \; k = 1, KSTG$

$$\begin{aligned} \eta RS_{i,k} &= \eta RS_{i,1} \\ \eta S_{i,k} &= \eta S_{i,1} \\ Cfs_{i,k} &= Cfs_{i,1} \\ \eta RR_{i,k} &= \eta RR_{i,1} \\ \eta R_{i,k} &= \eta R_{i,1} \\ CfR_{i,k} &= CfR_{i,1} \\ TFR_{i,k} &= TFR_{i,1} \end{aligned}$$

12. End of i loop, k loop

13. Test PCNH - 1.0

If $PCNH = 1.0$

14. $i = 1, ISECT$

$$PCNH_i = 1./SECT$$

15. End of i loop

If $PCNH < 1.,$

use input values

16. Set up twice station height

$k = 1, KSTG$

$$Sh_0 = Dt_{0k} - Dr_{0k}$$
$$Sh_1 = Dt_{1k} - Dr_{1k}$$
$$Sh_{1A} = Dt_{1Ak} - Dr_{1Ak}$$
$$Sh_2 = Dt_{2k} - Dr_{2k}$$
$$Sh_{2A} = Dt_{2Ak} - Dr_{2Ak}$$

Set up sector height

$i = 1, ISECT$

$$h_{0i,k} = .5\ PCNHi * Sh_0$$
$$h_{1i,k} = .5\ PCNHi * Sh_1$$
$$h_{1Ai,k} = .5\ PCNHi * Sh_{1A}$$
$$h_{2i,k} = .5\ PCNHi * Sh_2$$
$$h_{2Ai,k} = .5\ PCNHi * Sh_{2A}$$

Test for root sector (i - 1)

Root sector values, $i = 1$

20.

$$Dp_{0i,k} = Dr_{0k} + h_{01,k}$$
$$Dp_{1i,k} = Dr_{1k} + h_{1i,k}$$
$$Dp_{1Ai,k} = Dr_{1Ak} + h_{1Ai,k}$$
$$Dp_{2i,k} = Dr_{2k} + h_{2i,k}$$
$$Dp_{2Ai,k} = Dr_{2Ak} + h_{2A1,k}$$

$$Ann_{01,k} = (\pi/144.) * Dp_{0i,k} * h_{0i,k}$$
$$Ann_{1i,k} = (\pi/144.) * Dp_{1i,k} * h_{1i,k}$$
$$Ann_{1Ai,k} = (\pi/144.) * Dp_{1Ai,k} * h_{1Ai,k}$$
$$Ann_{2i,k} = (\pi/144.) * Dp_{2i,k} * h_{2i,k}$$
$$Ann_{2Ai,k} = (\pi/144.) * Dp_{2Ai,k} * h_{2Ai,k}$$

$$U1Ai,k = \pi * Dp_{1Ai,k} * rpm/720.$$
$$U2i,k = \pi * Dp_{2i,k} * rpm/720.$$

go to end of i loop

If not root sector, $i > 1$

21. Test for more than one sector, ISECT - 1

If more than one sector

17.

$$Dp_{0i,k} = Dp_{0i-1,k} + h_{0i-1,k} + h_{0i,k}$$
$$Dp_{1i,k} = Dp_{1i-1,k} + h_{1i-1,k} + h_{1i,k}$$
$$Dp_{1Ai,k} = Dp_{1Ai-1,k} + h_{1Ai-1,k} + h_{1Ai,k}$$
$$Dp_{2i,k} = Dp_{2i-1,k} + h_{2i-1,k} + h_{2i,k}$$
$$Dp_{2Ai,k} = Dp_{2Ai-1,k} + h_{2Ai-1,k} + h_{2Ai,k}$$

$$Ann_{0i,k} = (\pi/144.) * Dp_{i,k} * h_{0i,k}$$
$$Ann_{1i,k} = (\pi/144.) * Dp_{i,k} * h_{1i,k}$$
$$Ann_{1Ai,k} = (\pi/144.) * Dp_{i,k} * h_{1Ai,k}$$
$$Ann_{2i,k} = (\pi/144.) * Dp_{i,k} * h_{2i,k}$$
$$Ann_{2Ai,k} = (\pi/144.) * Dp_{i,k} * h_{2Ai,k}$$

$$U_{1Ai,k} = \pi * Dp_{1Ai,k} * rpm/720.$$
$$U_{2i,k} = \pi * Dp_{2i,k} * rpm/720.$$

18. End of i loop

19. End of k loop

Set test for odd or even number of sectors

IT = ISECT - 2 * (ISECT/2)

Test IT

If even number of sectors, IT = 0

22. IP = ISECT/2

If odd number of sectors, IT = 1

23. IP = (ISECT + 1)/2

Test for stator exit angle input

If angle not input, SDEA (1,1) = 0.0

Set stator angle fork

25. SDEAF = 0.0

Calculate stator exit angle

i = 1,ISECT;
k = 1,KSTG

$$\cos \alpha_{1i,k} = ndo_{i,k} * Cf_{Si,k} / (h/h_{ths})_k * \pi * Dp_{1i,k} * \sqrt{\eta_{Si,k}}$$

$$\alpha_{1i,k} = \tan^{-1}\left[\sqrt{1_o - (\cos \alpha_{1i,k})^2} / \cos \alpha_{1i,k}\right]$$

26. End of i loop, k loop

If stator angle input, SDEA(1,1) $\neq$ 0.0

27. i = 1,ISECT;
k = 1,KSTG

$$\alpha_{1i,k} = \alpha^o_{1i,k} * \pi/180.$$

$$\cos \alpha_{1i,k} = \cos(\alpha_{1i,k})$$

28. End of i loop, k loop

Test for rotor exit angle input

If angle not input, RDEA(1,1) = 0.0

Set rotor angle fòrk

29. RDEAF = 0.0

Calculate rotor exit angle

i = 1,ISECT;
k = 1,KSTG

$$\cos \beta_{2i,k} = ndo_{Ri,k} * Cf_{Ri,k}/(h/h_{thR})_k * \pi * Dp_{2i,k} * \sqrt{\eta R_{i,k}}$$

$$\beta_{2i,k} = \tan^{-1}\left[\sqrt{1. - (\cos \beta_{2i,k})^2} \Big/ \cos \beta_{2i,k}\right]$$

30. End of i loop, k loop

If rotor angle input, RDEA(1,1) $\neq$ 0.0

32. i = 1,ISECT;
k = 1,KSTG

$$\beta_{2i,k} = \beta^o_{2i,k} * \pi/180.$$

$$\cos \beta_{2i,k} = \cos(\beta_{2i,k})$$

33. End of i loop, k loop

34. i = 1,ISECT;
k = 1,KSTG

Stator inlet conditions

$PTP_{i,k}$ = PTIN
$PTO_{i,k}$ = PTIN
$TTO_{i,k}$ = TTIN
$PTO/PS1_{i,k}$ = PTPS

$$\alpha_{0i,k} = \alpha^{o}_{0i,k} * \pi/180.$$

Rotor inlet angle

$$\beta_{1i,k} = \beta^{o}_{1i,k} * \pi/180.$$

35. End i loop, k loop

Test RG

If RG not input, RG = 0.0

36. CALL R(PTIN,TTIN,FAIR,WAIR,RG)

Set gamma fork

GAMF = 0.0

If RG is input, RG $\neq$ 0.0

Set gamma fork

37. GAMF = 1.0

38. Test for previous error

If error occurred, J = 1.

39. Read in next case (3)

If no error occurred, J = 2.

40. RETURN

APPENDIX 1C

INPUT - READ INPUT

SUBROUTINE INPUT

The purpose of subroutine INPUT is to read in basic input, stage input, assign stage input to its proper location and check for change case input. One block of memory to hold one stage of data is set to BLANKS before each stage is read. After each stage is read, the elements are compared against BLANKS to see if the element is input. If the element is BLANKS the element is set to 0.0 for input printout. If the element is not BLANKS, the proper number stage value is set to the element.

Set up X-array $\begin{pmatrix} i = 6, \\ k = 8, \\ 1 = 37 \end{pmatrix}$

Y-array $\begin{pmatrix} i = 6, \\ 1 = 37 \end{pmatrix}$

where i - number of sector

k - number of stages

1 - number of elements in list

Set X(i = 1, k = 1, 1 = 1) equivalent to $\gamma_{i,k}$ (Common array)

Y(i = 1, 1 = 1) equivalent to γ_o (Input list)

Read the heading cards every time.

10 Read first card, assign to NAME

20 Read second card, assign to TITLE

Set write counter to 0

30 $i = 1,6$; $1 = 1,37$

25 $Y_{i,1}$ = BLANKS

Read file of stage data

40 Set counter for number of stages

50 Set counter for number of sectors

Compare stage data against BLANKS

60 $i = 1,6$; $1 = 1,37$

If Y = BLANKS

Set $Y_{i,1} = 0.$

GO TO end of loop (80)

If $Y \neq$ BLANKS

Set $X_{i,k,1} = Y_{i,1}$

80 End of i loop, 1 loop.

Relocate values

90 $(h/h_{thS})_k = h/h_{thS}$

100 $(h/h_{thR})_k = h/h_{thS}$

110 Test for stage one

If stage one (K = 1)

120 Write basic input data

Increase write counter

All stages

130 Write stage input data

140 Test for loss option

If loss option type case ($\bar{\omega}_S$ is input)

150 Write loss option input data.

160 Increase write counter

Test write counter to restore printer

If the write counter is even (AM = 0)

200 Restore printer

If the write counter is odd (AM = 1)

210 Test for last stage

If not last stage, go to read next stage (30)

170 If last stage,

RETURN.

APPENDIX 1D

STA01 - STATION 0-1

SUBROUTINE STA01

The purpose of subroutine STA01 is to establish the first stator exit flow which is then used as the basis for all stations. Start at pitchline, calculate in to root then out to tip. Set Pto = PTIN for first estimate and calculate the stator flow for each sector at station 1. Adjust flows for cooling air injection between stations 0 and 1. Find inlet Mach number and incidence angle loss at station 0. Since loss may vary with sector number, adjust Pt and get new flow at station 1 for final result.

Set i to pitchline value

Set Δi to -1

Set flow counter

Initialize $\qquad Wgt_{1k} = 0.0$

Test gamma fork (γF)

If gamma not input, $\qquad \gamma F = 0.0$

Starting value of average temperature for γ_1

2. $\qquad T_{a,k} = .95 * T_{toIP,k}$

CALL GAMMA($P_{to1,1}$, T_{a1k}, FAIR, WAIR, γ_{1k})

If gamma input, $\qquad \gamma F = 1.0$

3. CALL FLOW1 (i)

Test previous error

If error GO TO (26)

$$Wgt_{1k} = Wgt_{1k} + Wg_{1ik}$$

Test for tip sector

If not tip sector, $\qquad ISECT > i$

4. Index i by Δi

Test for lower than root sector

If lower than root, $\qquad i = 0$

6. $\qquad \Delta i = 1$

$$i = IP + \Delta i$$

If not lower than root, $\qquad i > 0$

Find index for dp/dr

22. $$\ell = i - \Delta i$$

$$Ps_{1i,k} = Ps_{1l,k} + \text{FLOAT}(\Delta i) * dp/dr_{1l,k} * (h_{1i,k} + h_{1l,k}) / 2.$$

$$Pt_0/Ps_{1i,k} = Pt_{0i,k}/Ps_{1i,k}$$

Test $(Pto/Ps_{1i,k}-1.)$

If $Pt_0/Ps_{1i,k} < 1.$

27. Set $Pt_0/Ps_{1i,k} = 1.0$

$$PTRN = -1.$$

GO TO CALL FLOW1 (3)

If tip sector $i = ISECT$

Calculate STAO for incidence correction.

5. Test flow counter

If incidence correction has been completed,

$$JW = 2$$

GO TO end of calculation (18)

If incidence correction is not complete,

$$JW = 0$$

16. Test gamma fork

If gamma not input, $\gamma F = 0$

7. $$\gamma 0_k = \gamma 1_k$$

$$Wgt0_k = Wgt1_k/Rwg1_k$$

$$i = IP$$

$$Wg0_{i,k} = Wg0_{i,k}/Rwg1_k$$

$$FF/A0_{i,k} = Wg0_{i,k} * \sqrt{Tt0_{i,k}} \Big/ (144. * Pt0_{i,k} * Ann0_{i,k})$$

19. Set counter to improve gamma based on root sector average temperature.

$$j = 1$$

8. CALL PRATIO$(FF/A_{0i,k}, \gamma_{0,k}, RG, PRTOL, Pt_0/Ps_{0i,k})$

$$Ps_{0i,k} = Pt_{0i,k} \Big/ (Pt_0/Ps_{0i,k})$$

$$Tt_0/Ts_{0i,k} = (Pt_0/Ps_{0i,k})^{\frac{\gamma ok - 1.}{\gamma 0k}}$$

$$Ts_{0i,k} = Tt_{0i,k} \Big/ (Tt_0/Ts_{0i,k})$$

9. Test gamma fork

If gamma not input, $\gamma F = 0.0$

10. $$Ta_{0k} = .5 * (Tt_{0i,k} + Ts_{0i,k})$$

CALL GAMMA ($Pt_{01,1}$, Ta_{0k}, FAIR WAIR, γ_{0k})

Test counter to improve gamma

If $j = 1$

11. $j = j + 1$

GO TO CALL PRATIO (8) once to improve gamma

If repeat from CALL PRATIO, $j = 2$

12. $$Cp_{0k} = RG * \gamma_{0k}/(\gamma_{0k} - 1.) * J$$

$$i = 1, ISECT$$

$$Wg_{0i,k} = Wg_{1i,k}/Rwg_{1k}$$

$$PTOMO = Pt_{0i,k}$$

$$FF/A_{0i,k} = Wg_{0i,k} * Tt_{0i,k}/(144. * Pt_{0i,k} * Ann_{0i,k})$$

Test i

If $i = IP,$

GO TO 28

If $i \neq IP$

$$Ps_{0i,k} = Ps_{0IP,k}$$

$$Pt_0/Ps_{0i,k} = Ptp_{i,k}/Ps_{0i,k}$$

28. $$Tt_0/Ts_{0i,k} = (Pt_0/Ps_{0i,k})^{\frac{\gamma 0k - 1.}{\gamma 0_k}}$$

$$Ts_{0i,k} = Tt_{0i,k}/(Tt_0/Ts_0)_{i,k}$$

13. $$V_{0i,k} = \sqrt{.2 * g * J * Cp_{0k} * (Tt_{0i,k} - Ts_{0i,k})}$$

$$As_{0i,k} = \sqrt{\gamma_{0k} * g * RG * Ts_{0i,k}}$$

$$M0_{i,k} = V0_{i,k}/As0_{i,k}$$

$$SI_{i,k} = \alpha 0_{i,k} - \alpha 0^{*}_{i,k}$$

Test $SI_{i,k}$

If SI is negative

24. $EXPS = EXPN$

If SI is positive

20. $EXPS = EXPP$

21.

$$Pt_0/Ps0_{i,k} = \left[1 + \frac{\gamma 0_k - 1}{2} * \eta RS_{i,k} * (M0_{i,k})^2 \cos^{EXPS}(SI_{i,k}) \right]^{\frac{\gamma 0_k}{\gamma 0_{k-1}}}$$

$$Pt0_{i,k} = Ps0_{i,k} * (Pt_0/Ps_0)$$

$$Wg0_{i,k} = Wg0_{i,k} * Pt0_{i,k}/Pt_0MO$$

$$\rho s0_{i,k} = 144. * Ps0_{i,k}/(RG * Ts0_{i,k})$$

$$Vu_{i,k} = V0_{i,k} * \sin \alpha 0_{i,k}$$

$$Wg1_{i,k} = Wg1_{i,k} * Pt0_{i,k}/Pt_0MO$$

$$Vz0_{i,k} = V0_{i,k} * \cos \alpha 0_{i,k}$$

14. End of incidence loss correction loop

$$Wgt1_k = 0.$$

$$i = IP$$

$$\Delta i = -1$$

Set flow counter = 2

15. GO TO CALL FLOW1 3

18. End of calculation

Test for trace of loop

If $TRLOOP = 1.$

Write Wgt_0, Wg_0, Pt_0/Ps_0, Wgt_1, Wg_1, Pt_0/Ps_1.

Test for previous error

If error occurred, $j = 1$

25. CALL DIAGT 1

If no error occurred, $j = 2$

26. RETURN.

APPENDIX 1E

FLOW1 - FLOW 1

SUBROUTINE FLOW1

The purpose of subroutine FLOW1 is to calculate stator exit flow in a sector with two loss branches and supersonic weight flow decrease for supercritical sectors up to pitch sector critical pressure ratio. Sub-critical sectors have weight flow increase after pitch sector critical pressure ratio up to local sector critical pressure ratio. Super-critical sectors have cos αIE correction for constant supersonic weight flow function after the pitch sector is critical; sub-critical sectors have cos αIE correction after local sector critical. When the pitch sector first exceeds the critical pressure ratio, PRPC is set = 1.0, the pitchline pressure ratio is reset to the critical value, PRPC is reset = 2.0, and flow is calculated exactly at the critical pitchline pressure ratio. When the lowest pressure ratio sector exceeds the critical pressure ratio, SCRIT is set = 1.0 as an indicator that maximum flow has been reached.

γ_{1k} based on $TA1_k = .95\ T_{TO_{IP,k}}$ has been set up by STA01.

Compute isentropic stator temperature ratio

7. $$\phi_{1i,k} = (Pt_0/Ps_1)_{i,k}^{\frac{\gamma 1k - 1.}{\gamma 1k}}$$

Test $\bar{\omega}$ for loss coefficient input

If loss input, STPLC $\neq$ 0.0

1. CALL LOSS1

If efficiency input, STPLC = 0.0

Compute exit temperature

2. $$Ts_{1i,k} = Tt_{0i,k} * \left[1. - \eta s_{i,k} * (1. - 1./\phi_{1i,k})\right]$$

Test for pitch sector

If pitch sector i = IP

3. Test for gamma input

If gamma not input, $\gamma F = 0.0$

4. $$Ta_{1k} = .5(Tt_{0IP,k} + Ts_{1IP,k})$$

CALL GAMMA ($Pto_{IP,k}$, $Ta_{1,k}$, FAIR,WAIR,γ_{1k})

If gamma input, $\gamma F = 1.0$

Compute critical pressure ratio

5. $$Pt_o/Ps_{1ck} = (\gamma_{1k} + 1.)/2.^{\frac{\gamma 1k}{\gamma 1k - 1.}}$$

Specific heat at constant pressure

$$Cp_{1k} = RG * \gamma_{1k} / \left[(\gamma_{1k} - 1.) * J\right]$$

All sectors

Exit velocity

6. $$V_{1i,k} = \sqrt{2. * g * J * Cp_{1k} * (Tto_{i,k} - Ts_{1i,k})}$$

Exit pressure

$$Ps1_{i,k} = Pto_{i,k} / (Pto/Ps_1)_{i,k}$$

Exit density

$$\rho s1_{i,k} = 144. * Ps_{1i,k} / RG * Ts_{1i,k}$$

Test for critical pressure ratio

If greater than critical

8. Test for pitch sector

If pitch sector $i = IP$

9. Test PRPC

If previous non-critical, PRPC = 0.0

10. Set PRPC = 1.0

$$Pt0/Ps1_{IP,k} = Pt_0/Ps1c_k * (1. + Prtol)$$

go back to repeat from start $\emptyset 1_{i,k}$ (7)

If not pitch sector, $i \neq IP$

21. Test for pressure ratio less than pitchline pressure ratio

If pressure ratio is less than the pitchline pressure ratio or the pitchline is super-critical

22. Test for $I = 1$ or $I = ISECT$

If $I = 1$ or $I = ISECT$,

set SCRIT = 1.

Pitch supercritical or sector pressure ratio less than pitchline.

11. $$\emptyset 1c_k = (\gamma_{1k} + 1.)/2.$$

$$V1c_{i,k} = \sqrt{2. * g * J * Cp_{1k} * Tt0_{i,k} * \eta s_{i,k} * (\emptyset 1c_k - 1.)/\emptyset 1c_k}$$

$$Ts1c_{i,k} = Tt0_{i,k} * \left[1. - \eta s_{i,k} * (1. - 1./\emptyset 1c_k)\right]$$

$$\rho s1c_{i,k} = 144. * Pt0_{i,k}/(Pt0/Ps1c_k * Ts1c_{i,k} * RG)$$

$$Wg1c_{i,k} = \rho s1c_{i,k} * V1c_{i,k} * Ann1_{i,k} * \cos \alpha 1_{i,k}$$

$$Wg1_{i,k} = Wg1c_{i,k}$$

effective stator exit angle

13. $$\cos \alpha 1E_{i,k} = Wg1_{i,k}/(\rho s1_{i,k} * V1_{i,k} * Ann1_{i,k})$$

14. $$\alpha 1E_{i,k} = \tan^{-1}\left[\sqrt{1. - (\cos \alpha 1E_{i,k})^{2.}}/\cos \alpha 1E_{i,k}\right]$$

Skip to last block (16).

If pressure ratio is greater than pitchline pressure ratio

12. Test for pitchline pressure ratio critical

If pitchline critical, PRPC = 2.

Flow is calculated from constant flow function

24. $$Wg1_{i,k} = FFs_{i,k} * Pt0_{i,k}/\sqrt{Tt0_{i,k}}$$

Go to get effective angle (13)

If pressure ratio is less than critical or supersonic flow decrease

15. $$Wg1_{i,k} = \rho s1_{i,k} * V1_{i,k} * Ann_{i,k} * \cos \alpha 1_{i,k}$$

$$\cos \alpha 1E_{i,k} = \cos \alpha 1_{i,k}$$

$$\alpha 1E_{i,k} = \alpha 1_{i,k}$$

$$FFs_{i,k} = Wg1_{i,k} * \sqrt{Tt0_{i,k}}/Pt0_{i,k}$$

Last block

16. $$Vu1_{i,k} = V1_{i,k} * \sin(\alpha 1E_{i,k})$$

$$dp/dr1_{i,k} = (2./144.) * \rho s1_{i,k} * (Vu1_{i,k})^{2.}/(g * Dp1_{i,k})$$

$$Vz1_{i,k} = V1_{i,k} * \cos \alpha 1E_{i,k}$$

Test for $I < ISECT$

If	I $<$ ISECT,
GO TO CALL CHECK (17)	
If	I = ISECT
Test for	PRPC = 1.
If	PRPC = 1.,
Set	PRPC = 2.

17. CALL CHECK

If error occurred, CALL DIAGT

20. RETURN

APPENDIX 1F

LOSS1 - LOSS 1

<u>SUBROUTINE LOSS1(I,K,EX)</u>

where I - sector number

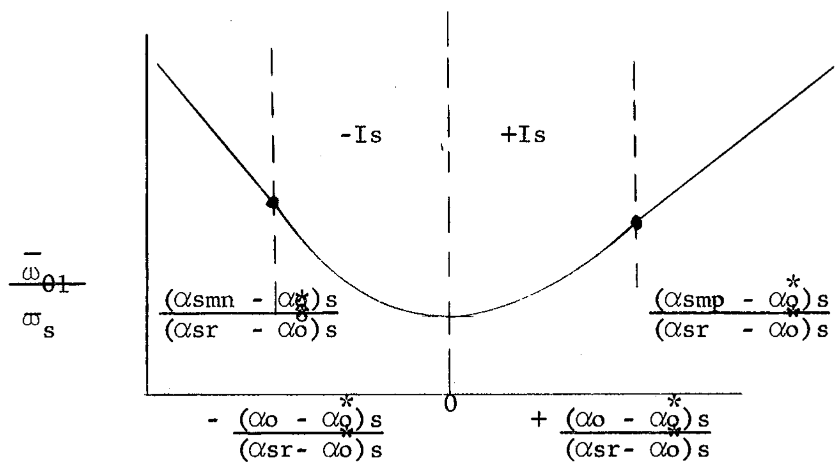
K - stage number

EX - $(\gamma - 1)/\gamma$

The purpose of subroutine LOSS1 is to set $\eta sr = 1.0$ and calculate ηs from a quadratic polynominal

$$\frac{\Delta Pt}{Pt0 - Ps1} = f\ (\bar{\omega}_{optimum},\ is)$$

For positive incidence angle, the polynominal is followed up to αsmp, then a linear multiplier is used for higher incidence angles. For negative incidence angle, the polynominal is followed down to αsmn, then a linear multiplier is used for lower incidence angles.



EXPN = 0.

EXPP = 0.

$\eta sr_{i,k} = 1.0$

Test $Is_{i,k}$

If negative, go to calculation negative quadratic (5)

If positive, go to calculation positive quadratic (2)

If zero

1. $$\bar{\omega}01 = \bar{\omega}s_{i,k}$$

Go to calculation ηs (9)

2. $$AS = A1_{i,k} \quad \text{(square coeff.)}$$
$$Ac = A2_{i,k} \quad \text{(cubic coeff.)}$$
$$AQ = A3_{i,k} \quad \text{(quartic coeff.)}$$

Test $(\alpha smp - \alpha o)_{i,k}$ i.e. $\left[(\alpha smp - \alpha o^*) - (\alpha o - \alpha o^*)\right]$

If negative

3. $$\frac{\bar{\omega}m}{\bar{\omega}s} = Is_{i,k} / \alpha smp_{i,k}$$
$$AR = \frac{(\alpha smp - \alpha o^*)_{i,k}}{(\alpha sr - \alpha o^*)_{i,k}}$$

Go to Calculation $\bar{\omega}01$

If zero

4. $$\frac{\bar{\omega}m}{\bar{\omega}s} = 1.0$$
$$AR = \frac{(\alpha smp - \alpha o^*)_{i,k}}{(\alpha sr - \alpha o^*)_{i,k}}$$

Go to calculation $\bar{\omega}01$ (8)

Negative Quadratic

$$AS = A4_{i,k} \quad \text{(square coeff.)}$$
$$Ac = A5_{i,k} \quad \text{(cubic coeff.)}$$
$$AQ = A6_{i,k} \quad \text{(quartic coeff.)}$$

Test $(\alpha o - \alpha smn)_{i,k}$ i.e. $\left[(\alpha o - \alpha o^*) - (\alpha smn - \alpha o^*)\right]$

If negative

6. $$\frac{\bar{\omega}m}{\bar{\omega}s} = \frac{Is_{i,k}}{(\alpha smn - \alpha o^*)_{i,k}}$$

$$AR = (\alpha smn - \alpha o^*)_{i,k} / (\alpha sr - \alpha o^*)_{i,k}$$

If positive,go to (4)

8. $$\bar{\omega}o_1 = \bar{\omega}s_{i,k} \left[1. + (AR)^2 \left\{AS + AR * (AC + AR * AQ)\right\}\right] * \left(\frac{\bar{\omega}m}{\bar{\omega}s}\right)$$

9. $$\eta s_{i,k} = \frac{1. - \left\{\dfrac{1}{(Pto/Ps1)_{i,k} * (1. - \bar{\omega}o_1) + \bar{\omega}o_1}\right\}^{\frac{\gamma 1_k - 1.}{\gamma 1_k}}}{\left(\emptyset 1_{i,k} - 1.\right) / \emptyset 1_{i,k}}$$

RETURN

APPENDIX 1G

R - GAS CONSTANT

Simple routine for the gas constant for the mixture of JP-4 fuel and/or water in standard air.

SUBROUTINE R(P,T,F,W,RX)

$$RX = 53.35045 + \frac{.658(F) + 32.433(W)}{1. + (F) + (W)}$$

RETURN

where P - pressure
T - temperature
F - fuel/air ratio
W - water/air ratio
RX - gas constant

APPENDIX 1H.

GAMMA - SPECIFIC HEAT RATIO

Simple routine for the specific heat ratio for the mixture of JP-4 fuel and/or water in air.

```
SUBROUTINE    GAMMA(P,T,F,W,GAMX)

   CALL  CPA(P,T,F,W,CPAX)
   Test  (F)
   If positive
   1. CALL CPF(PX,TX,FX,WX,CPFX)
   GO TO  2
   If negative or zero
   2. Test  (W)
   If positive
   3. CALL CPW(P,T,F,W,CPWX)
   GO TO  4
   If negative or zero
   4.
```

$$CPGX = \frac{CPAX + (F)(CPFX) + (W)(CPWX)}{1. + (F) + (W)}$$

```
   CALL  R(P,T,F,W,RX)
```

$$GAMX = \frac{CPGX}{CPGX - (RX/778.161)}$$

```
   RETURN
```

Note: The value of RX and CPGX as calculated in this subroutine GAMG are only used to obtain the best value of GAMX and should not get back to the main program or station sub-programs.

The sub-programs calculate CPG based on GAMX and a constant value of RG.

APPENDIX 1I.

CPA - CONSTANT PRESSURE SPECIFIC HEAT, AIR

Simple 7th order polynominal of specific heat at constant pressure for air at low pressure.

```
SUBROUTINE    CPA(P,T,F,W,CPAX)

   Test  (T - 100.)
   If negative
   1.                           TX = 100.
   GO TO  5
   If zero or positive
   2. Test  (6400. - T)
   If negative
   3.                           TX = 6400.
   4. GO TO  5
                                TX = T
   5.                      XT1  =  TX/1000
                           XT2  =  XT1 * XT1
                           XT3  =  XT1 * XT2
                           XT4  =  XT1 * XT3
                           XT5  =  XT1 * XT4
                           XT6  =  XT1 * XT5
                           XT7  =  XT1 * XT6

                         A0  =   2.4264907 E-01
                         A1  =  -2.6657395 E-02
                         A2  =   4.6617756 E-02
                         A3  =  -1.3546542 E-02
                         A4  =  -8.4500931 E-04
                         A5  =   1.0303393 E-03
                         A6  =  -1.7159795 E-04
                         A7  =   9.1627911 E-06

               CPAX = A0 + A1(XT1) + A2(XT2) + A3(XT3) +
                  A4(XT4) + A5(XT5) + A6(XT6) + A7(XT7)

   RETURN
```

APPENDIX 1J.

CPF - CONSTANT PRESSURE SPECIFIC HEAT, FUEL

Simple 7th order polynominal of equivalent specific heat at constant pressure for burned JP-4 fuel.

```
SUBROUTINE    CPF(P,T,F,W,CPFX)

   Test  (T - 400.)
   If negative

   1.                              TX = 400.
   GO TO  5
   If zero or positive

   2. Test  (3000. - T)
   If negative

   3.                              TX = 3000.
   GO TO  5

   4.                               TX = T

   5.                          XT1  =  TX/1000.
                               XT2  =  XT1 * XT1
                               XT3  =  XT1 * XT2
                               XT4  =  XT1 * XT3
                               XT5  =  XT1 * XT4
                               XT6  =  XT1 * XT5
   6.                          XT7  =  XT1 * XT6

                            A0  =   1.0625243 E-01
                            A1  =   9.5291284 E-01
                            A2  =  -7.2605169 E-01
                            A3  =   2.4481406 E-01
                            A4  =   5.3332162 E-02
                            A5  =  -6.4699814 E-02
                            A6  =   1.7495567 E-02
                            A7  =  -1.6029820 E-03

                   CPFX = A0 + A1(XT1) + A2(XT2) + A3(XT3) +
                       A4(XT4) + A5(XT5) + A6(XT6) + A7(XT7)

   RETURN
```

APPENDIX 1K.

CPW - CONSTANT PRESSURE SPECIFIC HEAT, WATER

Simple 7th order polynominal of specific heat at constant pressure for water vapor at low pressure.

```
SUBROUTINE    CPW(P,T,F,W,CPWX)

   Test  (T - 400.)
   If negative
   1.                              TX = 400.
   GO TO  5
   If zero or positive
   2. Test  (3000. - T)
   If negative
   3.                              TX = 3000.
   GO TO  5
   4.                              TX = T
   5.                         XT1  =  TX/1000.
                              XT2  =  XT1 * XT1
                              XT3  =  XT1 * XT2
                              XT4  =  XT1 * XT3
                              XT5  =  XT1 * XT4
                              XT6  =  XT1 * XT5
   6.                         XT7  =  XT1 * XT6

                            A0  =   4.5728850 E-01
                            A1  =  -9.7007556 E-02
                            A2  =   1.6536409 E-01
                            A3  =  -4.1138066 E-02
                            A4  =  -2.6979575 E-02
                            A5  =   2.2619243 E-02
                            A6  =  -6.2706207 E-03
                            A7  =   6.2246710 E-04

                 CPWX = A0 + A1(XT1) + A2(XT2) + A3(XT3) +
                    A4(XT4) + A5(XT5) + A6(XT6) + A7(XT7)

   RETURN
```

APPENDIX 1L.

PRATIO - PRESSURE RATIO

Purpose is to find a pressure ratio P_{T1}/P_{S1} or P_{TR2}/P_{S2} which are consistent with the respective flow functions $(W\sqrt{T_T}/P_T A)_1$ or $W\sqrt{T_{TR}}/P_{TR}A)_2$. Since the equation is trancendental, a successive iteration is employed. PRATIO is only called by the pitchline sector to start a continuity iteration and therefore only the sub-sonic case is of interest.

SUBROUTINE PRATIO(TFF,GAMX,RX,PTPS,PRTOL)

where TFF - flow function

GAMX - specific heat ratio

RX - gas constant

PTPS - pressure ratio

PRTOL - tolerance

$$PCRIT = \left[(GAMX + 1.)/2.\right]^{\frac{GAMX}{GAMX-1.}}$$

$$PUP = PCRIT$$

$$PLOW = 1.0$$

$$PTRMO = 0.0$$

$$G = 32.17405$$

1. $PTR = (PUP + PLOW)/2.0$

$$DELFM = \sqrt{(PTR)^{-\frac{2}{GAMX}} - (PTR)^{-\frac{GAMX + 1}{GAMX}}} - TFF\sqrt{\frac{RX}{2.G}\ \frac{GAMX\ -1.}{GAMX}}$$

Test (DELFM)

If negative

2. $PLOW = PTR$

GO TO 4

If positive or zero

3. $PUP = PTR$

4. PRE = (PTR - PTRMO)/PTR

Test/PRE/-PRTOL

If positive

5. PTRMO = PTR

GO TO 1

If negative or zero

6. Test (PCRIT - PTR)

If negative

7. PTPS = PCRIT

GO TO 9

If zero or positive

8. PTPSX = PTR

RETURN

APPENDIX 1M.

CHECK - CHECK SENSE LIGHTS

```
SUBROUTINE    CHECK (J)
                    where J = 1   PREVER = .TRUE.
                          J = 2   Normal return

 Test sense lights  I = 1,4
 If light was off

 1.                            J = 2
 RETURN
 If light was on

 2.                            J = 1
                       PREVER = .TRUE.

 RETURN
```

APPENDIX 1N

STA1A - STATION 1A

SUBROUTINE STA1A

The purpose of subroutine STA1A is to determine inlet flow conditions relative to the rotor and find incidence angle recovery on all rotor inlet stations. Start at pitchline to obtain gas properties for the station, calculate in to root then out to tip. The tangential component of absolute velocity is adjusted for diameter change to conserve angular momentum. The axial component of velocity is adjusted for weight flow change, area change, and density change from STA1. The radial gradient in pressure does not satisfy radial equilibrium, and continuity is conserved in each sector.

Set i to pitchline value

Set Δi to -1

Set first guess of $\quad Ts_{1A} = Ts1_{i,k}$

Find ratio of flow change

$$Wr = RWG1A_k / RWG1_k$$

Set total station flow

$$Wgt_{1A_k} = Wr * Wgt_{1_k}$$

Adjust tangential velocity

13. $$Vu_{1A_{i,k}} = Vu_{1_{i,k}} * Dp_{1_{i,k}} / Dp_{1A_{i,k}}$$

Adjust flow,

$$Wg_{1A_{i,k}} = Wr * Wg_{1_{i,k}}$$

Starting density

$$\rho str = \rho s_{1_{i,k}}$$

Adjust axial velocity

1. $$Vz_{1A_{i,k}} = Wr * Vz_{1_{i,k}} * \rho s_{1_{i,k}} * Ann_{1_{i,k}} / (\rho str * Ann_{1A_{i,k}})$$

$$V1A = \sqrt{(Vu_{1A_{i,k}})^2 + (Vz_{1A_{i,k}})^2}$$

Test for pitch sector

If pitch sector $\quad i = IP$

3. Test gamma fork, γF

If gamma not input, $\gamma F = 0.0$

12. $$Ta_{1A} = .5 * (Tto_{i,k} + Ts_{1A})$$

CALL GAMMA($Pto_{i,k}$, Ta_{1A}, FAIR,WAIR, γ_{1A_k})

If gamma input, GAMF = 1.0

4. $$Cp_{1A_k} = RG * \gamma_{1A_k} / \left[(\gamma_{1A_k} - 1.) * J \right]$$

$$\Delta Ts = \left[(V_{1_{i,k}})^{2.} - (V_{1A})^{2} \right] / (2. * g * J * Cp_{1A_k})$$

$$Ts_{1A} = Ts_{1_{i,k}} + \Delta Ts$$

$$Ps_{1A_{i,k}} = Ps_{1_{i,k}} * (1. + \Delta Ts / Ts_{1_{i,k}})^{\frac{\gamma_{1A_k}}{\gamma_{1A_{i,k}} - 1.}}$$

$$\rho s_{1A} = 144. * Ps_{1A_{i,k}} / (RG * Ts)$$

Find density error

$$\rho e = (\rho s_{1A} - \rho str) / \rho s_{1A}$$

Test $(|\rho e| - \rho tol)$

If error $>$ tol

5. $$\rho str = \rho s_{1A}$$

Go to calculation new Vz_{1A}

If error $<$ tol

6. $$Ru_{1A_{i,k}} = Vu_{1A_{i,k}} - U_{1A_{i,k}}$$

$$R_{1A_{i,k}} = \sqrt{(Ru_{1A_{i,k}})^{2.} + (Vz_{1A_{i,k}})^{2.}}$$

$$\sin\beta_{1A} = Ru_{1A_{i,k}} / R_{1A_{i,k}}$$

$$\beta_{1A_{i,k}} = \tan^{-1} \left[\sin\beta_{1A} / \sqrt{1. - \sin^2\beta_{1A}} \right]$$

Test $\bar{\omega} r$

If $\bar{\omega} r$ is input

7. $$\eta rr_{i,k} = 1.0$$

$$EXPR = 0.0$$

8. $$Mr_{1A_{i,k}} = R_{1A_{i,k}} / \sqrt{\gamma 1A_k * g * RG * Ts_{1A}}$$

$$Tr/Ts_{1A} = 1. + (Mr_{1A_{i,k}})^2 * (\gamma 1A_k - 1.)/2.$$

32. $$Ttr_{1A_{i,k}} = Ts_{1A} * (Tr/Ts_{1A})$$

$$RI_{i,k} = \beta 1A_{i,k} - \beta^*1A_{i,k}$$

Test RI,

If $RI > \pi/2$,

Set $RI = \pi/2$;

If $RI < -\pi/2$,

Set $RI = -\pi/2$

If RI is negative

9. $$EXPR = EXPN$$

If RI is positive

10. $$EXPR = EXPP$$

11. $$Pr/Ps_{1A} = \left[1. + (Tr/Ts_{1A} - 1.) * \eta rr_{i,k} * \cos^{EXPR}(RI_{i,k})\right]^{\frac{\gamma 1A_k}{\gamma 1A_k - 1.}}$$

$$Ptr_{1A_{i,k}} = Ps_{1A_{i,k}} * (Pr/Ps_{1A})$$

Test for tip sector

If not tip sector

14. $$i = i + \Delta i$$

Test for root sector

If root sector

15. $$\Delta i = 1$$

$$i = IP + \Delta i$$

If i positive

Go to next sector (13)

Tip sector

16. Test for previous error

If error has occurred

17. CALL DIAGT

No error

18. RETURN

APPENDIX 10

STA2 - STATION 2

SUBROUTINE STA2

The purpose of subroutine STA2 is to satisfy continuity of flow and radial equilibrium at a rotor exit station. Relative total temperature and pressure are adjusted for the radial change in sector diameter from rotor inlet to rotor exit. Flow is adjusted for cooling air injection between stations 1A and 2. The starting value of pitchline pressure ratio is obtained from a one-dimensional flow function per unit area relationship. Incidence angle loss in total pressure will vary with pressure ratio and sector number. Simple radial equilibrium is used to balance the radial distribution of flow.

Set index for critical condition = 0.0

Set index for flow loop = 1

Set index for pressure ratio limit = 0

$$WR = RWG_{2_k}/RWG_{1A_k}$$

$$i = 1, ISECT$$

$$Ttr2_{i,k} = Ttr_{1A_{i,k}} + (U_{2_{i,k}}^{2.} - U_{1_{i,k}}^{2.})/(2 * g * J * Cp_{1A_k})$$

$$Ptr2_{i,k} = Ptr_{1A_{i,k}} * (Ttr_{2_{i,k}}/Ttr_{1A_{i,k}})^{\frac{\gamma 1A}{\gamma 1A - 1.}}$$

1. $$Wg_{2_{i,k}} = WR * Wg_{1A_{i,k}}$$

End of i loop

$$Wgt_2 = WR * Wgt_{1A_{i,k}}$$

Set

$$i = IP$$

$$\Delta i = -1$$

$$Wgt_{2c} = 0.0$$

Test for choke iteration

If choke iteration, $ICHOKE \neq 0$,

GO TO (3)

If not choke iteration

26. Test loop iteration

If loop iteration, $LOPIN \neq 0$,

Go to (3)

27. Test γF

If gamma not input, $\gamma F = 0.0$

2. $Ta_{2_k} = .95 * Ttr_{2_{i,k}}$

CALL GAMMA($Ptr_{2_{i,k}}$, Ta_{2_k}, FAIR, WAIR, $\gamma 2_k$)

If gamma input, $\gamma_F = 1.0$

16. $FF/A2_{i,k} = Wg2_{i,k} * \sqrt{Ttr_{2_{i,k}}} / (144. * Ptr_{2_{i,k}} * Ann_{2_{i,k}} * \cos\beta_{2_{i,k}})$

CALL PRATIO($FF/_{A2_{i,k}}$, $\gamma 2_k$, RG, PRTOL, $Ptr/Ps_{2_{i,k}}$)

3. CALL FLOW2

Test for previous error

If error occurred, Go to (22)

If no error occurred

$$Wgt_{2c_k} = Wgt_{2c_k} + Wg_{2_{i,k}}$$

$$L = 1$$

Test for pressure ratio less than pitchline

If less, set $L = 1$

Test for tip sector

If not tip sector, $i < ISECT$

4. $i = 1 + \Delta i$

Test for lower than root sector

If lower than root, $i = 0$

5. $\Delta i = 1$

$$i = IP + \Delta i$$

If not lower than root, $i > 0$

Find index for dp/dr

6. $1 = i - \Delta i$

$$Ps_{2_{i,k}} = Ps_{2_{1,k}} + FLOAT(\Delta i) * dp/dr_{2_{1,k}} * (h_{2_{i,k}} + h_{2_{1,k}})/2.$$

$$Ptr/Ps_{2_{i,k}} = Ptr_{2_{i,k}}/Ps_{2_{i,k}}$$

Test for pressure ratio less than unity.

If $Ptr/Ps_2 < 1.$

19. $Ptrs_{2_{i,k}} = 1. + Prtol$

GO TO CALL FLOW2 (3)

If $Ptr/Ps_2 > 1.0$

GO TO CALL FLOW2 (3)

If tip sector, $i = ISECT$

Test index for pressure ratio limit

If first time past, counter = 0

8. Set index for pressure ratio limit = 1

$$Pr_{crit} = \left[(\gamma 2_k + 1.)/2\right]^{\frac{\gamma 2k}{\gamma 2_k - 1.}}$$

$$Pr_{up} = Ptr_{2_{IP,k}} * Pr_{crit} * Ps_{2_{1,k}}/(Ptr_{2_{1,k}} * Ps_{2_{IP,k}}) * (1. + PRTOL)$$

$$Pr_{low} = 1.0$$

On pressure ratio iteration, counter > 0

9. Set index to value plus 1.

10. Set counter for blade row

Test for choke iteration on this blade row.

If this blade row is on a choke iteration,set

$$Ptrs_{2_{IP,k}} = Pr_{up}.$$

If $Wgt_{2c} < Wgt_2$

11. $Pr_{low} = Ptr/Ps_{2_{IP,k}}$

If $Wgt_{2c} > Wgt_2$

12. $Pr_{up} = Ptr/Ps_{2_{IP,k}}$

Set flow error

13. $WE = 1. - Wgt_{2_k}/Wgt_{2c_k}$

Increase flow loop index

Test (index - 32)

Fails to converge in 32 passes

GO TO WRITE ERROR (18)

Converges on continuity

29. Test for choke iteration on this blade row

If this blade row is on a chóke iteration,

$$ICHOKE = L.$$

31. Set $SCRIT = -WE$

GO TO TEST TRLOOP (15)

30. Test for loop iteration

If loop iteration, $LOPIN = 1$,

Go to (15)

If not on loop iteration

14. $Pre = (Ptrs_{2_{IP,k}} - Ptrmo) / Ptrs_{2_{IP,k}}$

Test $|Pre|$ -Prtol

If pressure ratio error is less than tolerance

Test $|We|$ -Wtol

If flow error is less than tolerance, GO TO TEST TRLOOP (15)

If flow error is greater than tolerance, set

$SCRIT = 1.0$ (23)

If pressure ratio error is greater than tolerance.

24. $Ptrmo = Ptrs_{2_{IP,k}}$

$Wgt_{2c_k} = 0.0$

$i = IP$

$\Delta i = -1$

Test SCRIT

If $SCRIT = 1.$,

GO TO TEST TRLOOP (15)

If $SCRIT = 0.$

28. $Ptrs_{2_{IP,k}} = .5 * (Pr_{up} + Pr_{low})$

Test $(Ptrs_{2_{i,k}} - Pr_{crit})$

If $Ptrs_{2_{IP,k}} < Pr_{crit}$,

set $PRPC = 0.$

GO TO CALL FLOW2 (3)

23. $SCRIT = 1.$

15. Test for TRLOOP

If $TRLOOP \neq 0.$

WRITE: Prup, Prlow, We, Prcrit, j, Wgt_{2_k}, Wgt_{2c_k}, $Wg_{2_{i,k}}$, $Ptrs_{2_{i,k}}$

If $TRLOOP = 0.$

25. Test for previous error

If error has occurred

20. CALL DIAGT

If no error has occurred

21. CALL LOOP

22. RETURN

FLOW2 - FLOW 2

SUBROUTINE FLOW2

The purpose of subroutine FLOW2 is to calculate rotor exit flow in a sector with two loss branches and supersonic weight flow decrease in supersonic sectors up to pitch sector critical pressure ratio. Subsonic sectors have weight flow increase after the pitch sector critical pressure ratio up to local sector critical pressure ratio. Supersonic sectors have cos β_{2E} correction for constant weight flow function after the pitch sector is critical; subsonic sectors have cos β_{2E} correction after local sector critical. When the pitch sector first exceeds the critical pressure ratio, PRPC is set = 1.0, the pitchline pressure ratio is reset to the critical value, PRPC is reset = 2.0, and flow is calculated exactly at the critical pitchline pressure ratio. When the last subsonic sector exceeds the critical pressure ratio, SCRIT is set = 1.0 as an indicator that maximum flow has been reached.

γ_{2k} based on Ta_{2k} = .95 $Ttr_{2_{IP,k}}$ has been set up by STA2.

Compute isentropic rotor relative temperature ratio

10. $\emptyset_{2_{i,k}} = (Ptr/Ps_{2_{i,k}})^{\frac{\gamma_{2k} - 1.}{\gamma_{2k}}}$

Test $\bar{\omega}$r for loss coefficient input

If $\bar{\omega}$ is input RTPLC > 0.0

1. CALL LOSS2

If efficiency input, RTPLC = 0.0

Compute exit temperature

2. $$Ts_{2_{i,k}} = Ttr_{2_{i,k}} \left[1. - \eta r_{i,k}\,(1. - 1./\emptyset_{2_{i,k}})\right]$$

Test for pitch sector

If pitch sector, i = IP

3. Test for gamma input

If gamma not input, γ_F = 0.0

4. $$Ta_{2_k} = .5 * (Ttr_{2_{i,k}} + Ts_{2_{i,k}})$$

CALL GAMMA ($Ptr_{2_{i,k}}$, Ta_{2_k}, FAIR,WAIR,γ_{2k})

If gamma input, γ_F = 1.0

Compute critical pressure ratio

5. $$Pr/Ps_2\Big|_{crit_k} = \left[(\gamma_{2k} + 1.)/2\right]^{\frac{\gamma 2k}{\gamma 2k - 1.}}$$

Specific heat at constant pressure

6. $$Cp_{2k} = RG * \gamma_{2k} / \left[(\gamma_{2k} - 1.) * J\right]$$

All sectors

Relative exit velocity

$$R_{2_{i,k}} = \sqrt{2 * g * J * Cp_{2k} * (Ttr_{2_{i,k}} - Ts_{2_{i,k}})}$$

Exit pressure

$$Ps_{2_{i,k}} = Ptr_{2_{i,k}} / (Ptr/Ps_{2_{i,k}})$$

Exit density

$$\rho_{s2_{i,k}} = 144. * Ps_{2_{i,k}} / RG * Ts_{2_{i,k}}$$

Test for critical pressure ratio

If greater than critical

7. Test for pitch sector

If pitch sector

8. Test for previous critical pitch sector

If no previous pitch critical occurred,

$$PRPC = 0.0$$

9. $$PRPC = 1.0$$

$$Ptr/Ps_{2_{i,k}} = Pr/Ps_{2crit\ k}$$

go back to repeat from start $\phi_{2_{i,k}}$ (10)

22. Test for pressure ratio less than pitchline

If greater, go to supersonic flow decrease (13)

If pressure ratio less than pitchline

18. Test for $I = 1,$

or $I = ISECT$

If $I = 1$

or $I = ISECT,$

set $SCRIT = 1.$

11. $\phi_{2c_k} = (\gamma_{2k} + 1.)/2$ $\phi_{2c_k} = (\gamma_{2_k} + 1.)/2$

$$R_{2c_{i,k}} = \sqrt{2 * g * J * Cp_{2_k} * Ttr_{2_{i,k}} * \eta r_{i,k} * (\phi_{2c_k} - 1.)/\phi_{2c_k}}$$

$$Ts_{2c_{i,k}} = Ttr_{2_{i,k}} * \left[1. - \eta r_{i,k} * (1. - 1./\phi_{2c_k}\right]$$

$$\rho s_{2c_{i,k}} = 144. * Ptr_{2_{i,k}} / \left[RG * (Pr/Ps_{2\ c_k}) * Ts_{2c_{i,k}} \right]$$

$$Wg_{2c_{i,k}} = s_{2c_{i,k}} * R_{2c_{i,k}} * Ann_{2_{i,k}} * \cos \beta_{2_{i,k}}$$

$$Wg_{2_{i,k}} = Wg_{2c_{i,k}}$$

Skip to calc cos β_{2E}, β_{2E} (14)

If pressure ratio greater than pitchline

13. Test (PRPC - 1.)

If $\quad PRPC = 2.$

24. $$Wg_{2_{i,k}} = FFr_{i,k} * Ptr_{2_{i,k}} / \sqrt{Ttr_{2_{i,k}}}$$

Overexpansion after supersonic flow decrease

14. $$\cos \beta_{2E_{i,k}} = Wg_{2_{i,k}} / (\rho s_{2_{i,k}} * R_{2_{i,k}} * Ann_{2_{i,k}})$$

$$\beta_{2E_{i,k}} = \tan^{-1} \left[1. - (\cos \beta_{2E_{i,k}})^{2.} \right] / \cos \beta_{2E_{i,k}}$$

Skip to last block (16)

If pressure ratio less than critical, or

If supersonic flow decrease

15. $$Wg_{2_{i,k}} = \rho s_{2_{i,k}} * R_{2_{i,k}} * Ann_{2_{i,k}} * \cos \beta_{2_{i,k}}$$

$$\cos \beta_{2E_{i,k}} = \cos \beta_{2_{i,k}}$$

$$\beta_{2E_{i,k}} = \beta_{2_{i,k}}$$

$$FFr_{i,k} = Wg_{2_{i,k}} * \sqrt{Ttr_{2_{i,k}}} / Ptr_{2_{i,k}}$$

Last block

16. $$R\mu_{2_{i,k}} = R_{2_{i,k}} * \sin(\beta_{2E_{i,k}})$$

$$V\mu_{2_{i,k}} = R\mu_{2_{i,k}} - V_{2_{i,k}}$$

$$dp/dr_{2_{i,k}} = (2./144.) * \rho s_{2_{i,k}} * (V\mu_{2_{i,k}})^2 / (g * Dp_{2_{i,k}})$$

$$Vz_{2_{i,k}} = R_{2_{i,k}} * \cos \beta_{2E_{i,k}}$$

$$As_{2_{i,k}} = \sqrt{\gamma_{2k} * g * RG * Ts_{2_{i,k}}}$$

$$V_{2_{i,k}} = \sqrt{(Vz_{2_{i,k}})^{2.} + (Vu_{2_{i,k}})^{2.}}$$

$$M_{2_{i,k}} = V_{2_{i,k}} / As_{2_{i,k}}$$

$$Mr_{2_{i,k}} = R_{2_{i,k}} / As_{2_{i,k}}$$

$$Mf_{2_{i,k}} = Mr_{2_{i,k}} * \cos \beta_{2E_{i,k}}$$

If $I < ISECT$,

Go to text for error (17)

If PRPC = 1.,

set PRPC = 2.

17. Test for previous error

19. Error occurred, CALL DIAGT

21. No error,

RETURN

APPENDIX 1Q

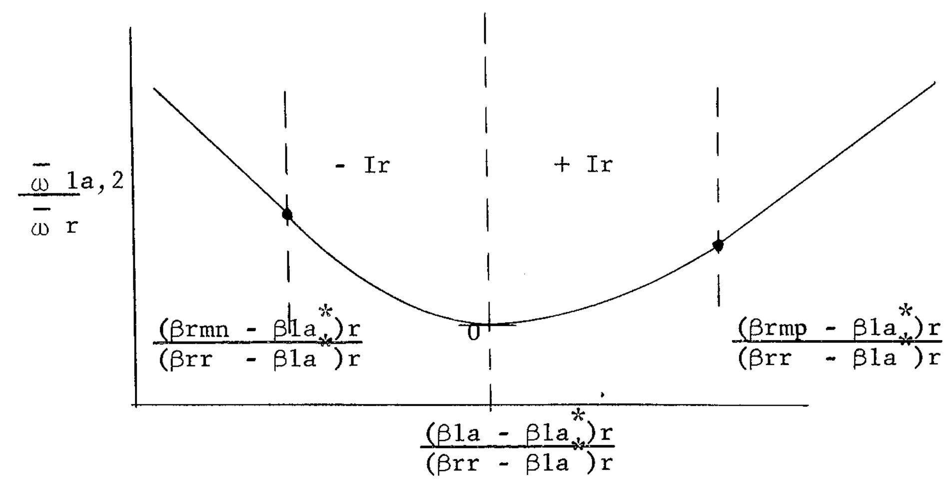
LOSS2 - LOSS 2

SUBROUTINE LOSS2

The purpose of subroutine LOSS2 is to set $\eta rr = 1.0$ and calculate ηr from a quadratic polynominal:

$$\frac{\Delta Pt}{Ptr_{1A} - Ps_2} = f(\bar{\omega}r_{optimum}, ir)$$

For positive incidence angle, the polynominal is followed up to β rmp, then a linear multiplier is used for higher incidence angles. For negative incidence angles, the polynominal is followed down to β rmm, then a linear multiplier is used for lower incidence angles



$$EXPN = 0.$$

$$EXPP = 0.$$

$$\eta rr_{i,k} = 1.0$$

Test (ir)

Incidence angle is zero

1. $$\bar{\omega}_{1A2} = \bar{\omega}r_{i,k}$$

Incidence is positive

$$AS = B1_{i,k} \quad \text{(square coeff.)}$$
$$AC = B2_{i,k} \quad \text{(cubic coeff.)}$$
$$AQ = B3_{i,k} \quad \text{(quartic coeff.)}$$

Test $(\beta rmp - \beta_{1A})$ i.e. $(\beta rmp - \left[\beta_{1A}^{*}) - (\beta_{1A} - \beta_{1A}^{*})\right]$

Within positive limit, Go to (6)

Exceeds positive limit

3. $$\bar{\omega}m/\bar{\omega}r = Ir_{i,k}/(\beta rmp - \beta_{1A}^{*})_{i,k}$$

$$AR = (\beta rmp - \beta_{1A}^{*})_{i,k}/(\beta rr - \beta_{1A}^{*})_{i,k}$$

Go to (7)

Incidence is negative

$$AS = B4_{i,k} \quad \text{(square coeff.)}$$
$$AC = B5_{i,k} \quad \text{(cubic coeff.)}$$
$$AQ = B6_{i,k} \quad \text{(quartic coeff.)}$$

Test $(\beta_{1A} - \beta rmn)$ i.e. $\left[(\beta_{1A} - \beta_{1A}^{*}) - (\beta rmn - \beta_{1A}^{*})\right]$

Within negative limit, Go to (6)

Exceeds negative limit

5. $$\bar{\omega}m/\bar{\omega}r = Ir_{i,k}/(\beta rmn - \beta_{1A}^{*})_{i,k}$$

$$AR = (\beta rmn - \beta_{1A}^{*})_{i,k}/(\beta rr - \beta_{1A}^{*})_{i,k}$$

Go to (7)

Incidence angle is within quadratic limits

6. $$\bar{\omega}m/\bar{\omega}r = 1.0$$

$$AR = (\beta_{1A} - \beta_{1A}^{*})_{i,k}/(\beta rr - \beta_{1A}^{*})_{i,k}$$

7. $$\bar{\omega}_{1A-2} = \bar{\omega}r_{i,k}\left[1. + (AR)^{2.}\left\{AS + (AR)\left[AC + (AR)\ AQ\right]\right\}\right] * \bar{\omega}m/\bar{\omega}r$$

$$\eta r_{i,k} = \frac{1. - \left\{\dfrac{1.}{(Ptr_{1A}/Ps_2)_{i,k} * (1. - \bar{\omega}_{1A}-2) + \bar{\omega}_{1A}-2}\right\}^{\frac{\gamma 1Ak}{\gamma 1A_k - 1.}}}{(\emptyset_{2i,k} - 1.)/\emptyset_{2i,k}}$$

Test for previous error

RETURN

APPENDIX 1R

LOOP - LOOP ITERATION

SUBROUTINE LOOP

The purpose of subroutine loop is to handle all the logic for iterating to obtain the exact choke point. Eight possible conditions may arise during the calculation:

1.) UNDERFLOW - For very low pitchline pressure ratios, due to the radial variation in blade row inlet total pressure and radial variation in blade row exit static pressure, the stator tip sector or rotor root sector pressure ratio may be less than unity, (PTRN = -1.) LOOP sets KN = 1, increases PR by DELPR and returns. (PATH 18-19-21)

2.) NO CHOKE - If the station flow is less than critical (SCRIT = 0.) and this blade row is not within an iteration, LOOP returns to the normal calculations. (PATH 1-3-5-18-19-21).

3.) INITIAL CHOKE DETECTION - If the station flow is greater than critical (SCRIT = 1.) when the program is not at the last blade row of an iteration loop (ICHOKE ≠ IBRC) and when no other choke has previously occurred at this pressure ratio (IPC = 0), then all the necessary counters and variables are saved at the last condition prior to choke detection and an iteration loop is entered. The pressure ratio increment is halved, the pressure ratio reduced by the resulting amount and a return made to the most upstream blade row of the iteration loop. (PATH 1-3-6-8-9-10-11-13-14-15-16-17-19-21).

4.) CHOKE ITERATION NOW SUB-CRITICAL - If the station flow is less than critical (SCRIT = 0.), but this is the last blade row of an iteration loop (ICHOKE = IBRC), then the pressure ratio increment is halved and the pressure ratio at the start of the loop increased by the resulting increment. The iteration then continues from the first blade row involved. (PATH 1-2-3-5-7-16-17-19-21).

5.) CHOKE ITERATION SUPER-CRITICAL AGAIN - If the station flow is greater than critical (SCRIT = 1.), when the program is at the last blade row of an iteration loop (ICHOKE = IBRC), the pressure ratio increment is halved, the pressure ratio reduced by that amount and a transfer made to the start of the iteration loop. (PATH 1-2-3-6-13-14-15-16-17-19-21).

6.) MULTIPLE CHOKE - If the station flow is greater than critical (SCRIT = 1.) and a choke has occurred upstream at this pressure ratio (IPC = 1), the program is restored to the status before the previous iteration loop, the pressure ratio increment is divided by four, the last pressure ratio in use before the previous choke occurred is increased by the new increment and, as if neither of the two chokes had occurred, the program starts at the most upstream blade row being calculated. (PATH 1-3-6-8-9-10-12).

7.) CHOKE ITERATION COMPLETE - If the pressure ratio increment (DELPR) is within the iteration tolerance (TOLPR), LOOP resets the pressure ratio increment to the input value for after a first blade row choke to the last blade row choke (DELL), sets those variables which control how far upstream the computation should start, and returns to normal calculations. (PATH 1-2-4).

8.) SUPERSONIC

Set value for last blade row

$$I_j = 8 + K_{stg}$$

Increase the blade row counter

$$I_{brc} = I_{brc} + 1$$

Test for negative sector pressure ratio

If negative pressure ratio occurred,

$$P_{trn} = -1.$$

Go to (18)

1. Test for choke iteration on this blade row

If choke iteration on this blade row

2. Test pressure ratio increment with tolerance

If no choke iteration on this blade row or the increment is greater than the tolerance

3. Test for station flow critical.

4. Increment is less than tolerance-choke iteration complete.

ICHOKE = 0

Ipc = Ibrc

Iss = Ibrc

Isorr = 2 + (Ibrc/2) * 2 - Ibrc

J1 = (Isorr - 1) * 8 + Kn

Test for next blade row = last blade row

If not last blade row, $J1 < Ij$

22. Delpr = Dell

Go to (24)

If last blade row, J1 = Ij

23. Delpr = Dela

24. Lopc = 0

Choke = 1.

Lstg = Kn

Lbrc = Ibrc + 1

Go to set Jump = 0, (18)

If the station flow is sub-critical

5. Test for choke iteration on this blade row

If no choke iteration, go to set Jump = 0, (18)

If choke iteration, sub-critical branch

Isorr = 1 for stator

Isorr = 2 for rotor

7. Delpr = Delpr/2.

J1 = (Isorr - 1) * 8 + Lstg

$$Pto/Psl_{IP,J1} = Pto/Psl_{IP,J1} + Delpr$$

Go to increase loop counter (16)

If the station flow is super-critical

6. Test for super-sonic increment on this blade row

If super-sonic increment on this blade row,

Go to set Jump = 0, (18)

If not super-sonic increment on this blade row

8. Test for choke iteration

If not choke iteration

80. WRITE Blade Row Choked

9. Test for single calculation point

If single calculation point, Go to set

Jump = 0, (18)

If pressure ratio is incremented

10. Test for previous choke

If no previous choke occurred, save conditions just prior to the first choke.

11.
$$Lbrcs = Lbrc$$
$$Isorrs = Isorr$$
$$J1 = (Isorr - 1) * 8 + Lstg$$
$$Sptps = Pto/Psl_{IP,J1} - Delpr$$
$$Lstgs = Lstg$$
$$Sdelpr = Delpr$$

Go to test pass, (13)

13. Test for previous complete calculation

If
$$PASS = 1.$$

14.
$$ICHOKE = Ibrc$$
$$Delpr = .5 * Delpr$$

If
$$PASS = 0.$$

15. Decrease proper P/r by Delpr
$$J1 = (Isorr - 1) * 8 + Lstg$$
$$Pto/Psl_{IP,J1} = Pto/Psl_{IP,J1} - Delpr$$

Go to increase loop counter (16).

If previous choke was occurred, - multiple choke.

12.
$$J1 = Lstgs + (Isorrs - 1) * 8$$
$$Delnu = (Pto/Psl_{IP,J1} - Sptps)/4.$$

Test Delnu

If
$$Delnu < 0.0001,$$

Set
$$Delnu = Delpr/4.$$
$$Delpr = Delnu$$

WRITE - Two blade rows choked, new increment.
$$Lbrc = Lbrcs$$
$$Lstg = Lstgs$$
$$Pto/Psl_{IP,J1} = Sptps + Delpr$$
$$Lopc = 10$$
$$ICHOKE = 0$$
$$Ipc = 0$$
$$Iss = 0$$
$$CHOKE = 0.0$$

Go to set $Jump = 1$ (17)

16. Increase loop counter

17. Set Jump for choke iteration, Jump = 1

    Go to (19)

18. Set Jump for no choke or choke iteration complete,

    Jump = 0

19. Test (TRLOOP)

If TRLOOP is positive

20. WRITE Ibrc, Lbrc, Isorr, Kn, Lstg, Ipc, Iss, ICHOKE, Jump, Lbrcs, Isorrs, Lstgs, Sptps, $Pto/Psl_{IP,J1}$, Delpr, Dell, Scrit, Lopc.

21. RETURN

APPENDIX 1S

STA2A - STATION 2A

SUBROUTINE STA2A

The purpose of subroutine STA2A is to determine inlet flow conditions to all stators after the first and find stator incidence angle recovery. Stage exit average conditions $\bar{P}$ and $\bar{T}$ are determined based on an average weighted entropy. The profile averaging fork (PAF) is tested to set the inlet profile for the next stage inlet. Three profile options are available. Start at pitchline to obtain gas properties for the station, calculate into root then out to tip. Absolute tangential component of velocity is adjusted for diameter change to conserve angular momentum. Axial component of velocity is adjusted for weight flow change, area change, and density change from STA1. The radial gradient in pressure does not satisfy radial equilibrium, and continuity is conserved in each sector.

Set i to pitchline value

Set $\triangle$i to -1

Set first guess of $\quad Ts_{2A_{i,k}} = Ts_{2_{i,k}}$

Find ratio of flow change $\quad Wr = RWG_{2A_k}/RWG_{2_k}$

Set total station flow $\quad Wgt_{2A_k} = Wr * Wgt_{2_k}$

Initialize summations

$$\left[\sum (TR)Wg_{2A}/Wgt_{2A} = 0.0\right]$$

$$\left[\sum \ln(TR)Wg_{2A}/Wgt_{2A} = 0.0\right]$$

$$\left[\sum \ln(PR)Wg_{2A}/Wgt_{2A} = 0.0\right]$$

Adjust tangential velocity

12. $\quad Vu_{2A_{i,k}} = Vu_{2_{i,k}} * Dp_{2_{i,k}}/Dp_{2A_{i,k}}$

Starting density $\quad \rho str = \rho s_{2_{i,k}}$

Adjust flow $\quad Wg_{2A_{i,k}} = Wr * Wg_{2_{i,k}}$

Adjust axial velocity

1. $$Vz_{2A_{i,k}} = Wr * Vz_{2_{i,k}} * \rho s_{2_{i,k}} * Ann_{2_{i,k}} / (\rho str * Ann_{2A_{i,k}})$$

$$V_{2A_{i,k}} = \sqrt{(Vu_{2A_{i,k}})^{2.} + (Vz_{2A_{i,k}})^{2.}}$$

Test for pitch sector

If pitch sector $i = IP$

2. Test gamma fork

If gamma not input, $\gamma F = 0.0$

3. $$Ta_{2A} = .5 * (Ttr_{2_{i,k}} + Ts_{2A_{i,k}})$$

CALL GAMMA($Ptr_{2_{i,k}}$, Ta_{2A}, FAIR,WAIR, $\gamma 2A_k$)

If gamma input, $\gamma F = 1.0$

4. $$Cp_{2A_k} = RG * \gamma 2A_k / \left[(\gamma 2A_k - 1.) * J\right]$$

$$\Delta Ts = \left[(V_{2_{i,k}})^{2.} - (V_{2A_{i,k}})^{2.}\right] \Big/ (2. * g * J * Cp_{2A_k})$$

$$Ts_{2A_{i,k}} = Ts_{2_{i,k}} + \Delta Ts$$

32. $$Ps_{2A_{i,k}} = Ps_{2_{i,k}} * (1. + \Delta Ts / Ts_{2_{i,k}})^{\frac{\gamma 2A_k}{\gamma 2Ak - 1.0}}$$

$$\rho s_{2A} = 144. * Ps_{2A_{i,k}} / (RG * Ts_{2_{i,k}})$$

Find density error

$$\rho e = (\rho s_{2A} - \rho str) / \rho s_{2A}$$

Test($|\rho e| - \rho tol$)

If error $>$ tol.

5. $$\rho str = \rho s_{2A}$$

Go to Calculation new Vz_{2A} (1)

If error $<$ tol.

6. $$\sin\alpha_{2A} = Vu_{2Ai,k} / V_{2A_{i,k}}$$

$$\alpha_{2A_{i,k}} = \tan^{-1}\left[\sin\alpha_{2A} \Big/ \sqrt{1. - \sin^{2.}\alpha_{2A}}\right]$$

Test for pitch sector

If pitch sector, $i = IP.$

24. Test gamma fork

If gamma not input, $\gamma F = 0.$

25. $$Tas_k = .5 * (Ta_{1k} + Ta_{2k})$$

$$Pas_k = .5 * (Pt0_{IP,k} + Pt_{2AIP,k})$$

CALL GAMMA(Pas_k, Tas_k, FAIR,WAIR, γs_k)

Go to (27)

If gamma input, $\gamma F = 1.0$

26. $$\gamma s_k = .5 * (\gamma 1_k + \gamma 2_k)$$

$$Cps_k = RG * \gamma s_k / \left[(\gamma s_k - 1.) * J\right]$$

28. $$\Delta hVD_{i,k} = (U_{1A_{i,k}} * Vu_{1Ai,k} + U_{2i,k} * Vu_{2i,k}) /(g * J)$$

$$M_{2A_{i,k}} = V_{2A_{i,k}} / \sqrt{\gamma 2A_{i,k} * g * RG * Ts_{2A_{i,k}}}$$

$$\Delta Tt = TFR_{i,k} * \Delta hVD_{i,k} /Cps_k$$

$$Tt_{2Ai,k} = Tt0_{i,k} - \Delta Tt$$

$$Tt/Ts_{2A_{i,k}} = 1. + (M_{2A_{i,k}})^{2.} * (\gamma 2A_k - 1.)/2.$$

$$Pt/Ps_{2A_{i,k}} = (Tt/Ts_{2A_{i,k}})^{\frac{\gamma 2Ak}{\gamma 2A_k - 1.}}$$

$$Pt_{2A_{i,k}} = Ps_{2A_{i,k}} * (Pt/Ps_{2A_{i,k}})$$

$$Mf_{2Ai,k} = M_{2A_{i,k}} * \cos(\alpha_{2A_{i,k}})$$

Test for tip sector

If not the tip sector $i \neq ISECT$

13. $i = i + \Delta i$

Test (i)

If below root, $i = 0.$

14. $\Delta i = -1$

$i = IP + \Delta i$

Go to next sector (12)

If tip sector is completed, $i = ISECT$

15. $i = 1, ISECT$

$$\sum_{i=1}^{ISECT} (Tt_{2A_{i,k}}/Tt_{2A_{IP,k}}) * (Wg_{2A_{i,k}}/Wgt_{2A_k})$$

$$\sum_{i=1}^{ISECT} \ln(Tt_{2A_{i,k}}/Tt_{2A_{IP,k}}) * (Wg_{2A_{i,k}}/Wgt_{2A_k})$$

$$\sum_{i=1}^{ISECT} \ln(Pt_{2A_{i,k}}/Pt_{2A_{IP,k}}) * (Wg_{2A_{i,k}}/Wgt_{2A_{i,k}})$$

$$\overline{Tt}_k = Tt_{2A_{IP,k}} * \sum (TR) Wg_{2A_{i,k}}/Wgt_{2A_k}$$

$$\overline{Pt}_k = Pt_{2A_{IP,k}} * e^{[Power]}$$

where

$$[Power] = \sum_{i=1}^{ISECT} \ln(PR) * \frac{Wg_{2A_{i,k}}}{Wgt_{2A_k}} + \frac{\gamma_{2A_k}}{(\gamma_{2A_k} - 1.)} \left[\ln(TR) * \frac{Wg_{2A_{i,k}}}{Wgt_{2A_{i,k}}} - \sum \ln(TR) * \frac{Wg_{2A_{i,k}}}{Wgt_{2A_{i,k}}} \right]$$

Test for last stage

If not last stage

17. $Tt0s_{k+1} = \overline{Tt}_k$

$$Pt0s_{k+1} = \overline{Pt}_k$$

$i = 1, ISECT$

29. $Is_{i,k+1} = \alpha_{2A_{i,k}} - \alpha_0{}^*_{i,k+1}$

If $Is > \pi/2$,

Set $Is = \pi/2$

If $Is < -\pi/2$

Set $Is = -\pi/2$

Test $\bar{\omega}s$

If $\bar{\omega}s$ is input,

7. $\eta rs_{i,k} = 1.0$

$Expsi = 0.$

Go to (117)

If $\bar{\omega}s$ is not input

8. Test (Is)

For negative incidence

9. $$Expsi = Expn$$

Go to (117)

For positive incidence

10. $$Expsi = Expp$$

117. Test passage area fork (PAF - 1.)

If $$PAF = 0.,$$

uniform profiles

19. $$Ptp_{i,k+1} = \overline{Pt}_k$$

$$Pt0_{i,k+1} = \frac{Ptp_{i,k+1}\left[1. + \left(\frac{Tt}{Ts_{2A_{i,k}}} - 1.\right) * \eta rs_{i,k} * \cos^{Expsi} Is_{i,k}\right]^{\frac{\gamma 2A_k}{\gamma 2A_k - 1.}}}{\left(\frac{Tt}{Ts_{2A_{i,k}}}\right)^{\frac{\gamma 2A_k}{\gamma 2A_k - 1.}}}$$

$$Tt0_{i,k+1} = Tt_k$$

Go to (23)

If $$PAF = 1.0,$$

save profiles

20. $$Ptp_{i,k+1} = Pt_{2A_{i,k}}$$

$$Pt0_{i,k+1} = \frac{Ptp_{i,k+1}\left[1. + \left(\frac{Tt}{Ts_{2A_{i,k}}} - 1.\right) * \eta rs_{i,k} * \cos^{Expsi} Is_{i,k}\right]^{\frac{\gamma 2A_k}{\gamma 2A_k - 1.}}}{\left(\frac{Tt}{Ts_{2A_{i,k}}}\right)^{\frac{\gamma 2A_k}{\gamma 2A_k - 1.}}}$$

Go to (22)

If $$PAF = 2.0,$$

smooth pressure profile

21. $$Ptp_{i,k+1} = \frac{\overline{Pt}_k}{\left(\frac{\overline{Tt}_k}{Tt_{2A_{i,k}}}\right)^{\frac{\gamma 2A_k}{\gamma 2A_k - 1.}}}$$

22. $$Tt_{0_{i,k+1}} = Tt_{2A_{i,k}}$$

23. Continue

18. $$Mf_{stop} = Mf_{2A_{IP,k}}/Aacs$$

Test for previous error

If error

30. CALL DIAGT

If no error

31. RETURN

APPENDIX 1T

STA1 - STATION 1

SUBROUTINE STA1

The purpose of subroutine STA1 is to satisfy continuity of flow and radial equilibrium at the exit of all stators after the first. Flow is adjusted for cooling air injection between stations 2A and 1. The starting value of pitchline pressure ratio is obtained from a one-dimensional flow function per unit area relationship. Incidence angle loss in total pressure will vary with pressure ratio and sector number. Simple radial equilibrium is used to balance the radial distribution of flow.

Set index for flow loop = 1

Set index for critical condition = 0.0

$$Wr = Rwg1_k / Rwg2A_{k-1}$$

$$i = 1, ISECT$$

$$Wg1_{i,k} = Wr * Wg_{2A_{i,k-1}}$$

$$\alpha o_{i,k} = \alpha_{2A_{i,k-1}}$$

$$Ps0_{i,k} = Ps_{2A_{i,k-1}}$$

$$V0_{i,k} = V_{2A_{i,k-1}}$$

$$Ts0_{i,k} = Ts_{2A_{i,k-1}}$$

$$Vu0_{i,k} = Vu_{2A_{i,k-1}}$$

$$Vz0_{i,k} = Vz_{2A_{i,k-1}}$$

$$M0_{i,k} = M_{2A_{i,k-1}}$$

1. $$Wgt1_k = Wr * Wgt_{2A_{k-1}}$$

End of i loop

Set $i = IP$

$\Delta i = -1$

$Wgtlc_k = 0.0$

Set index for pressure ratio limit = 0.0

Test for choke iteration

If choke iteration, $ICHOKE \neq 0$,

Go to (16)

If not choke iteration

17. Test loop iteration

If loop iteration, $LOPIN \neq 0$,

Go to (16)

If not loop iteration

18. Test γF

If gamma not input, $\gamma F = 0.0$

$$Ta1_k = .95 * Tto_{IP,k}$$

CALL GAMMA($Pto_{IP,k}$, $Ta1_k$, FAIR, WAIR, $\gamma 1_k$)

If gamma input, $\gamma F = 1.0$

3. $FF/A1_{i,k} = Wg1_{i,k} * \sqrt{Tto_{i,k}} / (144. * Pto_{i,k} * Annl_{i,k} * \cos\alpha 1_{i,k})$

CALL PRATIO($FF/A1_{i,k}$, $\gamma 1_k$, RG, $Pto/Ps1_{i,k}$, Prtol)

16. CALL FLOW1

Test for previous error

If error, Go to (25)

If no error occurred

$$Wgtlc_k = Wgtlc_k + Wg1_{i,k}$$

$$\ell = 1$$

Test for pressure ratio less than pitchline.

If pressure ratio is less than pitchline,

Set $\ell = i$

Test for tip sector

If not tip sector, $i < ISECT$

4. $i = i + \Delta i$

Test for lower than root sector

If lower than root, $i = 0$

5. $\Delta i = -1$

$i = IP + \Delta i$

If not lower than root, $i > 0$

Find index for dp/dr

6. $l = i - \Delta i$

$$Ps1_{i,k} = Ps1_{l,k} + FLOAT(\Delta i) * dp/dr1_{\ell,k} * (h1_{\ell,k} + h1_{i,k})/2.$$

$$Pt_0/Ps1_{i,k} = Pt0_{i,k}/Ps1_{i,k}$$

Go to CALL FLOW1 (16)

If tip sector, $i = ISECT$

Test index for pressure ratio limit

If first time past, counter = 0

8. Set index for pressure ratio limit = 1

$$Pr_{crit} = \left[(\gamma 1_k + 1.)/2.\right]^{\frac{\gamma 1_k}{\gamma 1_k - 1.}}$$

$$Pr_{up} = Pto/Ps1_{IP,k} * Pr_{crit}/(Pto/Ps1_{\ell,k}) * (1. + Prtol)$$

$$Pr_{low} = 1.0$$

On pressure ratio interation,counter $> 0.$

9. Set counter to value + 1.

10. Set counter for blade row

Test for choke iteration on this blade row.

If this blade row is on a choke iteration, set

$$Pto/Ps1_{IP,k} = Pr_{up}$$

11. If $Wgt1c < Wgt1$: $Pr_{low} = Pto/Ps1_{IP,k}$

12. If $Wgt1c > Wgt1$: $Pr_{up} = Pto/Ps1_{IP,k}$

13. $We = 1. - Wgt1_k/Wgt1c_k$

Increase flow loop index

Test (Index - 32)

Fails to converge in 32 passes

GO to WRITE ERROR (22)

29. Test for choke iteration on this blade row.

If this blade row is on a choke iteration,

ICHOKE = L

31. Set $SCRIT = -We$

Go to test TRLOOP (15)

30. Test for loop iteration

If loop iteration, $LOPIN = 1,$

Go to (15)

If not on a loop iteration

14. $Pre = (Pto/Ps1_{IP,k} - Ptrmo)/Pto/Ps1_{IP,k}$

Test $|Pre| - Prtol$

If pressure ratio error is less than tolerance

Test $|We| - Wtol$

If flow error is less than tolerance,

Go to test TRLOOP (15)

If flow error is greater than tolerance,

Set $Scrit = 1.0$

If pressure ratio error is greater than tolerance

27. $Ptrmo = Pto/Ps1_{IP,k}$

$Wgt1c_k = 0.0$

$i = IP$

$\Delta i = -1$

Test Scrit

If $Scrit = 1.,$

Go to test TRLOOP (15)

If $Scrit = 0.$

19. $Pto/Ps1_{IP,k} = .5 * (Pr_{up} + Pr_{low})$

Test $(Pto/Ps1_{IP,k} - Pr_{crit})$

If $Pto/Ps1_{IP,k} < Pr_{crit},$

Set $PRPC = 0.$

Go to CALL FLOW1 (16)

20. $Scrit = 1.0$

15. Test for TRLOOP

If $TRLOOP \neq 0$

WRITE: Pr_{up}, Pr_{low}, We, Pr_{crit}, j, $Wgt1_k$, $Wgt1c_k$, $Wg1_{i,k}$, $Pto/Ps1_{i,k}$

If $TRLOOP = 0$

28. Test for previous errror

If error has occurred

23. CALL DIAGT

If no error has occurred

24. CALL LOOP

25. RETURN

APPENDIX 1U

OVRALL - OVERALL PERFORMANCE

SUBROUTINE OVRALL

The purpose of subroutine OVRALL is to calculate turbine overall performance from Stage 1 inlet to last stage exit. Find the average overall γ and Cp, $\Sigma Up^2, \Sigma Ur^2, \Sigma \Delta h$, Δht_I, Δs_I, Δhat_I

Initialize inlet data and overall summations.

$$Tt0s = Ttin$$
$$Pt0s = Ptin$$
$$Ta0 = 0.$$
$$Pa0 = 0.$$
$$\gamma 0 = 0.$$
$$Up^2o = 0.$$
$$Ur^2o = 0.$$
$$\Delta hto = 0.$$

$$k = 1, Kstg$$

Test for gamma input

If gamma not input

1.

$$\Sigma Tao = \Sigma Tas_k$$
$$\Sigma Pao = \Sigma Pas_k$$

If gamma input

2.

$$\Sigma \gamma o = \Sigma \gamma s_k$$

Initialize stage summations

$$\Delta hts = 0.$$
$$\Delta htIs = 0.$$
$$\Delta hsIs = 0.$$
$$\Delta htIs = 0.$$
$$i = 1, ISECT$$
$$Rw = Wg2A_{i,k}/Wgt2A_k$$
$$\Delta ht_{i,k} = \Delta hvd_{i,k} * TF_{i,k}$$

$$\Delta ht_{I_{i,k}} = Cps_k * Tt0_{i,k}\left[1. - (Pt_{2A_{i,k}}/Pt_{p_{i,k}})^{\frac{\gamma s}{\gamma s - 1.}}\right]$$

$$\eta tt_{i,k} = \Delta ht_{i,k}/\Delta ht_{I_{i,k}}$$

$$\Delta hs_{I_{i,k}} = Cps_k * Tt0_{i,k}\left[1. - (Ps_{2A_{i,k}}/Pt_{p_{i,k}})^{\frac{\gamma s}{\gamma s - 1.}}\right]$$

$$\eta ts_{i,k} = \Delta ht_{i,k}/\Delta hs_{I_{i,k}}$$

$$Ptat_{i,k} = Ps_{2A_{i,k}} * \left[1. + (\gamma 2A_k - 1/2.) * (Mf_{2A_{i,k}})^{2.}\right]^{\frac{\gamma 2A}{\gamma 2A - 1}}$$

$$\Delta hat_{I_{i,k}} = Cps_k * Tt0_{i,k}\left[1. - (Ptat_{2A_{i,k}}/Pt_{p_{i,k}})\right]^{\frac{\gamma s}{\gamma s - 1.}}$$

$$\eta tat_{i,k} = \Delta ht_{i,k}/\Delta hat_{I_{i,k}}$$

$$\Delta hts_k = \sum Rw * \Delta ht_{i,k}$$

$$\Delta ht_{Is_k} = \sum Rw * \Delta ht_{I_{i,k}}$$

$$\Delta hs_{Is_k} = \sum Rw * \Delta hs_{I_{i,k}}$$

$$\Delta hat_{Is_k} = \sum Rw * \Delta hat_{I_{i,k}}$$

end of i loop

Stage output

$$\alpha 0s_k = \alpha 0_{IP,k} * 180./\pi$$

$$Iss_k = Is_{IP,k} * 180./\pi$$

$$\beta 1As_k = \beta 1A_{IP,k} * 180./\pi$$

$$Irs_k = Ir_{IP,k} * 180./\pi$$

$$\alpha 2s_k = \alpha 2_{IP,k} * 180./\pi$$

$$\theta cr_k = \left(\left[(\gamma 0_k/\gamma 0k + 1.) * RG * Ttos_k\right] / \left[(\gamma s1/\gamma s1 + 1.) * Rs1 * Ts1\right]\right)$$

$$\epsilon_k = \gamma s1/\gamma 0_k \left[(\gamma 0_k + 1./2)^{\frac{\gamma 0_k}{\gamma 0_k - 1.}} / (\gamma s1 + 1./2)^{\frac{\gamma s1}{\gamma s1 - 1.}} \right]$$

$$\delta_k = Pt0s_k/Ps1$$

$$\eta tts_k = \Delta hts_k/\Delta htIs_k$$

$$\eta tss_k = \Delta hts_k/\Delta hsIs_k$$

$$\eta tats_k = \Delta hts_k/\Delta hatIs_k$$

$$Wgt0_k = Wgt_1/RWG_1$$

$$W\sqrt{T}/Ps_k = Wgt0_k * \sqrt{Tt0s_k}\ /Pt0s_k$$

$$N/\sqrt{T}s_k = RPM/\sqrt{Tt0s_k}$$

$$\Delta h/Ts_k = \Delta hts_k/Tt0s_k$$

$$\Delta h/\theta crs_k = \Delta hts_k/\theta cr_k$$

$$N/\sqrt{\theta cr}s_k = RPM/\sqrt{\theta cr_k}$$

$$W\sqrt{\theta cr}\,\epsilon/\delta_k = Wgt0_k * \sqrt{\theta cr_k} * \epsilon_k/\delta_k$$

$$Pt0/Pt2_k = Pt0s_k/\overline{Pt}_k$$

$$Pt0/Ps2_k = Pt0s_k/Ps2_{IP,k}$$

$$Tt_2/Tt0_k = \overline{Tt}_k/Tt0s_k$$

$$Ttr_{1A}/Tt0_k = Ttr_{1A_{IP,k}}/Tt0s_k$$

$$Up^2{}_k = (U_{1A_{IP,k}} + U_{2_{IP,k}}/2.)^{2.}$$

$$Ur^2_k = \left[U_{1A_{1,k}} * Dr_{1A_k}/Dp_{1A_k} + U_{2_{1,k}} * Dr_{2_k}/Dp_{2_k} \Big/ 2.\right]^{2.}$$

$$Up^2{}_o = \sum Up^2{}_k$$

$$Ur^2{}_o = \sum Ur^2{}_k$$

$$\Delta hto = \sum \Delta ht_{s_k}$$

Test (ΔhsI)

If negative or zero

14. $$Visen = 1.0 \quad \text{(dummy isentropic velocity)}$$

If positive

15. $$Visen_k = \sqrt{2. * g * J * \Delta hsI_{IP,k}}$$

16. $$Up/Visen_k = Up_{s_k}/Visen_k$$

$$Ur/Visen_k = Ur_{s_k}/Visen_k$$

$$\Psi p_{s_k} = g * J * \Delta ht_{s_k}/(2. * Up^2{}_k)$$

$$\Psi r_{s_k} = g * J * \Delta ht_{s_k}/(2. * Ur^2{}_k)$$

$$Rx_{p_k} = 1. - \left[1. - (Ps1_{IP,k}/Pt_{p_{IP,k}})^{\frac{\gamma s_k - 1.}{\gamma s_k}} \Big/ 1. - (Ps2_{IP,k}/Pt_{p_{IP,k}})^{\frac{\gamma s_k - 1.}{\gamma s_k}}\right]$$

Root values

$$Vu1_r = Vu1_{1,k} * Dp1_{1,k}/Dr1_k$$

$$V1_r = \sqrt{(Vu1_r)^{2.} + (Vz1_{i,k})^2}$$

$$\emptyset r = 1./\left[1. - V1r^{2.}/(2. * g * J * Cp1_k * Tt_{01,k} * \eta_{s_{1,k}})\right]$$

$$Pt_0/Ps1_r = (\emptyset_r)^{\frac{\gamma 1k}{\gamma 1k - 1.}} * Ptp_{1,k}/Pt_{o_{1,k}}$$

$$Rx_{r_k} = 1. - \left[1. - (1./(Pto/Ps1_r)^{\frac{\gamma sk}{\gamma sk - 1}} \Big/ 1. - (Ps2_{1,k}/Ptp_{1,k})^{\gamma sk - 1}\right]$$

$$\Delta\beta r_k = (\beta_{1A_{1,k}} + \beta_{2e_{1,k}}) * 180./\pi$$

$$Ml s_k = Vl_{IP,k} / \sqrt{\gamma l_k * g * RG * Tsl_{IP,k}}$$

$$Tsl_r = Tt0_{1,k} - (Vl_r)^{2.} / (2. * g * J * Cp1_k)$$

$$Mlrs_k = Vlr / \sqrt{\gamma l_k * g * RG * Tsl_r}$$

$$Vu1A_r = Vu1A_{i,k} * Dp1A_{1,k} / Dr1A_k$$

$$V1A_r = \sqrt{(Vu1A_r)^{2.} + (Vz1A_{i,k})^{2.}}$$

$$Ts1A_r = Tt0_{1,k} - (V1A_r)^{2.} / (2. * g * J * Cp1A_k)$$

$$Ru1A_r = Vu1A_r - U1A_{1,k} * Dr1A_k / Dp1A_{1,k}$$

$$R1A_r = \sqrt{(Ru1A_r)^{2.} + (Vz1A_{1,k})^{2.}}$$

$$Mr1A_r = R1A_r / \sqrt{\gamma 1A_k * g * RG * Ts1A_r}$$

Tip values

$$Vu2_t = Vu2_{ISECT,k} * Dp2_{ISECT,k} / Dt2_k$$

$$V2_t = \sqrt{(Vu2_t)^{2.} + (Vz2_{ISECT,k})^{2.}}$$

$$Ts2_t = Vu2_t + \left[(V2_{ISECT,k})^{2.} - (V2_t)^{2.} \right] / (2. * g * RG * Cp2_k)$$

$$Ru2_t = Vu2_t + U2_{ISECT,k} * Dt2_k / Dp2_{ISECT,k}$$

$$R2_t = \sqrt{(Ru2_t)^{2.} + (Vz2_{ISECT,k})^{2.}}$$

$$Mr2_t = R2_t / \sqrt{\gamma 2_k * g * RG * Ts2_t}$$

17. end of k loop

Test for gamma input

If gamma is not input, $\gamma F = 0.$

4. $Tao = \sum Tao/Kstg$

$Pao = \sum Pao/Kstg$

CALL GAMMA(Pao,Tao,FAIR,WAIR,γo)

If gamma is input, $\gamma F = 1.$

7. $$\gamma o = \Sigma \gamma o / Kstg$$

8. $$Cpo = RG(\gamma o - 1.)/(\gamma o * J)$$

Initialize exit summations

$$\Delta hT_{I_o} = 0.$$

$$\Delta hs_{I_o} = 0.$$

$$\Delta hat_{I_o} = 0.$$

$$k = Kstg$$

$$i = I, ISECT$$

$$Rw = Wg_{2A_{i,k}}/Wgt_{2A_k}$$

$$\Delta ht_{I_o} = \Sigma Cp_o * Tt0_{I,1} * \left[1. - Rw(Pt2A_{i,k}/Pt_{p_{i,1}})^{\frac{\gamma o - 1.}{\gamma o}}\right]$$

$$\Delta hs_{I_o} = \Sigma Cp_o * Tt0_{I,1} * \left[1. - Rw(Ps_{2A_{i,k}}/Pt_{p_{i,1}})^{\frac{\gamma o - 1.}{\gamma o}}\right]$$

9. $$\Delta hat_{I_o} = \Sigma Cp_o * Tt0_{I,1} * \left[1. - Rw(Pat_{2A_{i,k}}/Pt_{p_{i,1}})^{\frac{\gamma o - 1.}{\gamma o}}\right]$$

end of i loop

$$\psi_{Po} = g * J * \Delta ht_o/(2. * Up^2_o)$$

$$\psi r_o = g * J * \Delta ht_o/(2. * Ur^2_o)$$

$$W\sqrt{T}/P_o = W\sqrt{T}/Ps_1$$

$$(WN\epsilon/60\delta)_o = W\sqrt{\theta cr}\,\epsilon/\delta_1 * N/\sqrt{\theta cr}_{s_1}/60.$$

$$N/\sqrt{\theta cr}_o = N/\sqrt{\theta cr}_{s_1}$$

$$N/\sqrt{T}_o = N/\sqrt{T}_{s_1}$$

$$\Delta h/T_o = \Delta ht_o/Ttin$$

$$Pt0/Pt2_o = \overline{Pt}_{in}/Pt_{K_{stg}}$$

$$Pt0/Ps2_o = Pt_{in}/Ps2_{IP,Kstg}$$

$$Pt0/Pat2_o = Pt_{in}/Pat_{2A_{IP,Kstg}}$$

$$\eta tt_o = \Delta ht_o / \Delta ht_{I_o}$$

$$\eta ts_o = \Delta ht_o / \Delta hs_{I_o}$$

$$\eta tat_o = \Delta ht_o / \Delta hat_{I_o}$$

$$\Delta h/\theta cr = \Delta ht_o / \theta cr_1$$

WRITE OVRALL output

RETURN

APPENDIX IV

INSTG - INTERSTAGE EQUATIONS

SUBROUTINE INSTG

The purpose of subroutine INSTG is to set up for printout of interstage output. For more than three sectors, only the sector pitchline values are printed. For three sectors or less, the hub value and casing value are also calculated based on a free vortex distribution in the root sector and tip sector respectively. Interstage data is relocated or calculated one stage at a time.

$$k = 1, Kstg$$

Relocate pitchline values, convert angles to degrees.

4. $i = 1, Isect$

$$\rho s0 = 144. * Ps0_{i,k}/(Ts0_{i,k} * RG)$$

$$\Delta\alpha = \alpha 0_{i,k} + \alpha 1E_{i,k}$$

$$M1 = V1_{i,k}/\sqrt{\gamma 1_k * g * RG * Ts1_{i,k}}$$

$$Zinc_s = \cos(90 - 2\alpha 1E_{i,k}) * \left[(\tan\alpha 0_{i,k}/\tan\alpha 1E_{i,k}) + 1.\right]$$

$$Cp_s = 1. - (V0_{i,k}/V1_{i,k})^{2.}$$

$$\Delta\beta = \beta 1A_{i,k} + \beta 2E_{i,k}$$

$$Rxp = 1. - \left[1. - (Ps1_{i,k}/Ptp_{i,k})^{\frac{\gamma s_k - 1.}{\gamma s_k}} \Big/ 1. - (Ps2_{i,k}/Ptp_{i,k})^{\frac{\gamma s_k - 1.}{\gamma s_k}}\right]$$

$$\Psi p = 2. * g * J * \Delta ht_{i,k}/\left[(U_{1Ai,k})^{2.} + (U2_{i,k})^{2.}\right]$$

$$Zinc_r = \cos(90 - 2\beta 2E_{i,k}) * \left[(\tan\beta 1A_{i,k}/\tan\beta 2E_{i,k}) + 1.\right]$$

$$Cp_r = 1. - (R1A_{i,k}/R2_{i,k})^{2.}$$

5. $$\rho_s 2 = 144. * Ps2_{i,k}/(Ts2_{i,k} * RG)$$

end of i loop

Calculate hub values, relocate constant values.

$$\tan\alpha 1_r = \tan\alpha 1_{i,k} * Dp1A_{i,k}/Dr1A_k$$

Station 0 - Stator Inlet

10. $$Vu0_h = Vu0_{1,k} * Dp0_{1,k}/Dr0_k$$

$$V0_h = \sqrt{(Vu_h)^{2.} + (Vz0_{i,k})^{2.}}$$

$$Ts0_h = Tt0_{1,k} - (V0_h)^{2.}/(2. * g * J * Cp0_k)$$

$$Ps0_h = Ps0_{1,k} * (Ts0_h/Ts0_{1,k})^{\frac{\gamma 0_k}{\gamma 0_k - 1.}}$$

$$\rho s0_h = 144. * Ps0_h/(RG * Ts0_h)$$

$$\alpha 0_h = \tan^{-1}(Vu0_h/Vz0_h)$$

$$Is_h = \alpha 0_h - \tan^{-1}(\tan\alpha_0^* * Dp0_{1,k}/Dr0_k)$$

$$As0_h = \sqrt{\gamma o_k * g * RG * Ts0_h}$$

$$M0_h = V0_r/As0_h$$

Station 1 - Stator Exit.

$$Vu1_h = Vu1_{1,k} * Dp1_{1,k}/Dr1_k$$

$$V1_h = \sqrt{(Vz1_{1,k})^{2.} + (Vu1_h)^{2.}}$$

$$Ts1_h = Tt0_{1,k} - (V1_h)^{2.}/(2 * g * J * Cp1_k)$$

$$Ps1_h = Ps1_{1,k} * (Ts1_h/Ts1_{1,k})^{\frac{\gamma 1_k}{\gamma 1_k - 1.}}$$

$$\rho s1_h = 144. * Ps1_h/(RG * Ts1_h)$$

$$\alpha 1E_h = \tan^{-1}(Vu1_h/Vz1_h)$$

$$\Delta\alpha_h = \alpha 0_h + \alpha 1E_h$$

$$As1_h = \sqrt{\gamma 1_k * g * RG * Ts1_h}$$

$$M1_h = V1_r/As1_h$$

$$Zinc_{s_h} = \cos(90 - 2\alpha 1E_h) * \left[(\tan\alpha 0_h/\tan\alpha 1E_h) + 1.\right]$$

$$Cps_h = 1. - (V0_h/V1_h)^{2.}$$

Station 1A - Rotor Inlet.

$$Vu1A_h = Vu1A_{1,k} * Dp1A_{1,k}/Dr1A_k$$

$$Ru1A_h = Vu1A_h - V1A_{1,k} * Dr1A_k/Dp1A_{1,k}$$

$$\beta 1_h = \tan^{-1}(Ru1A_h/Vz1A_{1,k})$$

$$Ir_h = \beta 1_h - \tan^{-1}\left[\tan\alpha 1_r - (\tan\alpha 1_{1,k} - \tan\beta 1A^*_{1,k}) * Dr1A_k/Dp1A_{1,k}\right]$$

$$R1A_h = \sqrt{(Ru1A_h)^{2.} + (Vz1A_{1,k})^{2.}}$$

$$\Delta Ts_h = (V1_{1,k})^{2.} - \left[(Vz1A_{i,k})^{2.} + (Vu1A_h)^{2.}\right] \Big/ (2. * g * J * Cp1A_k)$$

$$Ts1A_k = Ts1_{1,k} + \Delta Ts_h$$

$$Mr1A_h = R1A_h/\sqrt{\gamma 1A_k * g * R * Ts1A_h}$$

$$Ttr/Ts1A_h = 1. + (Mr1A_h)^{2.} * (\gamma 1A_{k-1.})/2.$$

$$Ttr1A_h = Ts1A_h/(Ttr/Ts1A_h)$$

Test Ir

If negative incidence

2. $$Expri = Expn$$

If positive incidence

7. $$Expri = Expp$$

11. $$Ptr/Ps_{1A_h} = \left\{1. + \left[(Ttr/Ts_{1A_h}) - 1.\right] * rr_{1,k} * (\cos Ir_{1,k})^{Expri}\right\}^{\frac{\gamma 1A_k}{\gamma 1A_k - 1.}}$$

$$Ptr_{1A_h} = Ps_{1A_h} * (Ptr/Ps_{1A_h})$$

$$U_{1A_h} = U_{1A_{1,k}} * Dr_{1A_k}/Dp_{1A_{1,k}}$$

Station 2 - Rotor Exit.

$$Vu_{2_h} = Vu_{2_{1,k}} * Dp_{2_{1,k}}/Dr_{2_k}$$

$$Ru_{2_h} = Vu_{2_h} - U_{2_{1,k}} * Dr_{2_k}/Dp_{2_{1,k}}$$

$$\beta_{2_h} = \tan^{-1}(Ru_{2_h}/Vz_{2_{1,k}})$$

$$\Delta\beta_h = \beta_{1A_h} + \beta_{2_h}$$

$$R_{2_h} = \sqrt{(Ru_{2_h})^{2.} + (Vz_{2_{1,k}})^{2.}}$$

$$\Delta Ts_h = (V_{2_{1,k}})^{2.} - \left[(Vz_{2_{1,k}})^{2.} + (Vu_{1A_h})^{2.}\right] \Big/ (2. * g * J * Cp_{2_k})$$

$$Ts_{2_h} = Ts_{2_{1,k}} + \Delta Ts_{2_h}$$

$$Mr_{2_h} = R_{2_h}/\sqrt{\gamma 2_k * g * RG * Ts_{2_h}}$$

$$U_{2_h} = U_{2_{1,k}} * Dr_{2_k}/Dp_{2_{1,k}}$$

$$Ps_{2_h} = Ps_{2_{1,k}}(Ts_{2_h}/Ts_{2_{1,k}})^{\frac{\gamma 2k}{\gamma 2k - 1.}}$$

$$Rx_h = 1. - \left(\left[1. - (Ps_{1_h}/Ptp_{1,k})^{\frac{\gamma s_k - 1.}{\gamma s_k}}\right] \Big/ \left[1. - (Ps_{2_h}/Ptp_{1,k})^{\frac{\gamma s_k - 1.}{\gamma s_k}}\right]\right)$$

$$\Delta ht_h = (U_{1A_h} * Vu_{1A_h} + U_{2_h} * Vu_{2_h}) * TF_{1,k} / (g * J)$$

$$\psi_h = 2. * g * J * \left[\Delta ht_h / \quad (U_{1A_h})^{2.} + (U_{2_h})^{2.}\right]$$

$$\eta tt_h = \Delta ht_h / \Delta ht_{I_{1,k}}$$

$$\eta ts_h = \Delta ht_h / \Delta hs_{I_{1,k}}$$

$$\eta tat_h = \Delta ht_h / \Delta hat_{I_{1,k}}$$

$$Zincr_h = \cos(90 - 2\beta_{2E_h}) * \left[(\tan\beta_{1A_h} / \tan\beta_{2E_h}) + 1.\right]$$

$$Cpr_h = 1. - (R_{1A_h} / R_{2_h})^{2.}$$

$$Vu_{2A_h} = Vu_{2A_{1,k}} * Dp_{2A_{1,k}} / Dr_{2A_k}$$

$$V_{2A_h} = \sqrt{(Vu_{2A_h})^{2.} + (Vz_{2A_{1,k}})^{2.}}$$

$$\alpha_{2A_h} = \tan^{-1}(Vu_{2A_h} / Vz_{2A_{1,k}})$$

$$\Delta Ts_{2A_h} = (V_{2A_{1,k}})^{2.} - \left[(Vz_{2A_{1,k}})^{2.} + (Vu_{2A_h})^{2.}\right] \Big/ (2. * g * J * Cp_{2A_k})$$

$$Ts_{2A_h} = Ts_{2A_{1,k}} + \Delta Ts_{2A_h}$$

$$Ps_{2A_h} = Ps_{2A_{1,k}} * \left(1. + \Delta Ts_{2A_h} / Ts_{2A_{1,k}}\right)^{\frac{\gamma_{2A_k}}{\gamma_{2A_k} - 1.}}$$

$$\rho s_{2A_h} = 144. * Ps_{2A_h} / (RG * Ts_{2A_h})$$

$$M_{2A_h} = V_{2A_h} / \sqrt{\gamma_{2A_k} * g * RG * Ts_{2A_h}}$$

$$MF_{2A_h} = M_{2A_h} * \cos\alpha_{2A_h}$$

Calculate tip values by repeating from (10) with t replacing h and Isect replacing 1.

8. CALL Wout

9. End of k loop

RETURN

APPENDIX 1W

WΘUT - WRITE INTERSTAGE OUTPUT

SUBROUTINE WΘUT

The purpose of subroutine WOUT is to write the interstage output.

Print out interstage data.
RETURN

APPENDIX 1X

DIAGT - DIAGNOSTIC

SUBROUTINE DIAGT

The purpose of subroutine DIAGT is to write a brief diagnostic when an error is encountered. When TRDIAG is input positive, a diagnostic is recovered at the end of the main program from each station.

Test M

If M = 0

Go to STA0 diagnostic (10). If M = 1

Go to STA0 diagnostic (10). If M = 2

Go to STA1 diagnostic (19). If M = 3

Go to STA1A diagnostic (11). If M = 4

Go to STA2 diagnostic (12). If M = 5

Go to STA2A diagnostic (13).

10. STA0 diagnostic

If M = 0

Go to STA1 diagnostic (19)

19. STA1 diagnostic

If M = 0

Go to STA1A diagnostic (11)

11. STA1A diagnostic

If M = 0

Go to STA2 diagnostic (12)

12. STA2 diagnostic

If M = 0

Go to STA2A diagnostic (13)

13. STA2A diagnostic

RETURN

APPENDIX 1Y

PHIM - PHI MAXIMUM

SUBROUTINE PHIM

The purpose of subroutine PHIM is to determine the pressure ratio at which the flow function is a maximum including the efficiency of expansion $_s$ or $_r$.

SUBROUTINE PHIM (EXI, ETA, TR, PR)

where

EXI = function of specific heat ratio, $\frac{\gamma}{\gamma-1}$.

ETA = expansion efficiency, $_s$ or $_r$.

TR = isentropic temperature ratio, Ø.

PR = pressure ratio, $\text{Ø}^{\frac{\gamma}{\gamma-1}}$

$$A = EXI - \frac{1}{2}$$

$$B = - EXI + (1. - ETA)/2.$$

$$C = ETA/2.$$

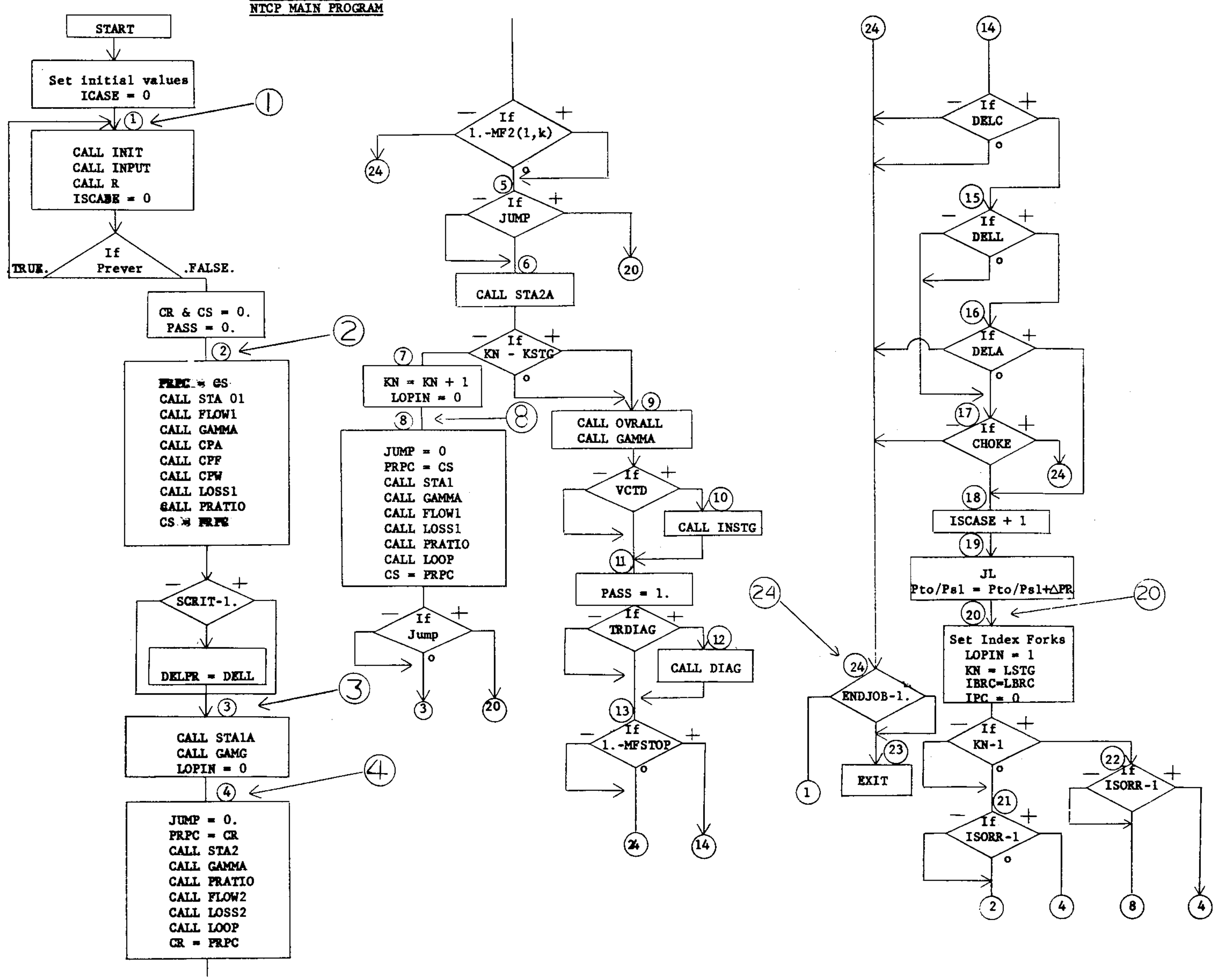
$$X = (-B - \sqrt{B^2 - 4AC}) / (2A)$$

$$TR = ETA/(ETA - X)$$

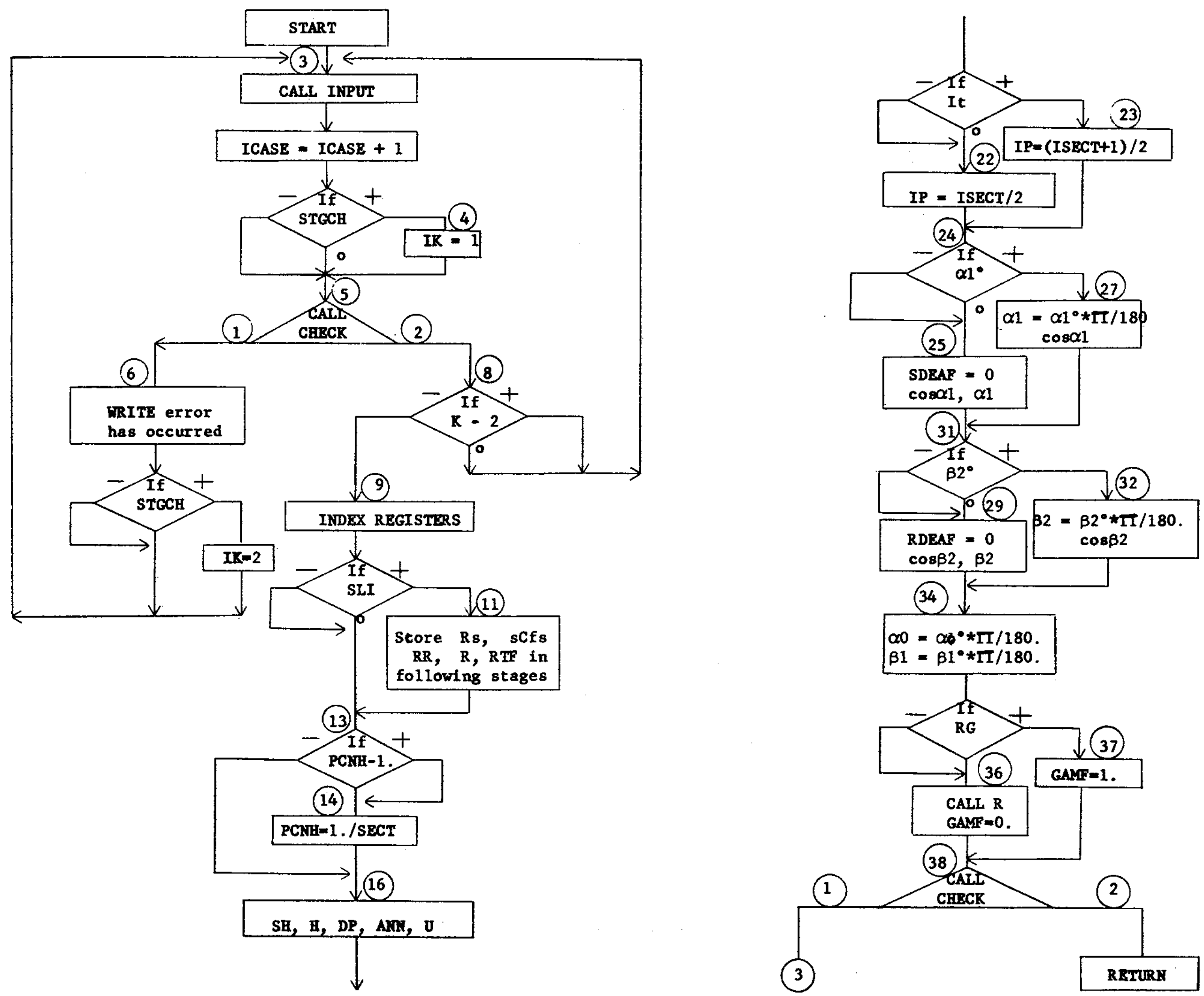
$$PR = (TR)^{EXI}$$

RETURN

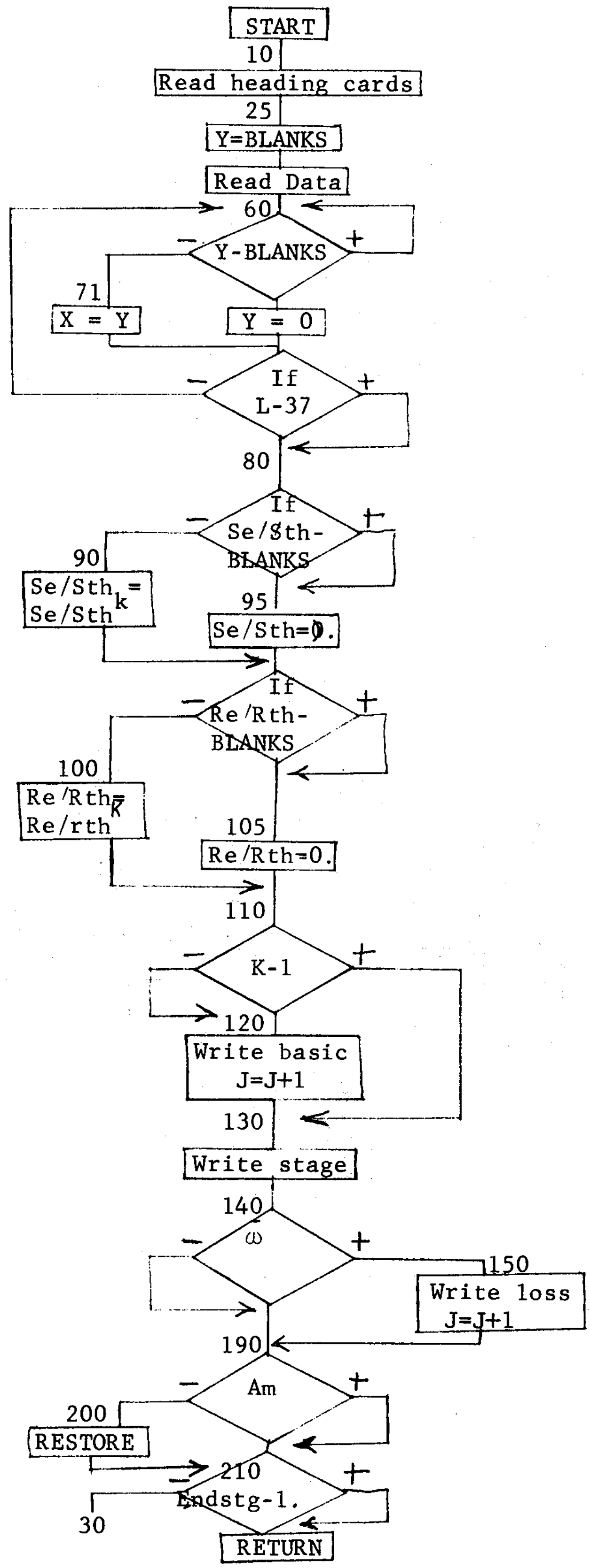
NTCP MAIN PROGRAM



INIT

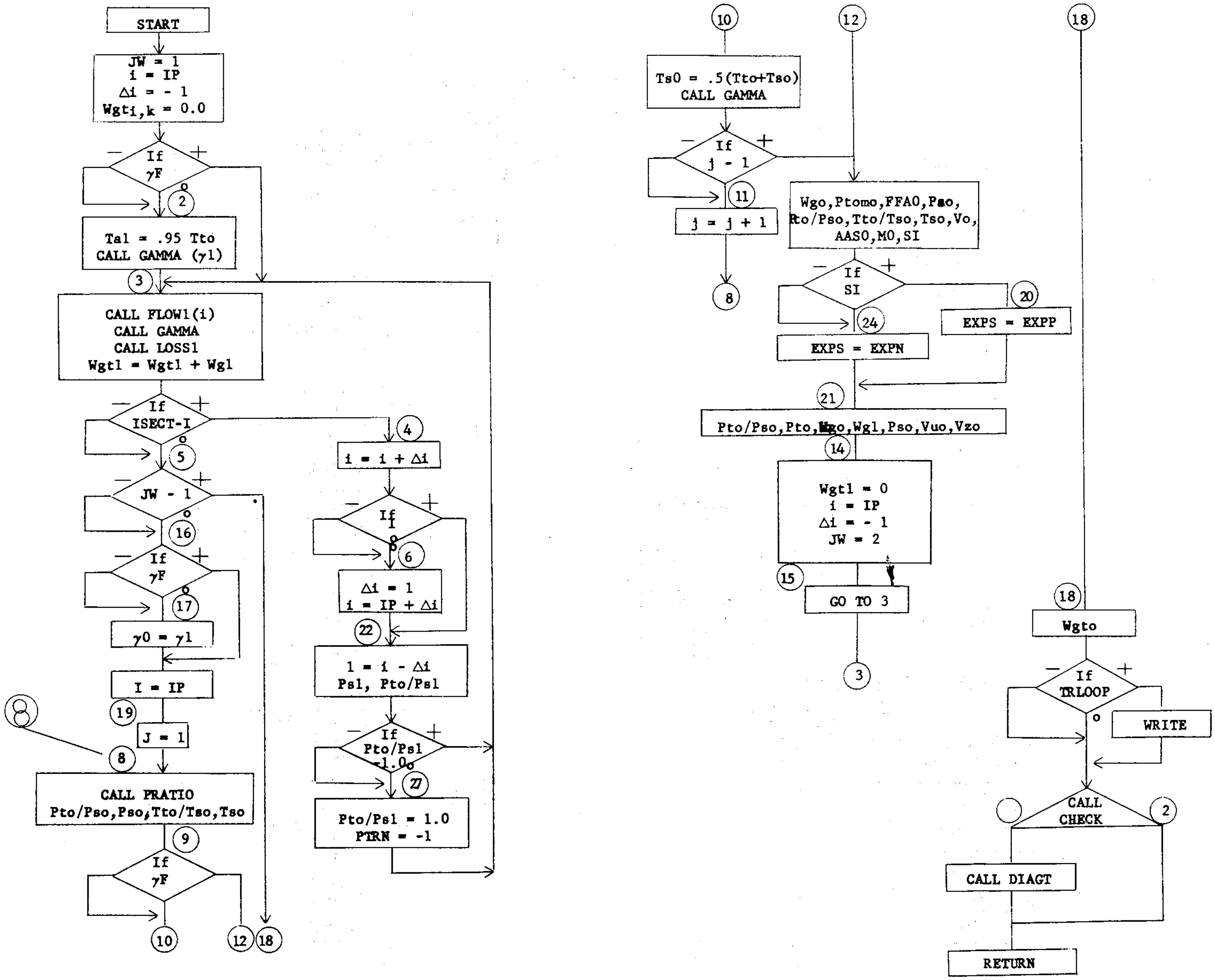

START
CALL INPUT
ICASE = ICASE + 1
If STGCH
IK = 1
CALL CHECK
WRITE error has occurred
If STGCH
IK=2
If K - 2
INDEX REGISTERS
If SLI
Store Rs, sCfs RR, R, RTF in following stages
If PCNH-1.
PCNH=1./SECT
SH, H, DP, ANN, U
If It
IP=(ISECT+1)/2
IP = ISECT/2
If α1°
α1 = α1°*π/180 cosα1
SDEAF = 0 cosα1, α1
If β2°
β2 = β2°*π/180. cosβ2
RDEAF = 0 cosβ2, β2
α0 = α0°*π/180. β1 = β1°*π/180.
If RG
GAMF=1.
CALL R GAMF=0.
CALL CHECK
RETURN

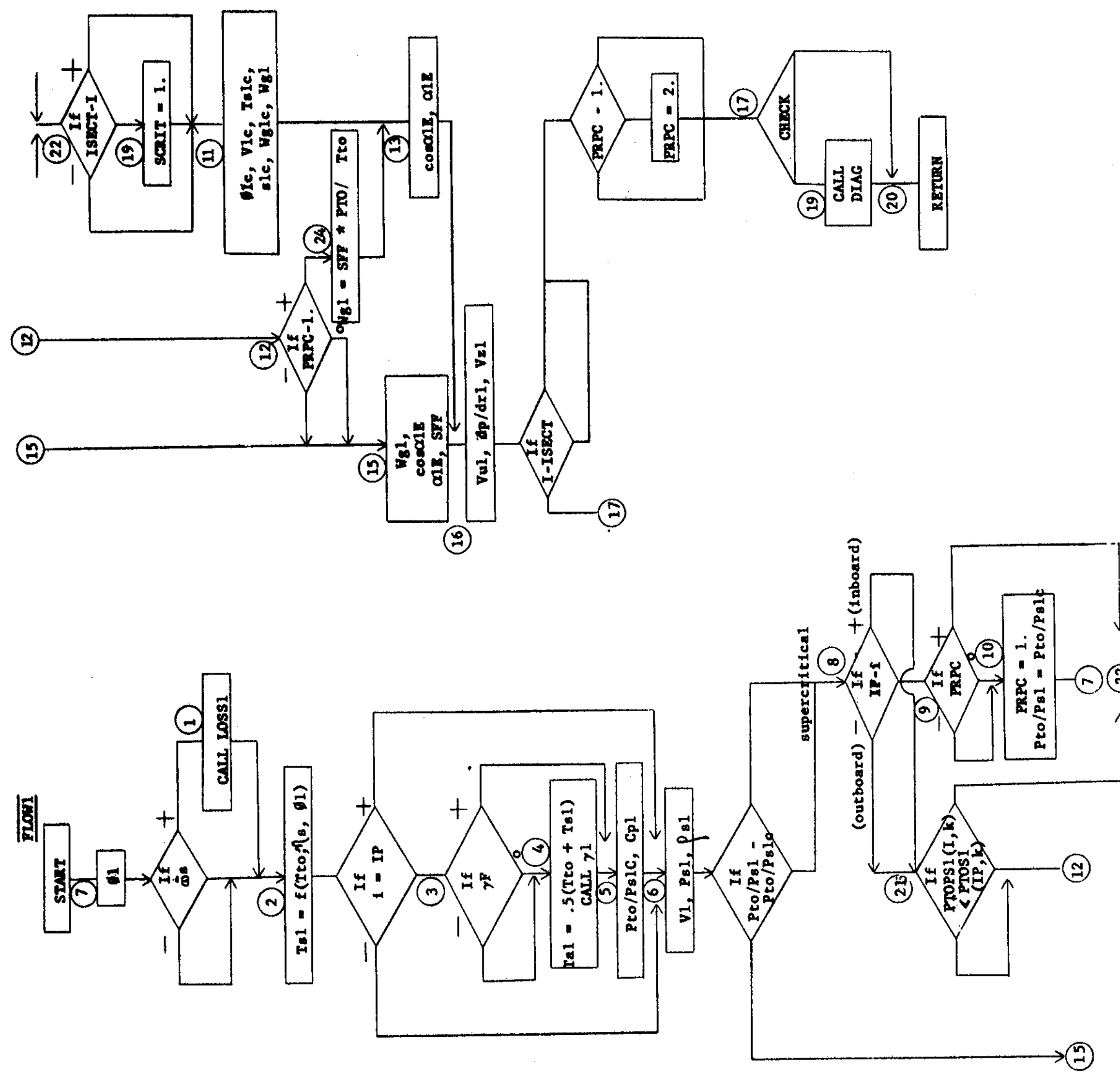
INPUT



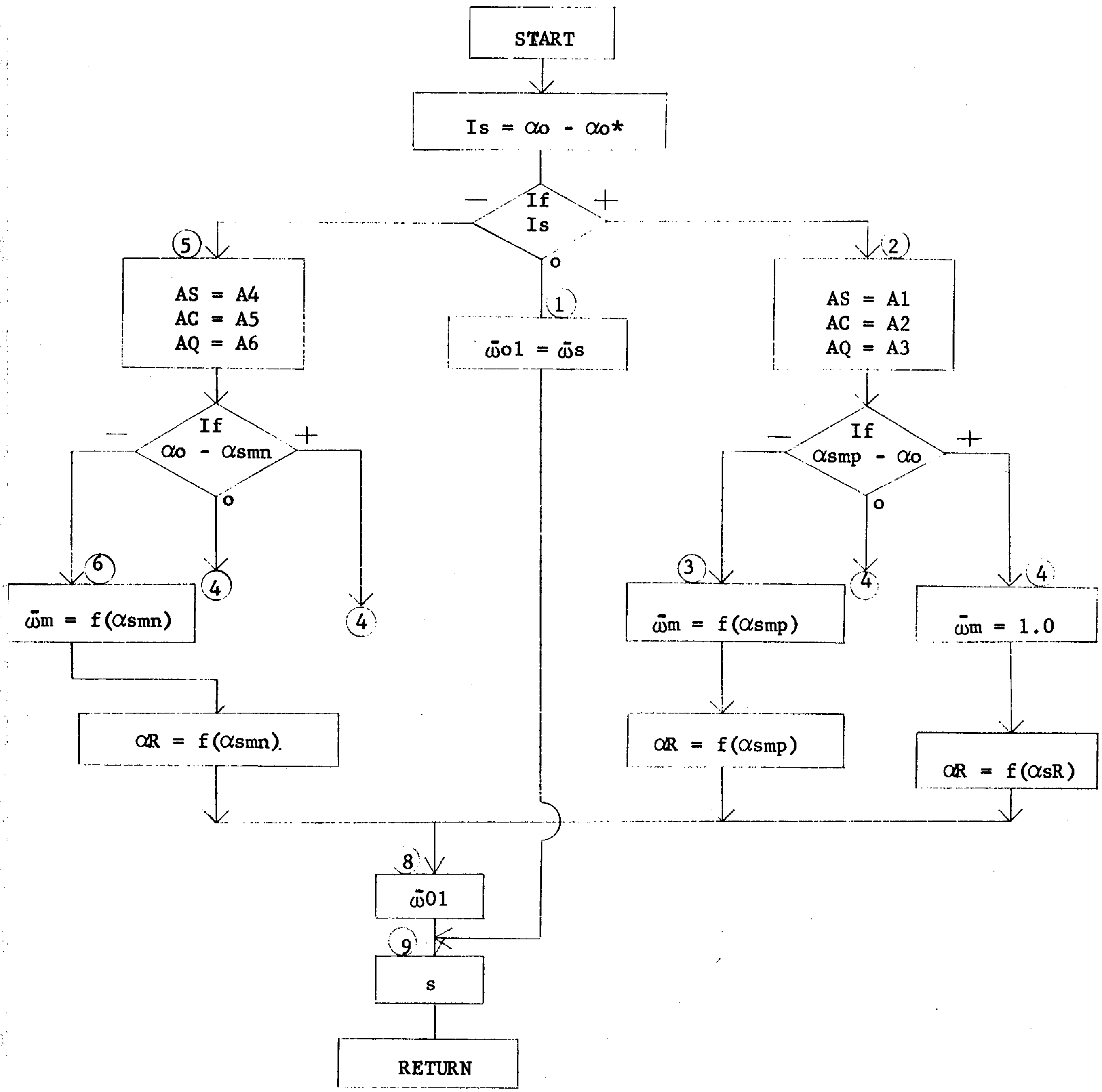
START
10
Read heading cards
25
Y=BLANKS
Read Data
60
Y-BLANKS
71
X = Y
Y = 0
If
L-37
80
If
Se/Sth-
BLANKS
90
Se/Sth$_k$ =
Se/Sth
95
Se/Sth=0.
If
Re/Rth-
BLANKS
100
Re/Rth$_K$=
Re/rth
105
Re/Rth=0.
110
K-1
120
Write basic
J=J+1
130
Write stage
140
ω
150
Write loss
J=J+1
190
Am
200
RESTORE
210
Endstg-1.
30
RETURN

STA 01


START
JW = 1
i = IP
Δi = - 1
Wgti,k = 0.0
If γF
2
Tal = .95 Tto
CALL GAMMA (γ1)
3
CALL FLOW1(i)
CALL GAMMA
CALL LOSS1
Wgt1 = Wgt1 + Wg1
If ISECT-I
5
4
i = i + Δi
JW - 1
16
If γF
17
γ0 = γ1
I = IP
19
J = 1
8
CALL PRATIO
Pto/Pso,Pso,Tto/Tso,Tso
9
If γF
10
12
18
If I
6
Δi = 1
i = IP + Δi
22
1 = i - Δi
Ps1, Pto/Ps1
If Pto/Ps1 -1.0
27
Pto/Ps1 = 1.0
PTRN = -1
Ts0 = .5(Tto+Tso)
CALL GAMMA
If j - 1
11
j = j + 1
Wgo,Ptomo,FFA0,Pso,
Pto/Pso,Tto/Tso,Tso,Vo,
AASO,M0,SI
If SI
20
24
EXPS = EXPP
EXPS = EXPN
21
Pto/Pso,Pto,Wgo,Wg1,Pso,Vuo,Vzo
14
Wgt1 = 0
i = IP
Δi = - 1
JW = 2
15
GO TO 3
Wgto
If TRLOOP
WRITE
CALL CHECK
CALL DIAGT
RETURN

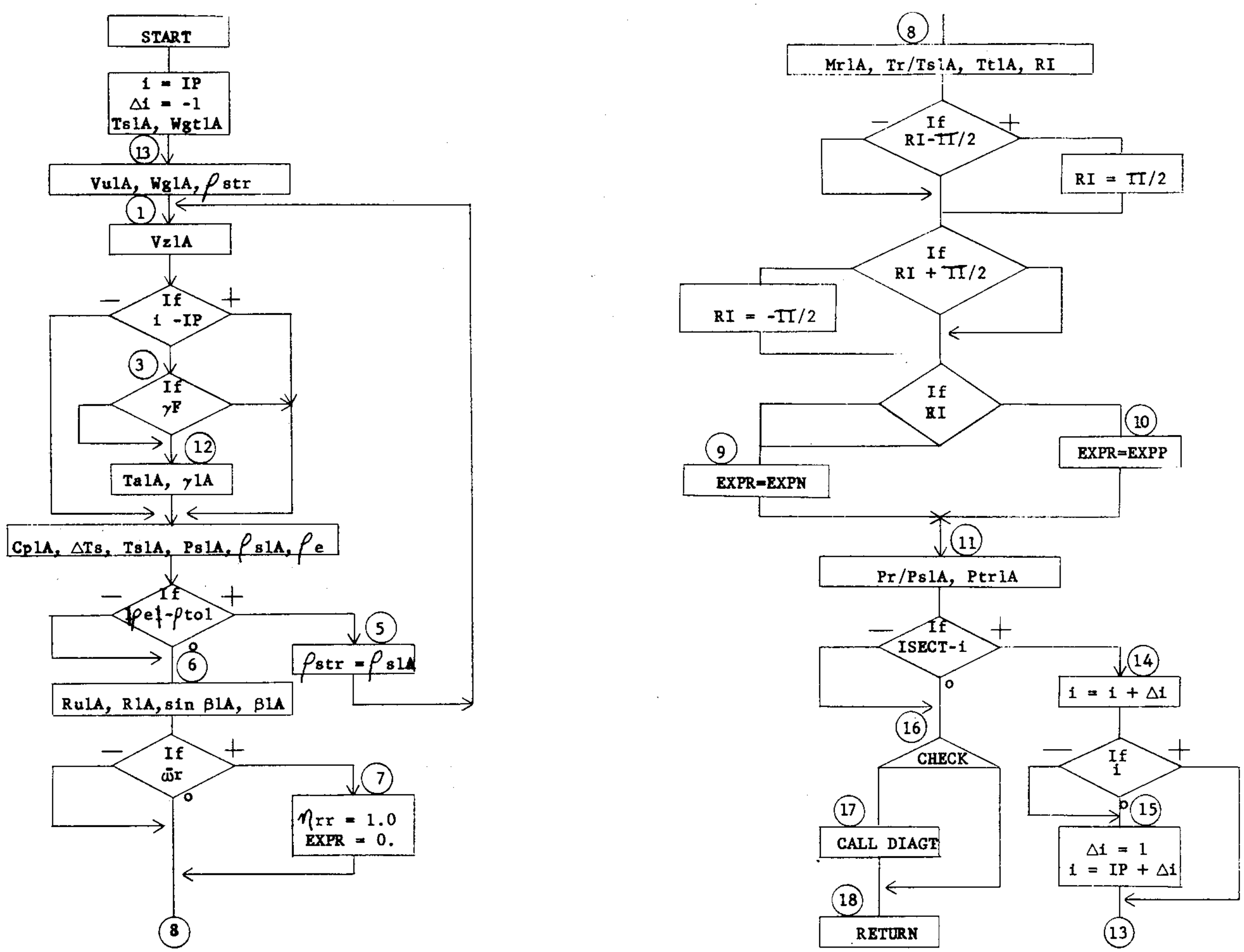

FLOW1
START
Ø1
If ωs
CALL LOSS1
Ts1 = f(Tto, s, Ø1)
If i = IP
If γF
Ts1 = .5(Tto + Ts1) CALL γ1
Pto/Ps1C, Cp1
V1, Ps1, ρs1
If Pto/Ps1 - Pto/Ps1c
supercritical
If IP-f
+(inboard)
(outboard) -
If FRPC
FRPC = 1.
Pto/Ps1 = Pto/Ps1c
If PTOPS1(I,K) < PTOS1(IP,K)
If ISECT-I
SCRIT = 1.
Ølc, V1c, Ts1c, s1c, Wg1c, Wg1
Wg1 = SFF * PTO/ Tto
cosα1E, α1E
If FRPC-1.
Wg1, cosα1E α1E, SFF
Vu1, ∂p/dr1, Vz1
If I-ISECT
FRPC - 1.
FRPC = 2.
CHECK
CALL DIAG
RETURN

TCP LOSS1

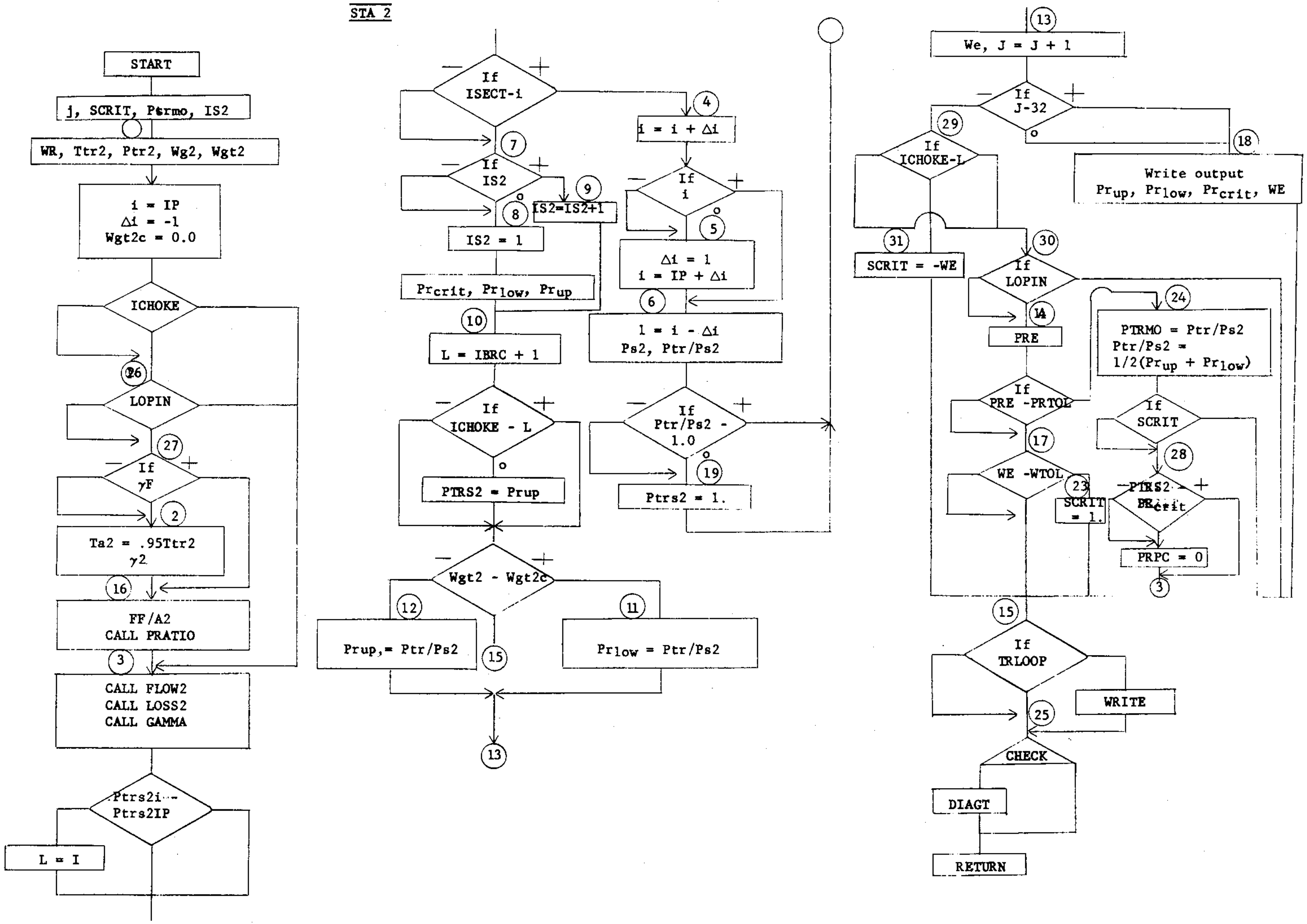


START
Is = αo - αo*
If Is
—
+
o
5
AS = A4
AC = A5
AQ = A6
If αo - αsmn
6
ω̄m = f(αsmn)
αR = f(αsmn)
4
1
ω̄o1 = ω̄s
2
AS = A1
AC = A2
AQ = A3
If αsmp - αo
3
ω̄m = f(αsmp)
αR = f(αsmp)
ω̄m = 1.0
αR = f(αsR)
8
ω̄01
9
s
RETURN

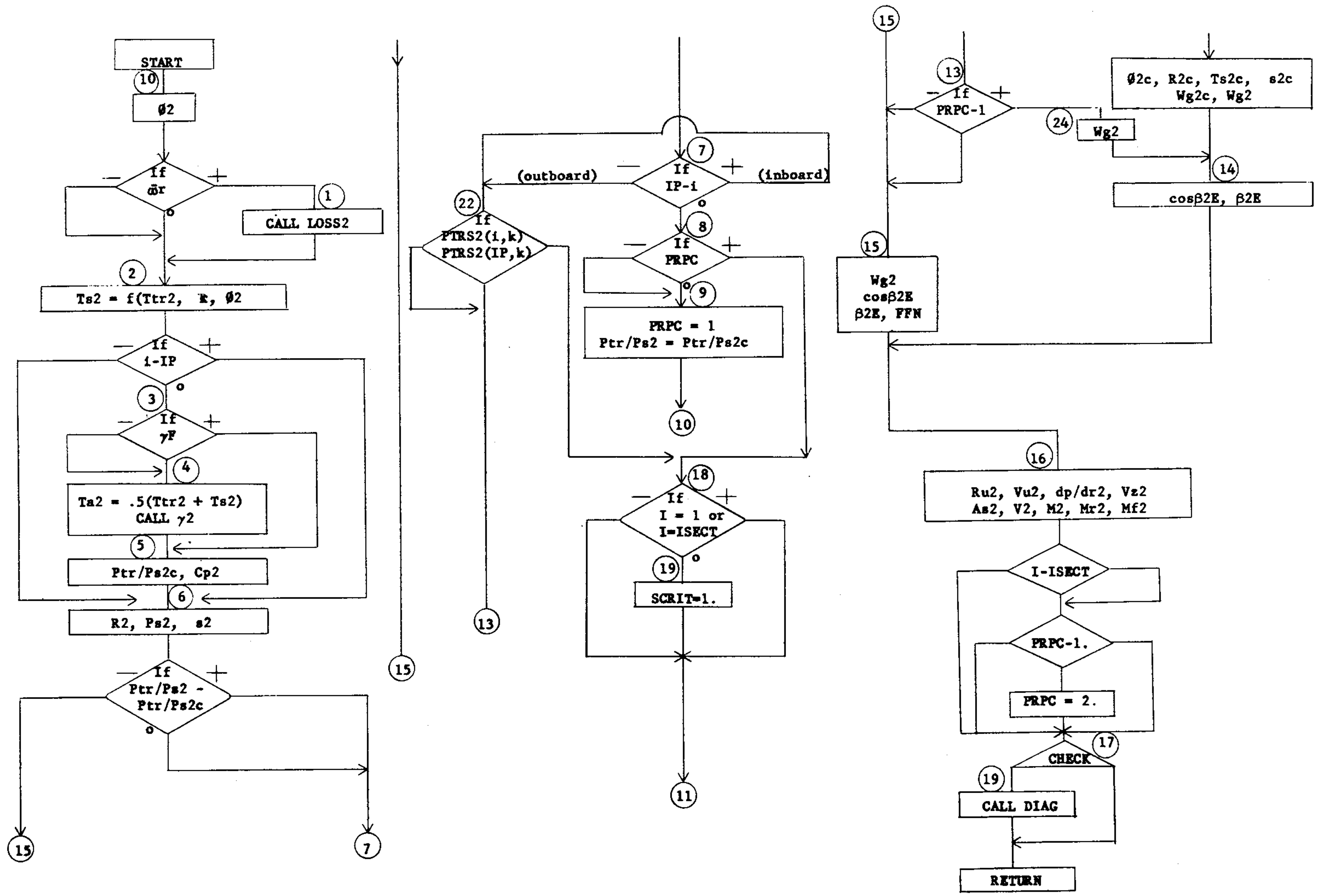
STA 1A


START
i = IP
Δi = -1
Ts1A, Wgt1A
13
Vu1A, Wg1A, ρstr
1
Vz1A
If
i -IP
3
If
γF
12
Ta1A, γ1A
Cp1A, ΔTs, Ts1A, Ps1A, ρs1A, ρe
If
|ρe|-ρtol
5
ρstr = ρs1A
6
Ru1A, R1A, sin β1A, β1A
If
ω̄r
7
ηrr = 1.0
EXPR = 0.
8
8
Mr1A, Tr/Ts1A, Tt1A, RI
If
RI-π/2
RI = π/2
If
RI + π/2
RI = -π/2
If
RI
9
EXPR=EXPN
10
EXPR=EXPP
11
Pr/Ps1A, Ptr1A
If
ISECT-i
14
i = i + Δi
16
CHECK
If
i
17
CALL DIAGT
15
Δi = 1
i = IP + Δi
18
RETURN
13

STA 2

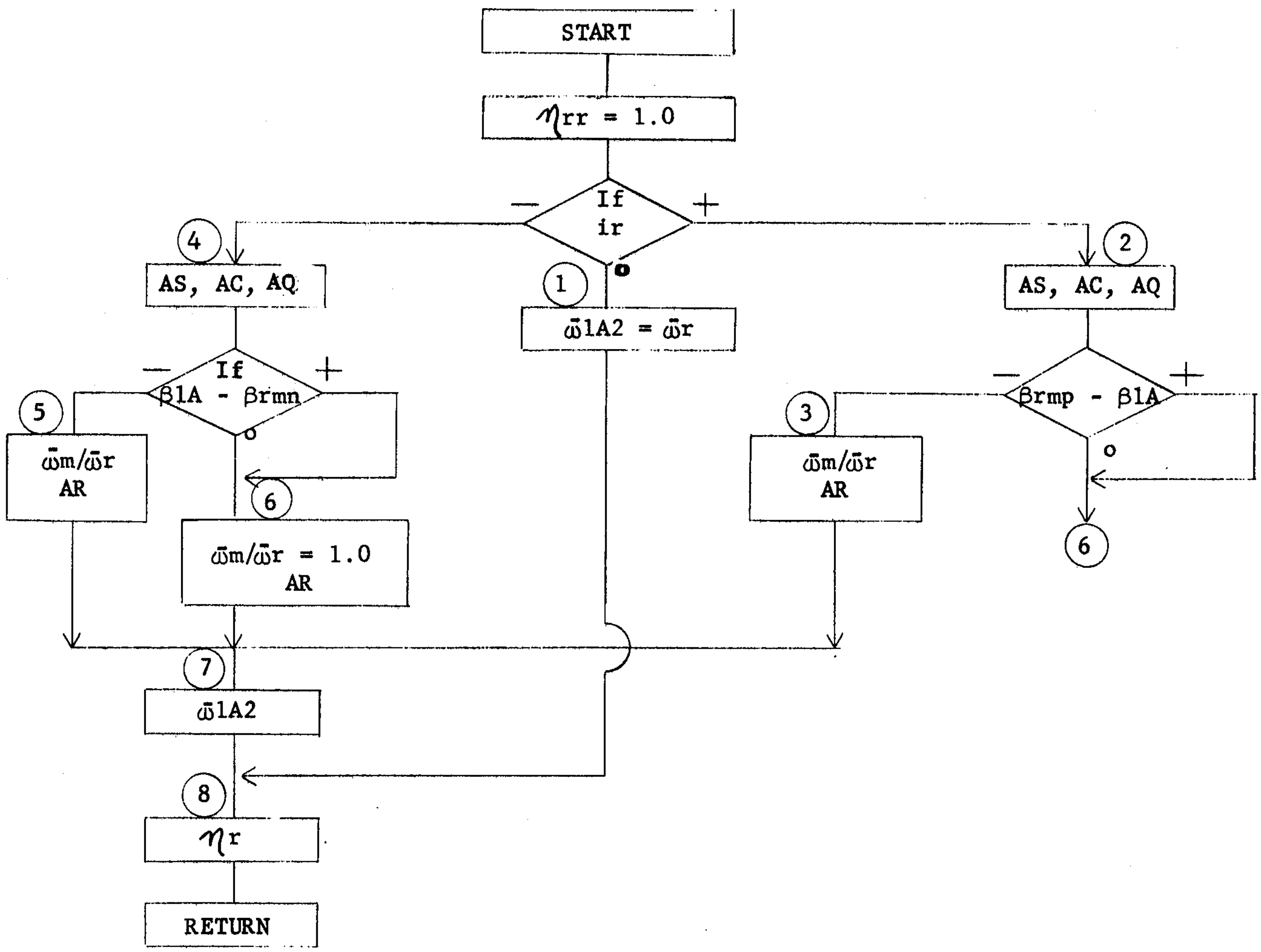

START
j, SCRIT, Ptrmo, IS2
WR, Ttr2, Ptr2, Wg2, Wgt2
i = IP
Δi = -1
Wgt2c = 0.0
ICHOKE
26
LOPIN
27
If
γF
2
Ta2 = .95Ttr2
γ2
16
FF/A2
CALL PRATIO
3
CALL FLOW2
CALL LOSS2
CALL GAMMA
Ptrs2i -
Ptrs2IP
L = I
If
ISECT-i
7
If
IS2
8
9
IS2=IS2+1
IS2 = 1
Prcrit, Prlow, Prup
10
L = IBRC + 1
If
ICHOKE - L
PTRS2 = Prup
Wgt2 - Wgt2c
12
Prup,= Ptr/Ps2
15
11
Prlow = Ptr/Ps2
13
4
i = i + Δi
If
i
5
Δi = 1
i = IP + Δi
6
i = i - Δi
Ps2, Ptr/Ps2
If
Ptr/Ps2 -
1.0
19
Ptrs2 = 1.
13
We, J = J + 1
If
J-32
29
If
ICHOKE-L
18
Write output
Prup, Prlow, Prcrit, WE
31
SCRIT = -WE
30
If
LOPIN
14
PRE
24
PTRMO = Ptr/Ps2
Ptr/Ps2 =
1/2(Prup + Prlow)
If
PRE -PRTOL
If
SCRIT
28
17
WE -WTOL
23
SCRIT
= 1.
PTRS2 -
PRcrit
PRPC = 0
3
15
If
TRLOOP
WRITE
25
CHECK
DIAGT
RETURN

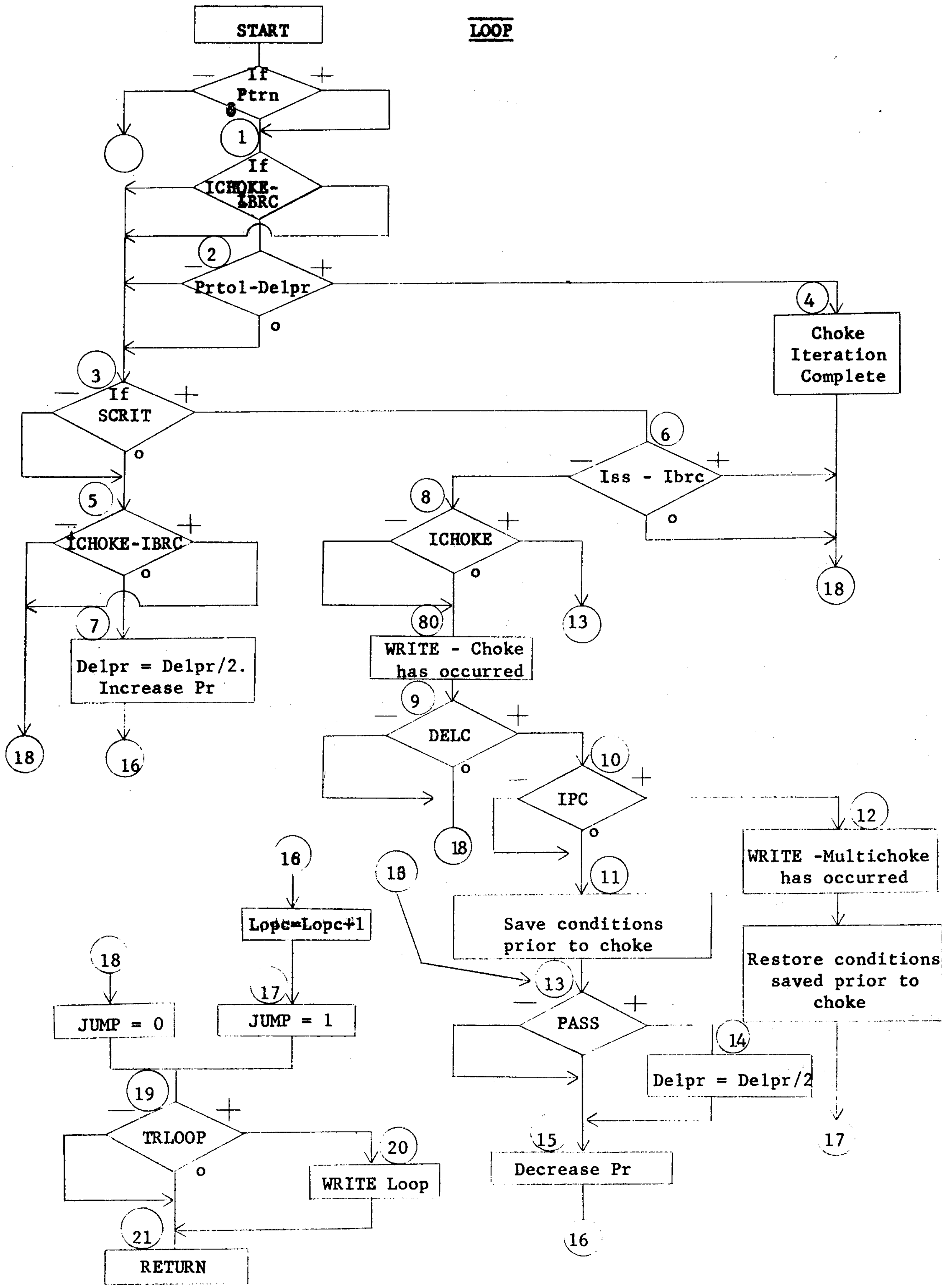
FLOW2



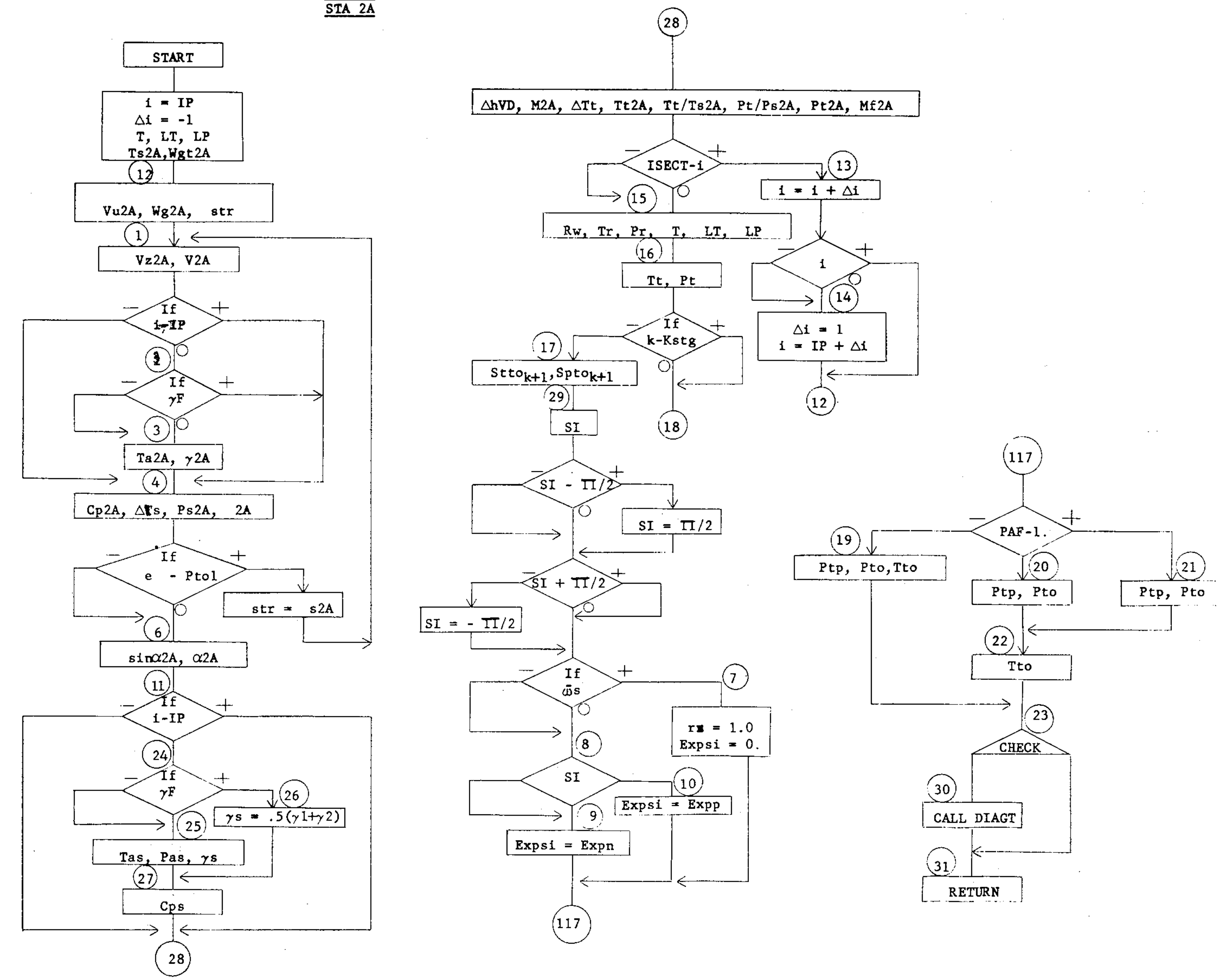
START
Ø2
If
ωr
CALL LOSS2
Ts2 = f(Ttr2, k, Ø2
If
i-IP
If
γF
Ta2 = .5(Ttr2 + Ts2)
CALL γ2
Ptr/Ps2c, Cp2
R2, Ps2, s2
If
Ptr/Ps2 -
Ptr/Ps2c
If
PTRS2(i,k)
PTRS2(IP,k)
(outboard)
If
IP-i
(inboard)
If
PRPC
PRPC = 1
Ptr/Ps2 = Ptr/Ps2c
If
I = 1 or
I=ISECT
SCRIT=1.
If
PRPC-1
Wg2
Ø2c, R2c, Ts2c, s2c
Wg2c, Wg2
cosβ2E, β2E
Wg2
cosβ2E
β2E, FFN
Ru2, Vu2, dp/dr2, Vz2
As2, V2, M2, Mr2, Mf2
I-ISECT
PRPC-1.
PRPC = 2.
CHECK
CALL DIAG
RETURN

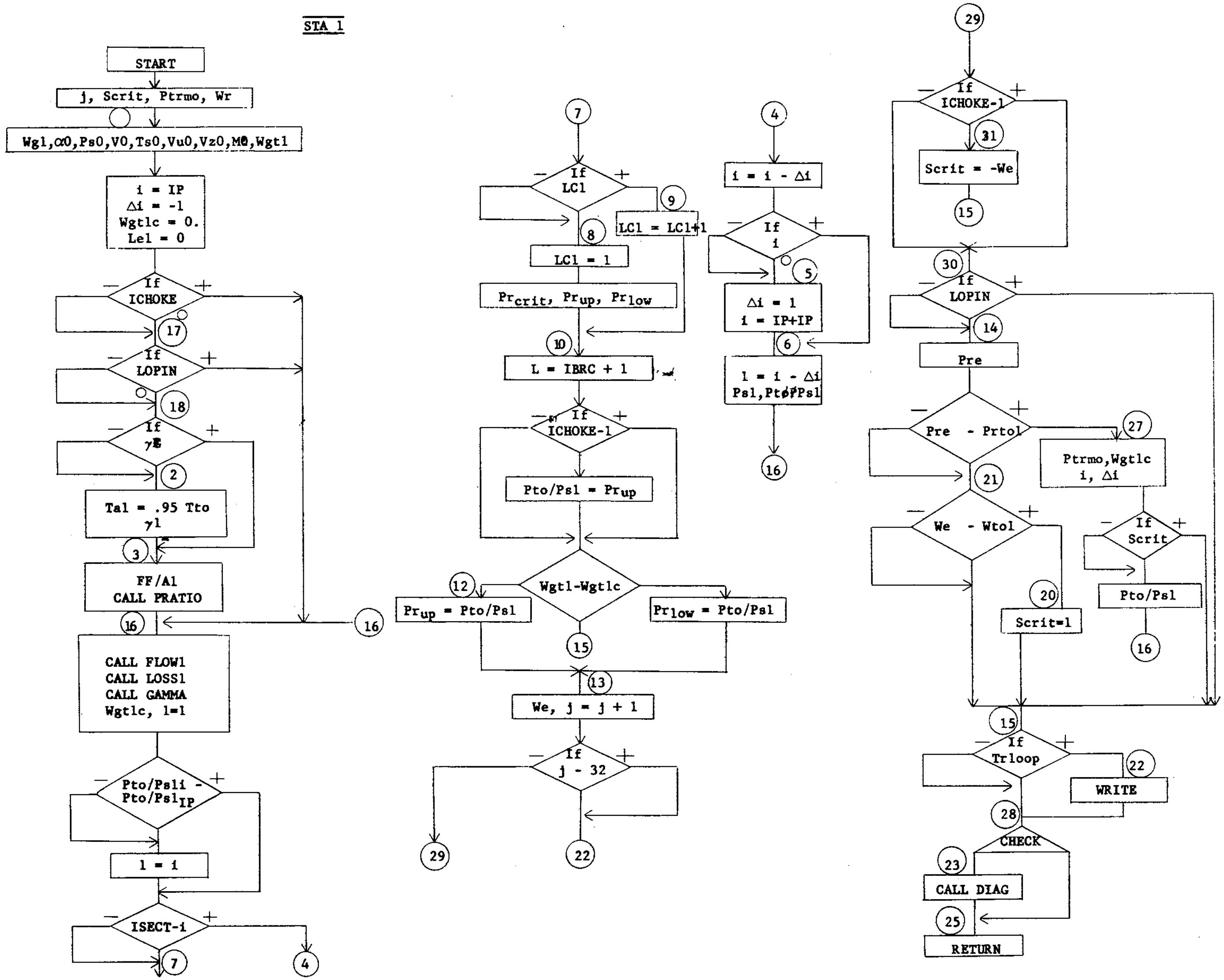
LOSS2


START
ηrr = 1.0
If
ir
—
+
o
4
AS, AC, AQ
1
$\bar{\omega}$1A2 = $\bar{\omega}$r
2
AS, AC, AQ
If
β1A - βrmn
—
+
o
5
$\bar{\omega}$m/$\bar{\omega}$r
AR
6
$\bar{\omega}$m/$\bar{\omega}$r = 1.0
AR
3
$\bar{\omega}$m/$\bar{\omega}$r
AR
βrmp - β1A
—
+
o
6
7
$\bar{\omega}$1A2
8
ηr
RETURN

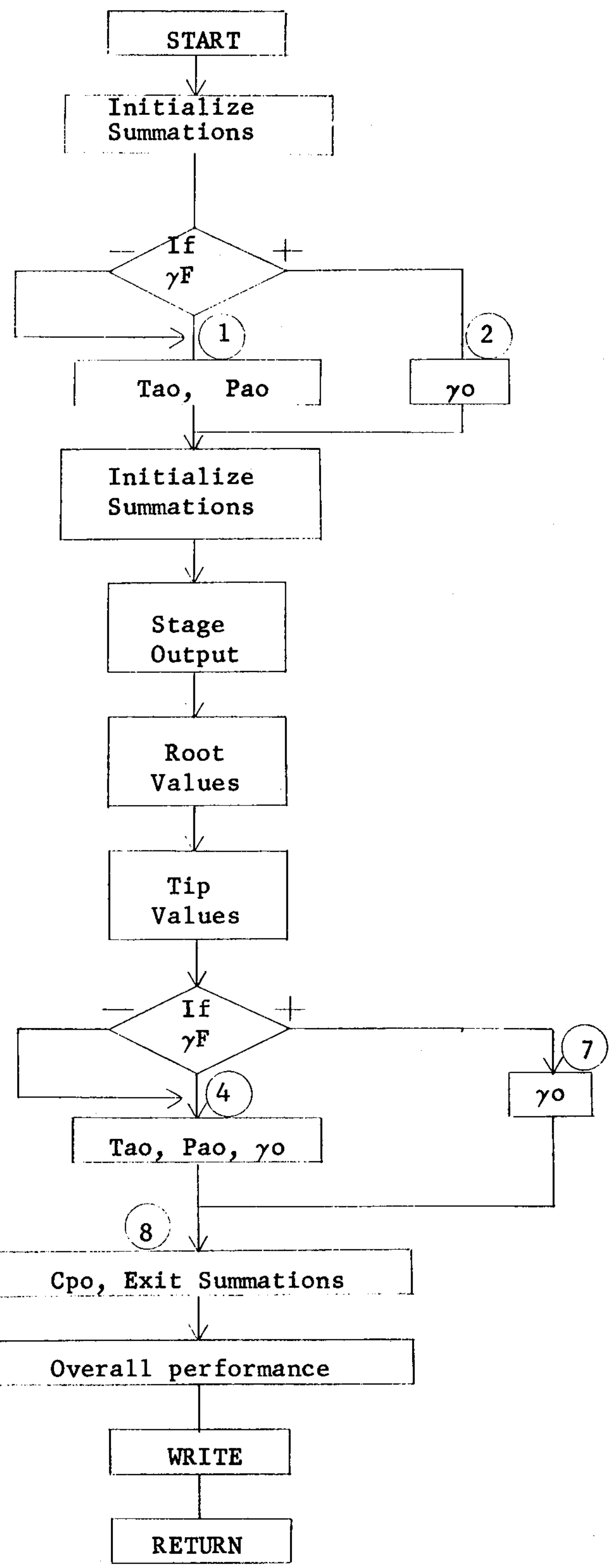

LOOP
START
If Ptrn
1
If ICHOKE-IBRC
2
Prtol-Delpr
3
If SCRIT
4
Choke Iteration Complete
5
ICHOKE-IBRC
6
Iss - Ibrc
7
Delpr = Delpr/2.
Increase Pr
8
ICHOKE
80
WRITE - Choke has occurred
9
DELC
10
IPC
11
Save conditions prior to choke
12
WRITE -Multichoke has occurred
Restore conditions saved prior to choke
13
PASS
14
Delpr = Delpr/2
15
Decrease Pr
16
Lopc=Lopc+1
17
JUMP = 1
18
JUMP = 0
19
TRLOOP
20
WRITE Loop
21
RETURN

STA 2A

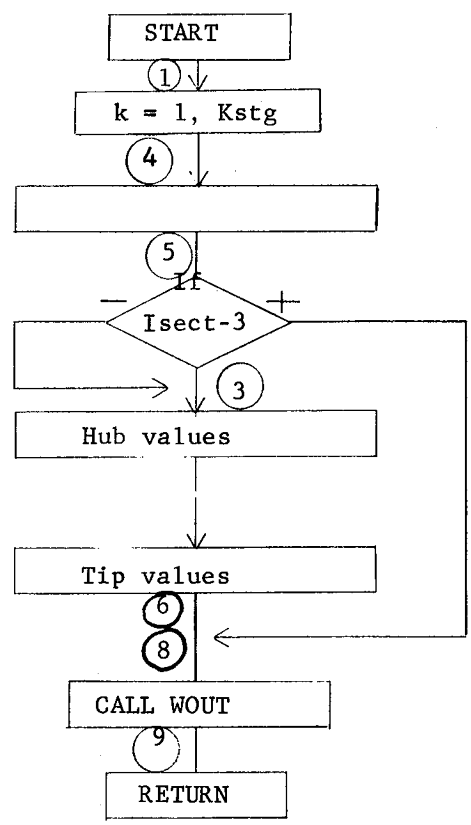

START
i = IP
Δi = -1
T, LT, LP
Ts2A,Wgt2A
12
Vu2A, Wg2A, str
1
Vz2A, V2A
If
i-IP
2
If
γF
3
Ta2A, γ2A
4
Cp2A, ΔTs, Ps2A, 2A
If
e - Ptol
str = s2A
6
sinα2A, α2A
11
If
i-IP
24
If
γF
26
γs = .5(γ1+γ2)
25
Tas, Pas, γs
27
Cps
28
28
ΔhVD, M2A, ΔTt, Tt2A, Tt/Ts2A, Pt/Ps2A, Pt2A, Mf2A
ISECT-i
13
i = i + Δi
15
Rw, Tr, Pr, T, LT, LP
16
Tt, Pt
i
14
Δi = 1
i = IP + Δi
12
If
k-Kstg
17
Sttok+1,Sptok+1
29
SI
18
SI - π/2
SI = π/2
SI + π/2
SI = - π/2
If
ω̄s
7
rs = 1.0
Expsi = 0.
8
SI
10
Expsi = Expp
9
Expsi = Expn
117
117
PAF-1.
19
Ptp, Pto,Tto
20
Ptp, Pto
21
Ptp, Pto
22
Tto
23
CHECK
30
CALL DIAGT
31
RETURN


STA 1
START
j, Scrit, Ptrmo, Wr
Wg1,α0,Ps0,V0,Ts0,Vu0,Vz0,M0,Wgt1
i = IP
Δi = -1
Wgtlc = 0.
Lel = 0
If ICHOKE
If LOPIN
If γE
Ta1 = .95 Tto γ1
FF/A1
CALL PRATIO
CALL FLOW1
CALL LOSS1
CALL GAMMA
Wgtlc, l=1
Pto/Psli - Pto/Psl IP
l = i
ISECT-i
If LC1
LC1 = LC1+1
LC1 = 1
Pr crit, Pr up, Pr low
L = IBRC + 1
If ICHOKE-1
Pto/Psl = Pr up
Wgt1-Wgtlc
Pr up = Pto/Psl
Pr low = Pto/Psl
We, j = j + 1
If j - 32
i = i - Δi
If i
Δi = 1
i = IP+IP
l = i - Δi
Psl,Pto/Psl
If ICHOKE-1
Scrit = -We
If LOPIN
Pre
Pre - Prtol
Ptrmo,Wgtlc
i, Δi
We - Wtol
If Scrit
Pto/Psl
Scrit=1
If Trloop
WRITE
CHECK
CALL DIAG
RETURN

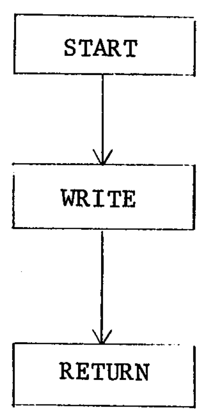
OVRALL


START
Initialize Summations
If γF
—
+
1
2
Tao, Pao
γo
Initialize Summations
Stage Output
Root Values
Tip Values
If γF
—
+
4
7
Tao, Pao, γo
γo
8
Cpo, Exit Summations
Overall performance
WRITE
RETURN

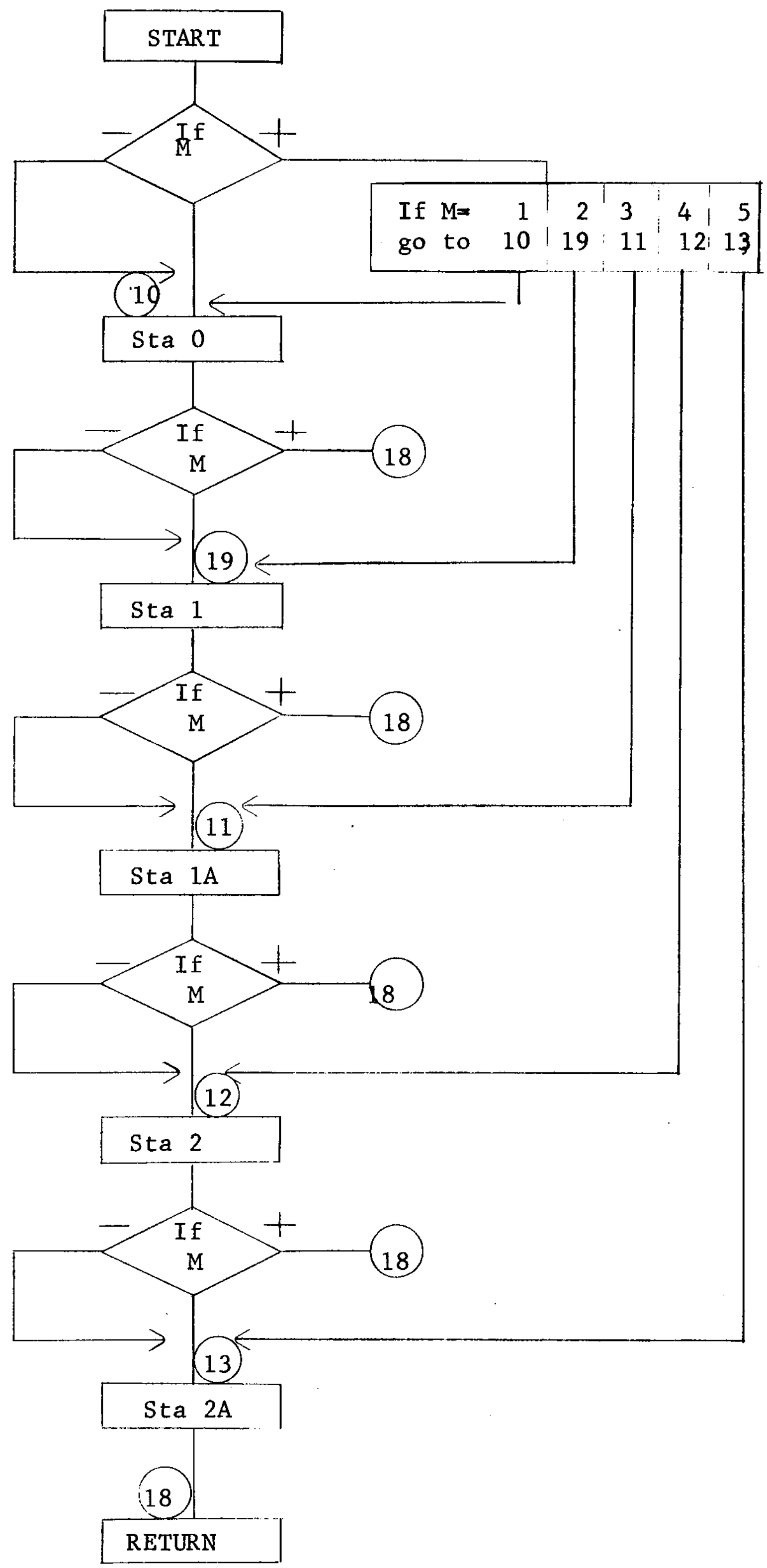
<u>INSTG</u>


START
1
k = 1, Kstg
4
5
If
Isect-3
—
+
3
Hub values
Tip values
6
8
CALL WOUT
9
RETURN

WOUT



DIAGT


START
If M
If M= 1 2 3 4 5
go to 10 19 11 12 13
10
Sta 0
If M
18
19
Sta 1
If M
18
11
Sta 1A
If M
18
12
Sta 2
If M
18
13
Sta 2A
18
RETURN

APPENDIX 3A

```
$IBFTC NTCP     FULIST,DECK,SDD
CNTCP
C      NASA TURBINE PROGRAM
C
       REAL MFSTOP
       LOGICAL PREVER
       COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
       COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6,8)
     6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
       REAL MR2,M2     ,MF2
       COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),RU2(6
     1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)
C
       DIMENSION CS(8),CR(8)
C
C
       CALL SLITE(0)
       WAIR=0.0
       FAIR=0.0
       PTPS=1.02
       DELC=0.0
       DELL=0.0
       DELA=0.0
       EXPN=2.0
       EXPP=2.0
       EXPRE=0.0
       RG=0.0
       PAF=0.0
       SLI=0.0
       AACS=1.0
       SECT=1.0
       VCTD=0.0
       WTOL=1.E-04
       RHOTOL=1.E-04
       PRTOL=1.E-06
       PCNH(1)=1.0
       GAM(1,1)=0.0
       RWG(1,1)=1.0
       ETAS(1,1)=0.0
       ALPHA1(1,1)=0.0
```

```
      ETAR(1,1)=0.0
      BETA2(1,1)=0.0
      TRLOOP=0.
      TRDIAG=0.0
      G=32.17405
      AJ=778.161
      ICASE=0
    1 PREVER=.FALSE.
      CALL INIT
      ISCASE=0
      IF (PREVER) GO TO 1
      DO 25 I=1,8
      CS(I)=0.0
   25 CR(I)=0.0
      PASS=0
    2 PRPC=CS(KN)
      CALL STA01
      IF (PREVER) GO TO 40
      IF(ICHOKE.NE.0)  GO TO 3
      IF(SCRIT.EQ.1.) SC=SC+1.
    3 CALL STA1A
      IF (PREVER) GO TO 40
      LOPIN=0
    4 JUMP=0
      PRPC=CR(KN)
      CALL STA2
      CR(KN)=PRPC
      IF (PREVER) GO TO 40
      IF (1.-MF2(1,KN))24,5,5
    5 IF (JUMP)6,6,20
    6 CALL STA2A
      IF (PREVER) GO TO 40
      IF (KN-KSTG)7,9,9
    7 KN=KN+1
      LOPIN=0
    8 JUMP=0
      PRPC=CS(KN)
      CALL STA1
      CS(KN)=PRPC
      IF (PREVER) GO TO 40
      IF (JUMP)3,3,20
    9 CALL OVRALL
      IF (VCTD)11,11,10
   10 CALL INSTG
   11 PASS=1.
      IF (TRDIAG)13,13,12
   12 CALL DIAGT(0)
   13 IF (1.-MFSTOP)24,24,14
   14 IF (DELC)24,24,15
   15 IF (DELL)17,17,16
   16 IF(DELPR)24,24,18
   17 IF (CHOKE)24,18,24
   18 ISCASE=ISCASE+1
```

```
 19 JL=(ISORR-1)*8+LSTG
    IF(SC.EQ.1.) DELPR=DELL
    PTOPS1(IP,JL)=PTOPS1(IP,JL)+DELPR
 20 LOPIN=1
    KN=LSTG
    IBRC=LBRC
    IPC=0
    IF (KN-1)21,21,22
 21 IF (ISORR-1)2,2,4
 22 IF (ISORR-1)8,8,4
 40 WRITE(6,106)
 24 IF (ENDJOB-1.)1,23,23
 23 CALL EXIT
106 FORMAT(//3X65HTHE PREVIOUS CASE HAS BEEN TERMINATED DUE TO ERRORS
   1- CHECK DUMP.)
    STOP
    END
```

APPENDIX 3B

```
$IBFTC INIT    FULIST,DECK,SDD
CINIT
C      SUBROUTINE FOR INITIALIZATION OF INPUT DATA
       SUBROUTINE INIT
C
       REAL MFSTOP
       LOGICAL PREVER
       COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
      1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
      2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
      3DELPR,PASS,IPC,LOPC,ISS
C
       COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
      1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
      2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
      3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
       COMMON /SINPUT/
      1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
      2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
      3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
      4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
      5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6,8)
      6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
      76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
      8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
       DIMENSION                           H1A(6,8),H0(6,8),H2A(6,8)
C
C      READ INPUT DATA, CHECK FOR ERRORS,
C            SKIP CHANGE CASES IF BASIC CASE HAS AN ERROR
     3 CALL INPUT
       ICASE=ICASE+1
       IF(STGCH)5,5,4
     4 IK=1
     5 CALL CHECK(L)
       GO TO(6,8),L
     6 WRITE(6,100)ICASE
       IF(STGCH)3,3,7
     7 IK=2
       GO TO 3
     8 IF (IK-2)9,3,3
C          INITIALIZE INDEX REGISTERS AND FORKS
     9 ISECT=SECT+.0001
       KSTG= STG+.0001
       LOPC=0
       CHOKE=0.
       ICHOKE=0
       ISORR=1
       KN=1
       LSTG=1
       IBRC=1
```

```
      LBRC=1
      DELPR=DELC
      SC=0.0
      RC=0.0
      PRPC=0.0
      IPC=0
      ISS=0
      PTRN=0.0
C        TEST STAGE LOSS INDICATOR
      IF(SLI)13,13,11
   11 DO 12 I=1,ISECT
      DO 12 J=1,KSTG
      ETARS(I,J)=ETARS(I,1)
      ETAS(I,J)=ETAS(I,1)
      CFS(I,J)=CFS(I,1)
      ETARR(I,J)=ETARR(I,1)
      ETAR(I,J)=ETAR(I,1)
      CFR(I,J)=CFR(I,1)
      TFR(I,J)=TFR(I,1)
   12 CONTINUE
C        TEST FOR EQUAL SECTORS
   13 IF(PCNH(1)-1.)16,14,14
   14 DO 15 I=1,ISECT
   15 PCNH(I)=      1./SECT
C        SET UP SECTOR HEIGHT, PITCH DIAMETER, ANNULUS AREA,
C             PITCHLINE WHEEL SPEED
   16 DO 19 K=1,KSTG
         SH0=DT(1,K)-DR(1,K)
         SH1=DT(2,K)-DR(2,K)
         SH1A=DT(3,K)-DR(3,K)
         SH2=DT(4,K)-DR(4,K)
         SH2A=DT(5,K)-DR(5,K)
      DO 18 I=1,ISECT
      H0(I,K)=.5*PCNH(I)*SH0
      H1(I,K)=.5*PCNH(I)*SH1
      H1A(I,K)=.5*PCNH(I)*SH1A
      H2(I,K)=.5*PCNH(I)*SH2
      H2A(I,K)=.5*PCNH(I)*SH2A
      IF(I-1)20,20,17
   20 DP0(I,K)=DR(1,K)+   H0(I,K)
      DP1(I,K)=DR(2,K)+   H1(I,K)
      DP1A(I,K)=DR(3,K)+   H1A(I,K)
      DP2(I,K)=DR(4,K)+   H2(I,K)
      DP2A(I,K)=DR(5,K)+   H2A(I,K)
      GO TO 21
   17 DP0(I,K)=         H0(I-1,K)+   H0(I,K)+DP0(I-1,K)
      DP1(I,K)=         H1(I-1,K)+   H1(I,K)+DP1(I-1,K)
      DP1A(I,K)=         H1A(I-1,K)+   H1A(I,K)+DP1A(I-1,K)
      DP2(I,K)=         H2(I-1,K)+   H2(I,K)+DP2(I-1,K)
      DP2A(I,K)=         H2A(I-1,K)+   H2A(I,K) +DP2A(I-1,K)
   21 ANN0(I,K)=.02182*DP0(I,K)*H0(I,K)
      ANN1(I,K)=.02182*DP1(I,K)*H1(I,K)
      ANN1A(I,K)=DP1A(I,K)*H1A(I,K)*.02182
```

```
      ANN2(I,K)=.02182*DP2(I,K)*H2(I,K)
      ANN2A(I,K)=.02182*DP2A(I,K)*H2A(I,K)
      U1A(I,K)= 3.14159*DP1A(I,K)*RPM/720.
      U2(I,K)= 3.14159*DP2(I,K)*RPM/720.
   18 CONTINUE
   19 CONTINUE
C        DEFINE PITCHLINE INDEX
      IT=ISECT-2*(ISECT/2)
      IF(IT)22,22,23
   22 IP=ISECT/2
      GO TO 24
   23 IP=(ISECT+1)/2
C        CALCULATE INLET AND EXIT ANGLES IN RADIANS
   24 IF (ALPHA1(1,1))25,25,27
   25 SDEAF=0.
      DO 26 K=1,KSTG
      DO 26 I=1,ISECT
      CSALF1(I,K)=ANDO(I,K)*CFS(I,K)/(SESTHI(K)*3.14159*DP1(I,K)*
     1SQRT(ETAS(I,K)))
   26 ALF1(I,K)=ATAN2(SQRT(1.-CSALF1(I,K)*CSALF1(I,K)),CSALF1(I,K))
      GO TO 31
   27 DO 28 K=1,KSTG
      DO 28 I=1,ISECT
        ALF1(I,K)=     ALPHA1(I,K)*.017453
   28 CSALF1(I,K)=COS(ALF1(I,K))
   31 IF (BETA2(1,1))29,29,32
   29 RDEAF=0.
      DO 30 K=1,KSTG
      DO 30 I=1,ISECT
      CSBET2(I,K)=ANDOR(I,K)*CFR(I,K)/(RERTHI(K)*3.14159*DP2(I,K)*
     1SQRT(ETAR(I,K)))
   30 BET2(I,K)=ATAN2( SQRT(1.-CSBET2(I,K)*CSBET2(I,K)),CSBET2(I,K))
      GO TO 34
   32 DO 33 K=1,KSTG
      DO 33 I=1,ISECT
        BET2(I,K)=     BETA2(I,K)*.017453
   33 CSBET2(I,K)=COS(BET2(I,K))
   34 DO 35 K=1,KSTG
      DO 35 I=1,ISECT
      PTP(I,K)=PTIN
      PTO(I,K)=PTIN
      TTO(I,K)=TTIN
      ALPHAO(I,K)=0.0
      PTOPS1(I,K)=PTPS
      RADSD(I,K)=ALPHAS(I,K)*.017453
   35 RADRD(I,K)=BETA1(I,K)*.017453
      IF(RG)36,36,37
   36 CALL R(PTIN,TTIN,FAIR,WAIR,RG)
      GAMF=0.0
      GO TO 38
   37 GAMF=1.0
   38 CALL CHECK(J)
      GO TO (39,40),J
```

```
 39 GO TO 3
 40 RETURN
100 FORMAT(28X,6HCASE  I5,13H HAS AN ERROR)
    END
```

APPENDIX 3C

```
$IBFTC INPT     FULIST,DECK,SDD
CINPUT
      SUBROUTINE INPUT
C***********************************************************************
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),SR(6,8),ST(6,8),SWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6,8)
     6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      DIMENSION X(6,8,37),Y(6,37)
C
      EQUIVALENCE (X(1,1,1),GAM(1,1)),(Y(1,1),GAMG(1))
C
      COMMON GAMG(6),DR(6),DT(6),RWG(6),SDIA(6),SDEA(6),SREC(6),SETA(6),
     1SCF(6),SPA(6),RDIA(6),RDEA(6),RREC(6),RETA(6),RCF(6),RTF(6),RPA(6)
     2,STPLC(6),SINR(6),SINMP(6),SINMN(6),SCPS(6),SCPC(6),SCPQ(6),SCNS(6
     3),SCNC(6),SCNQ(6),RTPLC(6),RINR(6),RINMP(6),RINMN(6),RCPS(6),RCPC(
     46),RCPQ(6),RCNS(6),RCNC(6),RCNQ(6)
C
      NAMELIST/DATAIN/ PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD
     1,STG,SECT,STAGE,EXPN,EXPP,EXPRE,RG,RPM,PAF,SLI,ENDSTG,ENDJOB,PCNH,
     2GAMG,DR,DT,RWG,SDIA,SDEA,SREC,SETA,SCF,SPA,RDIA,RDEA,RREC,RETA,RCF
     3,RTF,RPA,STPLC,SINR,SINMP,SIMMN,SCPS,SCPC,SCPQ,SCNS,SCNC,SCNQ,RTPL
     4C,RINR,RINMP,RINMN,RCPS,RCPC,RCPQ,RCNS,RCNC,RCNQ,SESTH,RERTH,
     5WTOL,RHOTOL,PRTOL,TRLOOP,TRDIAG,STGCH
C
      DATA BLANKS/66666666/
C
C
C     READ THE HEADING CARDS EVERY TIME ENTRY IS MADE
   10 READ(5,6669) (NAME(I),I=1,10)
   20 READ(5,6669) (TITLE(I),I=1,10)
      J=0
   30 DO 25 L=1,37
      DO 25 I=1,6
   25 Y(I,L)=BLANKS
      SESTH=BLANKS
      RERTH=BLANKS
      READ(5,DATAIN)
```

```
   40 K=STAGE+.0001                                                      INPT 052
   50 ISECT=SECT+.0001                                                   INPT 053
   60 DO 80 L=1,37                                                       INPT 054
   70 DO 80 I=1,6                                                        INPT 055
      IF (Y(I,L).NE.BLANKS)  GO TO 71                                    INPT 056
      Y(I,L)=0.0                                                         INPT 057
      GO TO 80                                                           INPT 058
   71 X(I,K,L)=Y(I,L)                                                    INPT 059
   80 CONTINUE                                                           INPT 060
      IF(SESTH.EQ.BLANKS) GO TO 95                                       INPT 061
   90 SESTHI(K)=SESTH                                                    INPT 062
      GO TO 96                                                           INPT 063
   95 SESTH=0.                                                           INPT 064
   96 IF(RERTH.EQ.BLANKS) GO TO 105                                      INPT 065
  100 RERTHI(K)=RERTH                                                    INPT 066
      GO TO 110                                                          INPT 067
  105 RERTH=0.                                                           INPT 068
  110 IF (K-1)120,120,130                                                INPT 069
  120 WRITE(6,6670)NAME,TITLE,TTIN,PTIN,WAIR,FAIR,PTPS,DELC,DELL,DELA,   INPT 070
     1STG,SECT,EXPN,EXPP,RG,PAF,SLI,AACS,RPM,VCTD,EXPRE,ENDSTG,ENDJOB,   INPT 071
     1PCNH                                                               INPT 072
      J=J+1                                                              INPT 073
  130 WRITE(6,6671)    K,GAMG,DR,DT,RWG,SDIA,SDEA,SREC,SETA,SCF,SPA,     INPT 074
     1SESTH,                                                             INPT 075
     1RDIA,RDEA,RREC,RETA,RCF,RTF,RPA,RERTH                              INPT 076
  140 IF (OMEGAS(1,K))160,160,150                                        INPT 077
  150 WRITE(6,6672)STPLC,SINR,SINMP,SINMN,SCPS,SCPC,SCPQ,SCNS,SCNC,SCNQ, INPT 078
     1RTPLC,RINR,RINMP,RINMN,RCPS,RCPC,RCPQ,RCNS,RCNC,RCNQ               INPT 079
  160 J=J+1                                                              INPT 080
  180 AM= J-2*(J/2)                                                      INPT 081
  190 IF(AM)200,210,200                                                  INPT 082
  200 WRITE(6,6673)                                                      INPT 083
  210 IF (ENDSTG-1.)30,170,170                                           INPT 084
  170 RETURN                                                             INPT 085
 6669 FORMAT(10A6)                                                       INPT 086
 6670 FORMAT (1H1,24X,24HTURBINE COMPUTER PROGRAM/6X,10A6/6X,10A6/2X,    INPT 087
     17H*DATAIN/2X7H  TTIN=F10.3,1X,7H  PTIN=F10.3,1X,6H WAIR=F10.3,1X,  INPT 088
     25HFAIR=F10.3/2X,7H  PTPS=F10.3,1X,7H  DELC=F10.3,1X,6H DELL=F10.3, INPT 089
     31X,5HDELA=F10.3/2X,7H   STG=F10.3,1X,7H  SECT=F10.3,1X,6H EXPN=F10 INPT 090
     4.3,1X,5HEXPP=F10.3/2X,7H    RG=F10.3,1X,7H   PAF=F10.3,1X,6H  SLI= INPT 091
     5F10.3,1X,5HAACS=F10.3/2X,7H   RPM=F10.3,1X,7H  VCTD=F10.3,1X,6HEXP INPT 092
     6RE=F10.3/2X,7HENDSTG=F10.3,1X,7HENDJOB=F10.3,1X/25X,21HINLET RADIA INPT 093
     7L PROFILES/2X,5HPCNH=6(F8.3,2X)/1H1)                               INPT 094
 6671 FORMAT(28X,15HSTANDARD OPTION/3X,6HSTAGE=I3,21X,14HAXIAL STATIONS/ INPT 095
     110X,6HSTA. 04X,6HSTA. 14X,6HSTA.1A4X,6HSTA. 23X,7HSTA. 2A/         INPT 096
     33X,6H GAMG=6(F8.3,2X)/3X,6H   DR=6(F8.3,2X)/3X,6H   DT=6(F8.3,2X)/ INPT 097
     33X,6H  RWG=6(F8.3,2X)//22X,27HSTATOR RADIAL DISTRIBUTIONS/         INPT 098
     411X,4HROOT,5X,5HPITCH,6X,3HTIP,8X,19H(FOR THREE SECTORS)/          INPT 099
     53X,6H SDIA=6(F8.3,2X)/3X,6H SDEA=6(F8.3,2X)/3X,6H SREC=6(F8.3,2X)/ INPT 100
     63X,6H SETA=6(F8.3,2X)/3X,6H  SCF=6(F8.3,2X)/3X,6H  SPA=6(F8.3,2X)/ INPT 101
     73X,6HSESTH=F8.3//23X,26HROTOR RADIAL DISTRIBUTIONS/                INPT 102
     83X,6H RDIA=6(F8.3,2X)/3X,6H RDEA=6(F8.3,2X)/3X,6H RREC=6(F8.3,2X)/ INPT 103
     93X,6H RETA=6(F8.3,2X)/3X,6H  RCF=6(F8.3,2X)/3X,6H  RTF=6(F8.3,2X)/ INPT 104
```

```
     13X,6H  RPA=6(F8.3,2X)/3X,6HRERTH=1F8.3/)
6672 FORMAT(/25X,23HLOSS COEFFICIENT OPTION/22X,27HSTATOR RADIAL DISTRI
     1BUTIONS/
     23X,6HSTPLC=6(F8.3,2X)/3X,6H SINR=6(F8.3,2X)/3X,6HSINMP=6(F8.3,2X)/
     33X,6HSINMN=6(F8.3,2X)/3X,6H SCPS=6(F8.3,2X)/3X,6H SCPC=6(F8.3,2X)/
     43X,6H SCPQ=6(F8.3,2X)/3X,6H SCNS=6(F8.3,2X)/3X,6H SCNC=6(F8.3,2X)/
     53X,6H SCNQ=6(F8.3,2X)/023X,26HROTOR RADIAL DISTRIBUTIONS/
     63X,6HRTPLC=6(F8.3,2X)/3X,6H RINR=6(F8.3,2X)/3X,6HRINMP=6(F8.3,2X)/
     73X,6HRINMN=6(F8.3,2X)/3X,6H RCPS=6(F8.3,2X)/3X,6H RCPC=6(F8.3,2X)/
     83X,6H RCPQ=6(F8.3,2X)/3X,6H RCNS=6(F8.3,2X)/3X,6H RCNC=6(F8.3,2X)/
     93X,6H RCNQ=6(F8.3,2X))
6673 FORMAT (1H1)
      END
```

```
$IBFTC ST01     FULIST,DECK,SDD
CSTA01
C       ESTABLISH FIRST STATOR EXIT FLOW, ADJUST FLOWS FOR COOLING
C          AIR INJECTION BETWEEN STATIONS 0 AND 1, FIND INLET
C          MACH NUMBER AND INCIDENCE ANGLE LOSS AT STATION 0,
C          ADJUST PT, GET NEW FLOW AT STATION 1 FOR FINAL RESULT.
      SUBROUTINE STA01
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      REAL M0
      COMMON /SSTA01/CP0(8),                    PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2              DPDR1(6,8),SI(6,8),  CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)
C
      DIMENSION WGT0(8),TA0(8),WG0(6,8),TTOTS0(6,8),PTOPS0(6,8),FFA0(6,8
     1),AAS0(6,8)
C
C
      SCRIT=0.0
      I=IP
      ID=-1
      WGT1(K)=0.0
      JW=1
      IF(GAMF)2,2,3
    2 TA1(K)=.95*TT0(IP,K)
      CALL GAMMA(PTIN,TA1(K),FAIR,WAIR,GAM(2,K))
    3 CALL FLOW1(I)
      IF (PREVER) GO TO 26
      WGT1(K)=WGT1(K)+WG1(I,K)
C          TEST FOR TIP SECTOR
```

```
      IF(ISECT-I)5,5,4
    4 I=I+ID
      IF(I)6,6,22
   22 L=I-ID
      PS1(I,K)=PS1(    L,K)+FLOAT(ID)*DPDR1(    L,K)*(
     1H1(I,K)+H1(    L,K))/2.
      PTOPS1(I,K)=PTO(I,K)/PS1(I,K)
      IF (PTOPS1(I,K)-1.)27,3,3
   27 PTRN=-1.
      PTOPS1(I,K)= 1.0
      GO TO 3
    6 ID=1
      I=IP+ID
      GO TO 22
C         CALCULATE STA 0 FOR INCIDENCE CORRECTION
    5 IF (JW-1)16,16,18
   16 IF(GAMF)7,7,17
    7 GAM(1,K)=GAM(2,K)
   17 EX=(GAM(1,K)-1.)/GAM(1,K)
      EXI=1./EX
      WGTO(K)=WGT1(K)/RWG(2,K)
      I= IP
      WGO(I,K)=WG1(I,K)/RWG(2,K)
      FFAO(I,K)=WGO(I,K)*SQRT(    TTO(I,K))/(144.*PTO(I,K)*
     1ANNO(I,K))
   19 J=1
    8 CALL PRATIO(FFAO(I,K),GAM(1,K),RG,PTOPSO(I,K),PRTOL)
      PSO(I,K)=PTP(I,K)/PTOPSO(I,K)
      TTOTSO(I,K)=PTOPSO(I,K)**EX
      TSO(I,K)=TTO(I,K)/TTOTSO(I,K)
    9 IF(GAMF) 10,10,12
   10 TAO(K)=.5*(TTO(I,K)+TSO(I,K))
      CALL GAMMA(PTIN,TAO(K),FAIR,WAIR,GAM(1,K))
      EX=(GAM(1,K)-1.)/GAM(1,K)
      EXI=1./EX
      IF(J-1)11,11,12
   11 J=J+1
      GO TO 8
   12 CPO(K)=RG*EXI/AJ
      DO 14 I=1,ISECT
      WGO(I,K)=WG1(I,K)/RWG(2,K)
      PTOMO= PTO(I,K)
      FFAO(I,K)=WGO(I,K)*SQRT(    TTO(I,K))/(144.*PTO(I,K)*
     1ANNO(I,K))
      IF(I.EQ.IP) GO TO 28
      PSO(I,K) =  PSO(IP,K)
      PTOPSO(I,K) = PTP(I,K)/ PSO(I,K)
   28 TTOTSO(I,K)=PTOPSO(I,K)**EX
      TSO(I,K)=TTO(I,K)/TTOTSO(I,K)
   13 VO(I,K)=SQRT(2.*G*AJ*CPO(K)*(TTO(I,K)-TSO(I,K)))
      AASO(I,K)=SQRT(GAM(1,K)*G*RG*TSO(I,K))
      MO(I,K)=VO(I,K)/AASO(I,K)
      SI(I,K)=ALPHAO(I,K)- RADSD(I,K)
```

```
      IF(SI(I,K))24,24,20
   24 EXPS=EXPN
      GO TO 21
   20 EXPS=EXPP
   21 PTOPSO(I,K)=(1.+EX*MO(I,K)*ETARS(I,K)*GAM(1,K)*MO(I,K)/2.
     1*(COS(SI(I,K))**EXPS))**EXI
      PTO(I,K)=PSO(I,K)*PTOPSO(I,K)
      WGO(I,K)=WGO(I,K)*PTO(I,K)/PTOMO
      WG1(I,K)=WG1(I,K)*PTO(I,K)/PTOMO
      RHOSO(I,K)=144.*PSO(I,K)/(RG*TSO(I,K))
      VUO(I,K)=VO(I,K)*SIN(ALPHAO(I,K))
      VZO(I,K)=VO(I,K)*COS(ALPHAO(I,K))
   14 CONTINUE
C         END OF INCIDENCE LOSS CORRECTION LOOP
      WGT1(K)=0.
      I=IP
      ID=-1
      JW=2
   15 GO TO 3
   18 CONTINUE
      WGTO(K)=WGT1(K)/RWG(2,K)
      IF(TRLOOP.EQ.0.) GO TO 23
      WRITE(6,1000) WGTO(K),WGT1(K),(WGO(L,K),L=1,ISECT)
      WRITE(6,1001)                 (PTOPSO(L,K),L=1,ISECT)
      WRITE(6,1002)                    (WG1(L,K),L=1,ISECT)
      WRITE(6,1003)                 (PTOPS1(L,K),L=1,ISECT)
 1000 FORMAT(2X,6H WGTO=F8.3,2X,6H WGT1=F8.3/2X,6H  WGO=6F8.3)
 1001 FORMAT(1X,7HPTOPSO=6F8.5)
 1002 FORMAT(2X,6H  WG1=6F8.3)
 1003 FORMAT(1X,7HPTOPS1=6F8.5)
   23 CALL CHECK (J)
      GO TO (25,26),J
   25 CALL DIAGT(1)
   26 RETURN
      END
```

```
$IBFTC FLW1    FULIST,DECK,SDD
CFLOW1
C      ESTABLISH VALUES FOR STATOR EXIT FLOW
       SUBROUTINE FLOW1(I)
C
       REAL MFSTOP
       LOGICAL PREVER
       COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
       COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
       COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6,8)
     6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
       REAL MO
       COMMON /SSTA01/CP0(8),                       PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2             DPDR1(6,8),SI(6,8),  CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),MO(6,8)
C
       DIMENSION PHI1C(8),PTPS1C(8),V1C(6,8),TS1C(6,8),RHOS1C(6,8),WG1C(6
     1,8),CSAL1E(6,8),SFF(6,8)
C
C
       EX=(GAM(2,K)-1.)/GAM(2,K)
C         COMPUTE ISENTROPIC STATOR TEMPERATURE RATIO
     7 PHI1(I,K)=PTOPS1(I,K)**EX
C         TEST FOR LOSS COEFFICIENT INPUT
       IF (OMEGAS(1,1))2,2,1
     1 CALL LOSS1(I,K,EX)
     2 TS1(I,K)=TT0(I,K)*(1.-ETAS(I,K)*(1.-1./PHI1(I,K)))
       IF(I-IP)6,3,6
     3 IF(GAMF)4,4,5
     4 TA1(K)=.5*(TT0(I,K)+TS1(I,K))
       CALL GAMMA(PT0(IP,K),TA1(K),FAIR,WAIR,GAM(2,K))
     5 EX=(GAM(2,K)-1.0)/GAM(2,K)
       EXI=1./EX
C      CRITICAL PRESSURE RATIO
       CALL PHIM(EXI,ETAS(I,K),PHI1C(K),PTPS1C(K))
```

```
      CP1(K)=RG*EXI/AJ
C        EXIT VELOCITY
    6 V1(I,K)=SQRT(2.*G*AJ*CP1(K)*(TTO(I,K)-TS1(I,K)))
C        EXIT PRESSURE
      PS1(I,K)=PTO(I,K)/PTOPS1(I,K)
C        EXIT DENSITY
      RHOS1(I,K)=144.*PS1(I,K)/(RG*TS1(I,K))
C        TEST CRITICAL PRESSURE RATIO
      IF(PTOPS1(I,K)-PTPS1C(K))15, 8,8
C        GREATER THAN CRITICAL
    8 IF (IP-I) 21,9,21
    9 IF (PRPC)10,10,22
C        PREVIOUS PITCH NONCRITICAL
   10 PRPC=1.
      PTOPS1(I,K)=PTPS1C(K)*(1.+PRTOL)
      GO TO 7
   21 IF (PTOPS1(I,K).LE.PTOPS1(IP,K))  GO TO 22
      GO TO 12
   22 IF ((I.EQ.1).OR.(I.EQ.ISECT))  SCRIT=1.
      GO TO 11
C        PITCH OR OUTBOARD SECTOR
   11 CONTINUE
      V1C(I,K)=SQRT(2.*G*AJ*CP1(K)*TTO(I,K)*ETAS(I,K)*(PHI1C(K)
     1-1.)/PHI1C(K))
      TS1C(I,K)=TTO(I,K)*(1.-ETAS(I,K)*(1.-1./PHI1C(K)))
      RHOS1C(I,K)=144.*PTO(I,K)/(  PTPS1C(K)*TS1C(I,K)*RG)
      WG1C(I,K)=RHOS1C(I,K)*V1C(I,K)*ANN1(I,K)*CSALF1(I,K)
      WG1(I,K)=WG1C(I,K)
   13 CSAL1E(I,K)=WG1(I,K)/(RHOS1(I,K)*V1(I,K)*ANN1(I,K))
C     EFFECTIVE STATOR EXIT ANGLE
   14 ALF1E(I,K)=ATAN2(SQRT(1.-CSAL1E(I,K)*CSAL1E(I,K)),
     1CSAL1E(I,K))
      GO TO 16
   12 IF(   PRPC-1.)15,15,24
   24 WG1(I,K)=SFF(I,K)*PTO(I,K)/SQRT(TTO(I,K))
      GO TO 13
C        PRESSURE RATIO LESS THAN CRITICAL OR SUPERSONIC FLOW DECREASE
   15 WG1(I,K)=RHOS1(I,K)*V1(I,K)*ANN1(I,K)*CSALF1(I,K)
      CSAL1E(I,K)=CSALF1(I,K)
      ALF1E(I,K)=ALF1(I,K)
      SFF(I,K)=WG1(I,K)*SQRT(TTO(I,K))/PTO(I,K)
   16 VU1(I,K)=V1(I,K)*SIN(ALF1E(I,K))
      DPDR1(I,K)=.013889*RHOS1(I,K)*VU1(I,K)*VU1(I,K)/
     1(G*DP1(I,K))
      VZ1(I,K)=V1(I,K)*CSAL1E(I,K)
      IF(I.LT.ISECT) GO TO 17
      IF(PRPC.EQ.1.) PRPC=2.
   17 CALL CHECK(J)
      GO TO (19,20),J
   19 CALL DIAGT(2)
   20 RETURN
      END
```

APPENDIX 3F

```
$IBFTC LOS1     FULIST,DECK,SDD
CLOSS1
C
C      CALCULATE EFFICIENCY
       SUBROUTINE LOSS1(I,K,EX)
C
       REAL MFSTOP
       LOGICAL PREVER
       COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
      1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
      2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,F
      3DELPR,PASS,IPC,LOPC,ISS
C
       COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6
      1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,
      2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
      3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
       COMMON /SINPUT/
      1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN
      2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
      3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,
      4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),E
      5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6
      6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),
      76,8),OMEGAR(6,8),BS1A(6,8),BSMP1A(6,8),BCMN1A(6,8),B1(6,8),B2(6
      8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
       REAL MO
       COMMON /SSTA01/CP0(8),                         PS0(6,8),V0(6,8),TS0
      18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8
      2              DPDR1(6,8),SI(6,8),   CP1(8),PHI1(6,8),TS1(6,8),V1(6
      3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),MO(6,8)
C
C
       EXPN=0.0
       EXPP=0.0
       ETARS(I,K)=1.0
       SI(I,K)=ALPHA0(I,K)- RADSD(I,K)
       IF(SI(I,K))5,1,2
     1 WO1=OMEGAS(I,K)
       GO TO 9
     2 AS=A1(I,K)
       AC=A2(I,K)
       AQ=A3(I,K)
       IF(ASMPO(I,K)-SI(I,K))3,4,4
     3 WMWS=SI(I,K)/ASMPO(I,K)
       AR=ASMPO(I,K)/ASO(I,K)
       GO TO 8
     4 WMWS=1.0
       AR=SI(I,K)/ASO(I,K)
       GO TO 8
```

```
    5 AS=A4(I,K)
      AC=A5(I,K)
      AQ=A6(I,K)
      IF(SI(I,K)-ACMNO(I,K))6,4,4
    6 WMWS=SI(I,K)/ACMNO(I,K)
      AR=ACMNO(I,K)/ASO(I,K)
    8 WO1=(1.+AR*AR*(AS+AR*(AC+AR*AQ)))*WMWS*OMEGAS(I,K)
    9 ETAS(I,K)=(1.-(1./(PTOPS1(I,K)*(1.-WO1)+WO1))**EX)*PHI1(I,K)/
     1(PHI1(I,K)-1.)
      CALL CHECK(J)
      RETURN
      END
```

APPENDIX 3G

```
$IBFTC R        FULIST,DECK,SDD
CR
C       CALCULATE GAS CONSTANT
        SUBROUTINE R(P,T,F,W,RX)
        RX=53.35045+(.658*F+32.433*W)/(1.+F+W)
        RETURN
        END
```

APPENDIX 3H

```
$IBFTC GAMA     FULIST,DECK,SDD
CGAMMA
C
C      CALCULATE SPECIFIC HEAT RATIO FOR MIXTURE
       SUBROUTINE GAMMA(P,T,F,W,GAMX)
       CALL CPA(P,T,F,W,CPAX)
       IF(F)2,2,1
     1 CALL CPF(P,T,F,W,CPFX)
     2 IF(W)4,4,3
     3 CALL CPW(P,T,F,W,CPWX)
     4 CPGX=(CPAX+F*CPFX+W*CPWX)/(1.+F+W)
       CALL R(P,T,F,W,RX)
       GAMX=CPGX/(CPGX-RX/778.161)
       RETURN
       END
```

APPENDIX 3I

```
$IBFTC CPA     FULIST,DECK,SDD
CCPA
C       CALCULATE SPECIFIC HEAT RATIO FOR AIR
        SUBROUTINE CPA(P,T,F,W,CPAX)
        DIMENSION
       1XT(7),A(7)
        IF(T-100.)1,2,2
      1 TX=100.
        GO TO 5
      2 IF(6400.-T)3,4,4
      3 TX=6400.
        GO TO 5
      4 TX=T
      5 XT(1)=TX/1000.
        DO 6 I=2,7
      6 XT(I)=XT(I-1)*XT(1)
        CPAX=2.4264907E-01-2.6657395E-02*XT(1)+4.6617756E-02*XT(2)
       1-1.3546542E-02*XT(3)-8.4500931E-04*XT(4)+1.0303393E-03*
       2XT(5)-1.7159795E-04*XT(6)+9.1627911E-06*XT(7)
        RETURN
        END
```

APPENDIX 3J

```
$IBFTC CPF      FULIST,DECK,SDD
CCPF
C      CALCULATE SPECIFIC HEAT RATIO FOR FUEL
       SUBROUTINE CPF(P,T,F,W,CPFX)
       DIMENSION
      1XT(7),A(7)
       IF(T-400.)1,2,2
     1 TX=400.
       GO TO 5
     2 IF(3000.-T)3,4,4
     3 TX=3000.
       GO TO 5
     4 TX=T
     5 XT(1)=TX/1000.
       DO 6 I=2,7
     6 XT(I)=XT(I-1)*XT(1)
       CPFX=1.0625243E-01+9.5291284E-01*XT(1)-7.2605169E-01*XT(2)
      1+2.4481406E-01*XT(3)+5.3332162E-02*XT(4)-6.4699814E-02*XT(5)
      2+1.7495567E-02*XT(6)-1.6029820E-03*XT(7)
       RETURN
       END
```

APPENDIX 3K

```
$IBFTC CPW      FULIST,DECK,SDD
CCPW
C       CALCULATE SPECIFIC HEAT FOR WATER VAPOR
        SUBROUTINE CPW(P,T,F,W,CPWX)
        DIMENSION
       1XT(7),A(7)
        IF(T-400.)1,2,2
      1 TX=400.
        GO TO 5
      2 IF(3000.-T)3,4,4
      3 TX=3000.
        GO TO 5
      4 TX=T
      5 XT(1)=TX/1000.
        DO 6 I=2,7
      6 XT(I)=XT(I-1)*XT(1)
        CPWX=4.5728850E-01+9.7007556E-02*XT(1)+1.6536409E-01
       1*XT(2)-4.1138066E-02*XT(3)-2.6979575E-02*XT(4)+2.2619243E-02
       2*XT(5)-6.2706207E-03*XT(6)+6.2246710E-04*XT(7)
        RETURN
        END
```

APPENDIX 3L

```
$IBFTC PRIO    FULIST,DECK,SDD                                      PRIO 000
CPRATIO                                                             PRIO 001
C       CALCULATE PRESSURE RATIO                                    PRIO 002
        SUBROUTINE PRATIO(TFF,GAMX,RX,PTPS,PRTOL)                   PRIO 003
        A=GAMX/(GAMX-1.)                                            PRIO 004
        B=2./GAMX                                                   PRIO 005
        C=(GAMX+1.)/GAMX                                            PRIO 006
        D=TFF*SQRT(RX/(64.3481*A))                                  PRIO 007
        PCRIT=((GAMX+1.)/2.)**A                                     PRIO 008
        PUP=PCRIT                                                   PRIO 009
        PLOW=1.0                                                    PRIO 010
        PTRMO=0.0                                                   PRIO 011
      1 PTR=(PUP+PLOW)/2.                                           PRIO 012
        DELFM=SQRT(1./(PTR**B)-1./(PTR**C))-D                       PRIO 013
        IF(DELFM)2,3,3                                              PRIO 014
      2 PLOW=PTR                                                    PRIO 015
        GO TO 4                                                     PRIO 016
      3 PUP=PTR                                                     PRIO 017
      4 PRE=(PTR-PTRMO)/PTR                                         PRIO 018
        IF (ABS(PRE)-PRTOL)6,6,5                                    PRIO 019
      5 PTRMO=PTR                                                   PRIO 020
        GO TO 1                                                     PRIO 021
      6 IF(PCRIT-PTR)7,8,8                                          PRIO 022
      7 PTPS=PCRIT                                                  PRIO 023
        GO TO 9                                                     PRIO 024
      8 PTPS=PTR                                                    PRIO 025
      9 CONTINUE                                                    PRIO 026
        RETURN                                                      PRIO 027
        END                                                         PRIO 028
```

```
$IBFTC CHCK     FULIST,DECK,SDD
CCHECK
C        SUBROUTINE TO CHECK SENSÉ LIGHTS
      SUBROUTINE CHECK(J)
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LST
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC
     3DELPR,PASS,IPC,LOPC,ISS
C
      DO 1 I=1,4
      CALL SLITET(I,J)
      GO TO (2,1),J
    1 CONTINUE
      J=2
      RETURN
    2 J=1
      PREVER=.TRUE.
      RETURN
      END
```

APPENDIX 3N

```
$IBFTC ST1A    FULIST,DECK,SDD
CSTA1A
      SUBROUTINE STA1A
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      REAL M0
      COMMON /SSTA01/CP0(8),                         PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2            DPDR1(6,8),SI(6,8),   CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)
      REAL MR1A
      COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),   CP1A(8),
     1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTR1A(6
     2,8),MR1A(6,8)
C        DETERMINE FLOW CONDITIONS RELATIVE TO ROTOR,  FIND INCIDENCE
C         ANGLE RECOVERY ROTOR INLET STATIONS,  OBTAIN GAS PROPERTIES,
C         ABSOLUTE TANGENTIAL COMPONENT VELOCITY ADJUSTED FOR DIAMETER
C         CHANGE TO CONSERVE ANGULAR MOMENTUM,  AXIAL COMPONENT
C         VELOCITY ADJUSTED FOR WEIGHT FLOW, AREA,, AND DENSITY CHANGE
C         FROM STA  1.
C
C
      I=IP
      ID=-1
      TS1A      =TS1(I,K)
C        RATIO OF FLOW CHANGE
      WR=RWG(3,K)/RWG(2,K)
C        TOTAL STATION FLOW
      WGT1A(K)=WR*WGT1(K)
C        ADJUST TANGENTIAL VELOCITY
   13 VU1A(I,K)=VU1(I,K)*DP1(I,K)/DP1A(I,K)
```

```
C         ADJUST FLOW
      WG1A(I,K)=WR*WG1(I,K)
      RHOSTR=RHOS1(I,K)
C         ADJUST AXIAL VELOCITY
    1 VZ1A(I,K)=WR*VZ1(I,K)*ANN1(I,K)*RHOS1(I,K)/(ANN1A(I,K)
     1*RHOSTR)
      V1A      =SQRT(VU1A(I,K)*VU1A(I,K)+VZ1A(I,K)*VZ1A(I,K))
      IF(I-IP)2,3,2
    2 EX=(GAM(3,K)-1.)/GAM(3,K)
      EXI=1./EX
      GO TO 4
    3 IF(GAMF)12,12,2
   12 TA1A    =.5*(TTO(I,K)+TS1A       )
      CALL GAMMA(PTO(I,K),TA1A    ,FAIR,WAIR,GAM(3,K))
      EX=(GAM(3,K)-1.)/GAM(3,K)
      EXI=1./EX
    4 CP1A(K)=RG*EXI/AJ
      DELTS=(V1(I,K)*V1(I,K)-V1A      *V1A      )/(2.*G*AJ*CP1A(K))
      TS1A      =TS1(I,K)+DELTS
      PS1A(I,K)=PS1(I,K)*(1.+DELTS/TS1(I,K))**EXI
      RHOS1A      =144.*PS1A(I,K)/(RG*TS1A      )
C         DENSITY ERROR
      RHOE=(RHOS1A      -RHOSTR)/RHOS1A
      IF (ABS(RHOE)-RHOTOL)6,6,5
    5 RHOSTR=RHOS1A
      GO TO 1
    6 RU1A(I,K)=VU1A(I,K)-U1A(I,K)
      R1A(I,K)=SQRT(RU1A(I,K)*RU1A(I,K)+VZ1A(I,K)*VZ1A(I,K))
      SBET1A      =RU1A(I,K)/R1A(I,K)
      BET1A(I,K)=ATAN2(SBET1A      ,SQRT(1.-SBET1A      *SBET1A      ))
      IF(OMEGAR(I,K))8,8,7
    7 ETARR(I,K)=1.
      EXPRE=0.0
    8 MR1A(I,K)=R1A(I,K)/SQRT(GAM(3,K)*G*RG*TS1A      )
      TRTS1A      =1.+(GAM(3,K)-1.)*MR1A(I,K)*MR1A(I,K)/2.
      IF(TRTS1A.GT.1.) GO TO 32
      PREVER = .TRUE.
      GO TO 17
   32 TTR1A(I,K)=TS1A      *TRTS1A
      RI(I,K)=BET1A(I,K)-RADRD(I,K)
      IF(RI(I,K).GT.1.5707) RI(I,K)=1.5707
      IF(RI(I,K).LT.-1.5707) RI(I,K)= -1.5707
      IF(RI(I,K))9,9,10
    9 EXPR=EXPN
      GO TO 11
   10 EXPR=EXPP
   11 PRPS1A      =(1.+(TRTS1A      -1.)*ETARR(I,K)*(COS(RI(I,K))**
     1EXPR))**EXI
      PTR1A(I,K)=PS1A(I,K)*PRPS1A
      IF (ISECT-I)14,16,14
   14 I=I+ID
      IF (I)15,15,13
   15 ID=1
```

```
      I=IP+ID
      GO TO 13
   16 CONTINUE
      CALL CHECK(J)
      GO TO (17,18),J
   17 CALL DIAGT(3)
   18 RETURN
      END
```

```
$IBFTC ST2     FULIST,DECK,SDD
CSTA2
C       SATISFY CONTINUITY OF FLOW AT ROTOR EXIT
        SUBROUTINE STA2
C
        REAL MFSTOP
        LOGICAL PREVER
        COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
      1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LST(
      2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC
      3DELPR,PASS,IPC,LOPC,ISS
C
        COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(
      1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD((
      2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
      3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
        COMMON /SINPUT/
      1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXI
      2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
      3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1((
      4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),
      5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(
      6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8)
      76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(
      8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
        REAL MR1A
        COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),  CP1A(8
      1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTF
      2,8),MR1A(6,8)
C
        COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2
      1            PS2(6,8),PHI2(6,8)
C
        REAL MR2,M2      ,MF2
        COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),F
      1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)
C
        DIMENSION WGT2C(8),FFA2(6,8),IS2(8)
C
C
        J=1
        SCRIT=0.0
        PTRMO=1.
        IS2(K)=0
        EXI=GAM(3,K)/(GAM(3,K)-1.)
        WR=RWG(4,K)/RWG(3,K)
        DO 1 I=1,ISECT
        TTR2(I,K)=TTR1A(I,K)+(U2(I,K)**2 - U1A(I,K)**2)/(2.*G*AJ*CP1A(
        PTR2(I,K)=PTR1A(I,K)*(TTR2(I,K)/TTR1A(I,K))**EXI
      1 WG2(I,K)=WR*WG1A(I,K)
```

```
      WGT2(K)=WR*WGT1A(K)
      I=IP
      ID=-1
      WGT2C(K)=0.
      IF(ICHOKE)26,26,3
   26 IF(LOPIN)27,27,3
   27 IF(GAMF)2,2,16
    2 TA2(K)=.95*TTR2(IP,K)
      CALL GAMMA(PTR2(I,K),TA2(K),FAIR,WAIR,GAM(4,K))
   16 FFA2(I,K)=WG2(I,K)*SQRT(TTR2(I,K))/(144.*PTR2(I,K)*CSBET2(I,K)*
     1ANN2(I,K))
      CALL PRATIO(FFA2(I,K),GAM(4,K),RG,PTRS2(I,K),PRTOL)
    3 CALL FLOW2(I)
      IF (PREVER) GO TO 22
      WGT2C(K)=WGT2C(K)+WG2(I,K)
      L=1
      IF (PTRS2(I,K).LE.PTRS2(IP,K))    L=I
      IF(ISECT-I)7,7,4
    4 I=I+ID
      IF(I)5,5,6
    5 ID=1
      I=IP+ID
    6 L=I-ID
      PS2(I,K)=PS2(L,K)+FLOAT(ID)*DPDR2(L,K)*(H2(I,K)+H2(L,K)
     1)/2.
      PTRS2(I,K)=PTR2(I,K)/PS2(I,K)
      IF (PTRS2(I,K)-1.)19,19,3
   19 PTRS2(I,K) = 1.0  + PRTOL
      GO TO 3
    7 IF(IS2(K))8,8,9
    8 EXI=GAM(4,K)/(GAM(4,K)-1.)
      CALL PHIM(EXI,ETAR(L,K),PHIX,PRCRIT)
      PRUP=PTR2(IP,K)*PRCRIT*PS2(L,K)/(PTR2(L,K)*PS2(IP,K))
     1*(1.+PRTOL)
      PRLOW=1.
      GO TO 10
    9 IS2(K)=IS2(K)+1
   10 L = IBRC + 1
      IF(ICHOKE.EQ.L) PTRS2(IP,K) = PRUP
      IF(WGT2(K)-WGT2C(K))12,15,11
   11 PRLOW= PTRS2(IP,K)
      GO TO 13
   12 PRUP= PTRS2(IP,K)
      IS2(K)=1
   13 WE=1.-WGT2(K)/WGT2C(K)
      J=J+1
      IF(J-32)29,18,18
   29 IF(ICHOKE-L) 30,31,30
   31 SCRIT= -WE
      GO TO 15
   30 IF(LOPIN)14,14,15
   14 PRE=(PTRS2(IP,K)-PTRMO)/PTRS2(IP,K)
      IF (ABS(PRE)-PRTOL)17,17,24
```

```
   17 CONTINUE
      IF (ABS(WE)-WTOL)15,15,23
   24 PTRMO=PTRS2(IP,K)
      WGT2C(K)=0.0
      I=IP
      ID=-1
      IF (SCRIT)28,28,15
   28 PTRS2(IP,K)=.5*(PRLOW+PRUP)
      IF (PTRS2(IP,K).LE.PRCRIT)   PRPC=0.0
      GO TO 3
   23 SCRIT= 1.
   15 IF(TRLOOP.EQ.0.) GO TO 25
   18 WRITE(6,1000)K,PRUP,PRLOW,WE,PRCRIT,J,WGT2(K),WGT2C(K),(WG2(L,K)
     1 L=1,ISECT)
      WRITE(6,1001)(PTRS2(L,K),L=1,ISECT)
 1000 FORMAT(2X,2HK=I4,     2X,6H PRUP=F8.5,2X,6HPRLOW=F8.5,2X,6H    WE=
     1F8.5,1X,7HPRCRIT=F8.5,2X,2HJ=I4/
     22X,6H WGT2=F8.3,2X,6HWGT2C=F8.3/
     32X,6H  WG2=6F8.3)
 1001 FORMAT(2X,6HPTPS2=6F8.5)
   25 CALL CHECK(J)
      GO TO (20,21),J
   20 CALL DIAGT(4)
      GO TO 22
   21 CALL LOOP
   22 RETURN
      END
```

APPENDIX 3P

```
$IBFTC FLW2    FULIST,DECK,SDD
CFLOW2
C      CALCULATE ROTOR EXIT SECTOR FLOW
       SUBROUTINE FLOW2(I)
C
       REAL MFSTOP
       LOGICAL PREVER
       COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
      1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
      2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
      3DELPR,PASS,IPC,LOPC,ISS
C
       COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
      1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
      2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
      3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
       COMMON /SINPUT/
      1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
      2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
      3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
      4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
      5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6,8)
      6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
      76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
      8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
       COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2(8),
      1            PS2(6,8),PHI2(6,8)
C
       REAL MR2,M2      ,MF2
       COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),RU2(6
      1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)
C
       DIMENSION P1AS2C(8),PHI2C(8),R2C(6,8),TS2C(6,8),RHOS2C(6,8),WG2C(6
      1,8),CBET2E(6,8),AS2(6,8),RFF(6,8)
C
C
       EX=(GAM(4,K)-1.)/GAM(4,K)
C         ISENTROPIC ROTOR RELATIVE TEMPERATURE RATIO
    10 PHI2(I,K)= PTRS2(I,K)**EX
       IF(OMEGAR(I,K))2,2,1
     1 CALL LOSS2(I,K)
C         EXIT TEMPERATURES
     2 TS2(I,K)=TTR2(I,K)*(1.-ETAR(I,K)*(1.-1./PHI2(I,K)))
       IF(I-IP)6,3,6
     3 IF(     GAMF)4,4,5
     4 TA2(K)=.5*(TTR2(I,K)+TS2(I,K))
       CALL GAMMA(PTR2(I,K),TA2(K),FAIR,WAIR,GAM(4,K))
     5 EXI=GAM(4,K)/(GAM(4,K)-1.)
       EX=1./EXI
C         CRITICAL PRESSURE RATIO
```

```
      CALL PHIM(EXI,ETAR(I,K),PHI2C(K),P1AS2C(K))
C        SPECIFIC HEAT AT CONSTANT PRESSURE
    6 CP2(K)=RG*EXI/AJ
C        RELATIVE EXIT VELOCITY
      R2(I,K)=SQRT(2.*G*AJ*CP2(K)*(TTR2(I,K)-TS2(I,K)))
C        EXIT PRESSURE
      PS2(I,K)= PTR2(I,K)/ PTRS2(I,K)
C        EXIT DENSITY
      RHOS2(I,K)=144.*PS2(I,K)/(RG*TS2(I,K))
C        TEST CRITICAL PRESSURE RATIO
      IF( PTRS2(I,K)-P1AS2C(K))15, 7,7
    7 IF (IP-I) 22,8,22
    8 IF (PRPC)9,9,18
    9       PRPC=1.
       PTRS2(I,K)=P1AS2C(K)*(1.+PRTOL)
      GO TO 10
   22 IF (PTRS2(I,K).LE.PTRS2(IP,K))   GO TO 18
      GO TO 13
   18 IF ((I.EQ.1).OR.(I.EQ.ISECT))  SCRIT=1.
      GO TO 11
   11 CONTINUE
      R2C(I,K)=SQRT(2.*G*AJ*CP2(K)*TTR2(I,K)*ETAR(I,K)*(
     1PHI2C(K)-1.)/PHI2C(K))
      TS2C(I,K)=TTR2(I,K)*(1.-ETAR(I,K)*(1.-1./PHI2C(K)))
      RHOS2C(I,K)=144.*PTR2(I,K)/(RG*P1AS2C(K)*TS2C(I,K))
      WG2C(I,K)=RHOS2C(I,K)*R2C(I,K)*ANN2(I,K)*CSBET2(I,K)
   12 WG2(I,K)=WG2C(I,K)
      GO TO 14
   13 IF(   PRPC-1.)15,15,24
   24 WG2(I,K)=RFF(I,K)*PTR2(I,K)/SQRT(TTR2(I,K))
      GO TO 14
C        OVEREXPANSION AFTER SUPERSONIC FLOW DECREASE
   14 CBET2E(I,K)=WG2(I,K)/(RHOS2(I,K)*R2(I,K)*ANN2(I,K))
      BET2E(I,K)=ATAN2(SQRT(1.-CBET2E(I,K)*CBET2E(I,K)),CBET2E(I,K))
      GO TO 16
   15 WG2(I,K)=RHOS2(I,K)*R2(I,K)*ANN2(I,K)*CSBET2(I,K)
      CBET2E(I,K)=CSBET2(I,K)
      BET2E(I,K)= BET2(I,K)
      RFF(I,K)=WG2(I,K)*SQRT(TTR2(I,K))/PTR2(I,K)
   16 RU2(I,K)=R2(I,K)*SIN(BET2E(I,K))
      VU2(I,K)=RU2(I,K)-U2(I,K)
      DPDR2(I,K)=  (RHOS2(I,K)*VU2(I,K)*VU2(I,K)/(G*DP2(I,K)))*.013889
      VZ2(I,K)=R2(I,K)*CBET2E(I,K)
      AS2(I,K)=SQRT(GAM(4,K)*G*RG*TS2(I,K))
      V2(I,K)=SQRT(VZ2(I,K)*VZ2(I,K)+VU2(I,K)*VU2(I,K))
      M2(I,K)=V2(I,K)/AS2(I,K)
      MR2(I,K)=R2(I,K)/AS2(I,K)
      MF2(I,K)=MR2(I,K)*CBET2E(I,K)
      IF(I.LT.ISECT) GO TO 17
      IF(PRPC.EQ.1.) PRPC=2.
   17 CALL CHECK(J)
      GO TO (19,21),J
   19 CALL DIAGT(4)
```

```
21 RETURN                                                         FLW2 105
   END                                                            FLW2 106
```

APPENDIX 3Q

```
$IBFTC LOS2    FULIST,DECK,SDD
CLOSS2
C      CALCULATE ETA R FROM QUADRATIC POLYNMONIAL
       SUBROUTINE LOSS2(I,K)
C
       REAL MFSTOP
       LOGICAL PREVER
       COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
      1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
      2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
      3DELPR,PASS,IPC,LOPC,ISS
C
       COMMON /SINPUT/
      1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
      2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
      3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
      4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
      5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(6,8)
      6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
      76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
      8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
       REAL MR1A
       COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),  CP1A(8),
      1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTR1A(6
      2,8),MR1A(6,8)
C
       COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2(8),
      1            PS2(6,8),PHI2(6,8)
C
C
       ETARR(I,K)=1.0
       IF(RI(I,K))4,1,2
     1 W1A2=OMEGAR(I,K)
       GO TO 8
     2 AS=B1(I,K)
       AC=B2(I,K)
       AQ=B3(I,K)
       IF(BSMPIA(I,K)-RI(I,K))3,6,6
     3 WMWR=RI(I,K)/BSMPIA(I,K)
       AR=BSMPIA(I,K)/BSIA(I,K)
       GO TO 7
     4 AS=B4(I,K)
       AC=B5(I,K)
       AQ=B6(I,K)
       IF(RI(I,K)-BCMNIA(I,K))5,6,6
     5 WMWR=RI(I,K)/BCMNIA(I,K)
       AR=BCMNIA(I,K)/BSIA(I,K)
       GO TO 7
     6 WMWR=1.0
       AR=RI(I,K)/BSIA(I,K)
     7 W1A2=OMEGAR(I,K)*(1.+AR*AR*(AS+AR*(AC+AR*AQ)))*WMWR
```

```
 8 EX=(GAM(3,K)-1.)/GAM(3,K)
   ETAR(I,K)=(1.-(1./(PTRS2(I,K)*(1.-W1A2)+W1A2))**EX)*PHI2(I,K)/
  1(PHI2(I,K)-1.)
   CALL CHECK(J)
   RETURN
   END
```

```
$IBFTC LOOP     FULIST,DECK,SDD
CLOOP
      SUBROUTINE LOOP
C        HANDLES ALL LOGIC FOR ITERATING TO OBTAIN EXACT CHOKE POINT
C                   UNDERFLOW, NO CHOKE INITIAL CHOKE, CHOKE ITERATIC
C                   SUBCRITICAL, CHOKE ITERATION SUPERCRITICAL,MULTIP
C                   CHOKE, CHOKE ITERATION COMPLETE
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTC
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,BLLD,STG,SECT,EXF
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),ASO(
     6,ASMPO(6,8),ACMNO(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8)
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      IJ=8+KSTG
C     INCREASE BLADE ROW COUNTER
      IBRC=IBRC+1
C     TEST NEGATIVE SECTOR PRESSURE RATIO
      IF (PTRN)18,1,1
C     TEST CHOKE ITERATION ON BLADE ROW
    1 IF (ICHOKE-IBRC)3,2,3
C     TEST INCREMENT TOLERANCE
    2 IF (PRTOL-DELPR)3,3,4
C     TEST STATION FLOW CRITICAL
    3 IF (SCRIT)5,5,6
C     CHOKE ITERATION COMPLETE
    4 ICHOKE=0
      IPC=IBRC
      ISS=IBRC
      ISORR=2+(IBRC/2)*2-IBRC
      JL=(ISORR-1)*8+KN
      IF (JL-IJ)22,23,23
   22 DELPR=DELL
   24 LOPC=0
      CHOKE=1.
      LSTG=KN
      LBRC=IBRC-1
      GO TO 18
   23 DELPR=DELA
      GO TO 24
    5 IF (ICHOKE-IBRC)18,7,18
```

```
C       TEST CHOKE ITERATION LOOP
      6 IF(ISS-IBRC)8,18,18
C       CHOKE ITERATION
C       ISORR = 1 FOR STATOR
C             = 2 FOR ROTOR
      7 DELPR=DELPR/2.
        JL=(ISORR-1)*8+LSTG
        PTOPS1(IP,JL)=PTOPS1(IP,JL)+DELPR
         GO TO 16
C       CHOKE HAS OCURRED
      8 IF(ICHOKE)80,80,13
     80 J=(IBRC-2*(KN-1)-1)*8+KN
        WRITE(6,801)IBRC,PTOPS1(IP,J)
    801 FORMAT(16X10HBLADE ROW I3,8H CHOKED,4X5HPTPS=F10.5)
C       TEST SINGLE CALCULATION POINT
      9 IF (DELC)18,18,10
C       TEST PREVIOUS CHOKE
     10 IF (IPC)11,11,12
C       SAVE COMBINATIONS PRIOR FIRST CHOKE
     11 LBRCS=LBRC
        ISORRS=ISORR
        JL=(ISORR-1)*8+LSTG
        SPTPS=PTOPS1(IP,JL)-DELPR
        LSTGS=LSTG
        SDELPR=DELPR
        GO TO 13
     12 JL=LSTGS+(ISORRS-1)*8
        DELNU = (PTOPS1(IP,JL)-SPTPS)/4.
        IF (DELNU.LE.0.0001) DELNU = SDELPR/4.
        DELPR = DELNU
        SDELPR = DELNU
        WRITE(6,1201)IPC,IBRC,DELPR
   1201 FORMAT(6X11HBLADE ROWS I5,5H AND I5,25H, CHOKED - INCREMENT NOW
       1F10.5)
        LBRC=LBRCS
        LSTG=LSTGS
        ISORR=ISORRS
        PTOPS1(IP,JL) = SPTPS + SDELPR
        LOPC=10
        ICHOKE=0
        IPC=0
        ISS=0
        CHOKE=0.0
        GO TO 17
C       TEST PREVIOUS COMPLETE CALCULATION
     13 IF (PASS)15,15,14
     14 ICHOKE=IBRC
        DELPR=.5*DELPR
     15 JL=(ISORR-1)*8+LSTG
        PTOPS1(IP,JL)=PTOPS1(IP,JL)-DELPR
C       SET INDEX REGISTERS
     16 CONTINUE
        LOPC=LOPC+1
```

```
C SET JUMP FOR CHOKE ITERATION
   17 JUMP=1
      GO TO 19
C     JUMP SET FOR NO CHOKE OR CHOKE COMPLETE
   18 JUMP=0
C     TEST LOOP-TRACE
   19 IF (TRLOOP)21,21,20
   20 WRITE(6,2001)IBRC,LBRC,ISORR,KN,LSTG,IPC,ISS,ICHOKE,JUMP,LBRCS,
     1ISORRS,LSTGS,SPTPS,PTOPS1(IP,JL),DELPR,DELL,SCRIT,LOPC
 2001 FORMAT(3X12I5/3X4F10.5,F10.0,I10)
   21 RETURN
      END
```

APPENDIX 3S

```
$IBFTC ST2A    FULIST,DECK,SDD
CSTA2A
C      DETERMINE INLET FLOW CONDITIONS TO ALL STATORS
C                      AFTER THE FIRST STATOR
      SUBROUTINE STA2A
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      REAL M0
      COMMON /SSTA01/CP0(8),                    PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2             DPDR1(6,8),SI(6,8),  CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)
C
      REAL MR1A
      COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),  CP1A(8),
     1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTR1A(6
     2,8),MR1A(6,8)
C
      COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2(8),
     1            PS2(6,8),PHI2(6,8)
C
      REAL MR2,M2     ,MF2
      COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),RU2(6
     1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)
C
      REAL M2A,MF2A
      COMMON /SSTA2A/WG2A(6,8),WGT2A(8),VU2A(6,8),VZ2A(6,8),PS2A(6,8),
     1ALF2A(6,8),TT2A(6,8),PT2A(6,8),TTBAR(8),PTBAR(8),STTO(8),SPTO(8),
     2M2A(6,8),MF2A(6,8),CP2A(8),V2A(6,8),TS2A(6,8),TAS(8),PAS(8),GAMS(8
     3),CPS(8),DELHVD(6,8)
```

```
C
      DIMENSION              TTTS2A(6,8)
C
C
      ID=-1
      I=IP
      TS2A(I,K)=TS2(I,K)
      WR=RWG(5,K)/RWG(4,K)
      SUMT=0.0
      SUMLT=0.0
      SUMLP=0.0
      WGT2A(K)=WR*WGT2(K)
   12 VU2A(I,K)=VU2(I,K)*DP2(I,K)/DP2A(I,K)
      WG2A(I,K)=WR*WG2(I,K)
      RHOSTR=RHOS2(I,K)
    1 VZ2A(I,K)=WR*VZ2(I,K)*ANN2(I,K)*RHOS2(I,K)/(ANN2A(I,K)*RHOSTI
      V2A(I,K)=SQRT(VU2A(I,K)*VU2A(I,K)+VZ2A(I,K)*VZ2A(I,K))
      IF(I-IP)4,2,4
    2 IF(     GAMF)3,3,4
    3 TA2A    =.5*(TTR2(I,K)+TS2A(I,K))
      CALL GAMMA(PTR2(IP,K),TA2A    ,FAIR,WAIR,GAM(5,K))
    4 EX=(GAM(5,K)-1.)/GAM(5,K)
      EXI=1./EX
      CP2A(K)=RG*EXI/AJ
      DELTS=(V2(I,K)*V2(I,K)-V2A(I,K)*V2A(I,K))/(2.*G*AJ*CP2A(K))
      TS2A(I,K)=TS2(I,K)+DELTS
      IF(TS2A(I,K).GT.0.) GO TO 32
      PREVER = .TRUE.
      MFSTOP = 2.
      GO TO 30
   32 PS2A(I,K)=PS2(I,K)*(1.+DELTS/TS2(I,K))**EXI
      RHOS2A      =144.*PS2A(I,K)/(RG*TS2A(I,K))
      IF(ABS(RHOSTR-RHOS2A      )-1.E-07)6,6,5
    5 RHOSTR=RHOS2A
      GO TO 1
    6 SALF2A      =VU2A(I,K)/V2A(I,K)
      ALF2A(I,K)=ATAN2(SALF2A      ,SQRT(1.-SALF2A      *SALF2A
   11 IF (I-IP)28,24,28
   24 IF (GAMF)25,25,26
   25 TAS(K)=.5*(TA1(K)+TA2(K))
      PAS(K)=.5*(PTO(IP,K)+PT2A(IP,K))
      CALL GAMMA(PAS(K),TAS(K),FAIR,WAIR,GAMS(K))
      GO TO 27
   26 GAMS(K)=.5*(GAM(2,K)+GAM(4,K))
   27 E4=GAMS(K)/(GAMS(K)-1.)
      CPS(K)=RG*E4/AJ
   28 DELHVD(I,K)=(U1A(I,K)*VU1A(I,K)+U2(I,K)*VU2(I,K))/AJ/G
      M2A(I,K)=V2A(I,K)/SQRT(GAM(5,K)*G*RG*TS2A(I,K))
      DELTT=TFR(I,K)*DELHVD(I,K)/CPS(K)
      TT2A(I,K)=TTO(I,K)-DELTT
      TTTS2A(I,K)=1.+(M2A(I,K)*M2A(I,K)*(GAM(5,K)-1.)/2.)
      PTPS2A      = (TTTS2A(I,K))**EXI
      PT2A(I,K)=PS2A(I,K)*PTPS2A
```

```
      MF2A(I,K)=M2A(I,K)*COS(ALF2A(I,K))
      IF (ISECT-I)13,15,13
   13 I=I+ID
      IF (I)14,14,12
   14 ID=1
      I=IP+ID
      GO TO 12
   15 CONTINUE
      DO 16 I=1,ISECT
      RW=WG2A(I,K)/WGT2A(K)
      TR=TT2A(I,K)/TT2A(IP,K)
      PR=PT2A(I,K)/PT2A(IP,K)
      SUMT=SUMT+RW*TR
      SUMLT=SUMLT+RW*ALOG(TR)
   16 SUMLP=SUMLP+RW*ALOG(PR)
      E3=GAM(5,K)/(GAM(5,K)-1.)
      TTBAR(K)=TT2A(IP,K)*SUMT
      PTBAR(K)=PT2A(IP,K)*EXP(SUMLP+E3*(ALOG(SUMT)-SUMLT))
      IF (K-KSTG)17,18,18
   17 STTO(K+1)=TTBAR(K)
      SPTO(K+1)=PTBAR(K)
      DO 23 I=1,ISECT
   29 SI(I,K+1)=ALF2A(I,K)- RADSD(I,K+1)
      IF(SI(I,K+1).GT. 1.5707) SI(I,K+1)= 1.5707
      IF(SI(I,K+1).LT.-1.5707) SI(I,K+1)=-1.5707
      IF(OMEGAS(I,K))8,8,7
    7 ETARS(I,K+1)=1.0
      EXPSI=0.
      GO TO 117
    8 IF(SI(I,K+1))9,9,10
    9 EXPSI=EXPN
      GO TO 117
   10 EXPSI=EXPP
  117 IF (PAF-1.)19,20,21
C     UNIFORM PROFILES
   19 PTP(I,K+1)=PTBAR(K)
      PTO(I,K+1)= PTP(I,K+1)
     1*(1.+(TTTS2A(I,K)-1.)*ETARS(I,K+1)*(COS(SI(I,K+1))**EXPSI))**EXI
     2/(TTTS2A(I,K))**EXI
      TTO(I,K+1)=TTBAR(K)
      GO TO 23
C     SAVE PROFILES
   20 PTP(I,K+1)=PT2A(I,K)
      PTO(I,K+1)= PTP(I,K+1)
     1*(1.+(TTTS2A(I,K)-1.)*ETARS(I,K+1)*(COS(SI(I,K+1))**EXPSI))**EXI
     2/(TTTS2A(I,K))**EXI
      GO TO 22
C     SMOOTH PRESSURE PROFILES
   21 PTP(I,K+1)=PTBAR(K)* ((TTRA(I,K)/TTBAR(K))**E3
      PTO(I,K+1)= PTP(I,K+1)
     1*(1.+(TTTS2A(I,K)-1.)*ETARS(I,K+1)*(COS(SI(I,K+1))**EXPSI))**EXI
     2/(TTTS2A(I,K))**EXI
   22 TTO(I,K+1)=TT2A(I,K)
```

```
23 CONTINUE
18 MFSTOP=MF2A(IP,K)/AACS
   CALL CHECK(J)
   GO TO (30,31),J
30 CALL DIAGT(5)
31 RETURN
   END
```

APPENDIX 3T

```
$IBFTC ST1     FULIST,DECK,SDD
CSTA1
C      SATISFY CONTINUITY OF FLOW AT EXIT OF ALL STATORS
C                            AFTER THE FIRST STATOR
      SUBROUTINE STA1
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1 K,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      REAL M0
      COMMON /SSTA01/CP0(8),                    PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2             DPDR1(6,8),SI(6,8),  CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)
C
      REAL M2A,MF2A
      COMMON /SSTA2A/WG2A(6,8),WGT2A(8),VU2A(6,8),VZ2A(6,8),PS2A(6,8),
     1ALF2A(6,8),TT2A(6,8),PT2A(6,8),TTBAR(8),PTBAR(8),STT0(8),SPT0(8),
     2M2A(6,8),MF2A(6,8),CP2A(8),V2A(6,8),TS2A(6,8),TAS(8),PAS(8),GAMS(8
     31),CPS(8),DELHVD(6,8)
C
      DIMENSION WGT1C(8),LC1(8),FFA1(6,8)
C
C
      J=1
      SCRIT=0.0
      PTRM0=1.
      WR=RWG(2,K)/RWG(5,K-1)
      DO 1 I=1,ISECT
      WG1(I,K)=WR*WG2A(I,K-1)
      ALPHA0(I,K)  =ALF2A(I,K-1)
        PS0(I,K) =   PS2A(I,K-1)
         V0(I,K) =    V2A(I,K-1)
```

```
      TSO(I,K) =   TS2A(I,K-1)
      VUO(I,K) =   VU2A(I,K-1)
      VZO(I,K) =   VZ2A(I,K-1)
       MO(I,K) =    M2A(I,K-1)
    1 WGT1(K)=WR*WGT2A(K-1)
      I=IP
      ID=-1
      WGT1C(K)=0.0
      LC1(K)=0
      IF(ICHOKE)17,17,16
   17 IF(LOPIN)18,18,16
   18 IF(GAMF)2,2,3
    2 TA1(K)=.95*TTO(IP,K)
      CALL GAMMA(PTO(IP,K),TA1(K),FAIR,WAIR,GAM(2,K))
    3 FFA1(I,K)=WG1(I,K)*SQRT(TTO(I,K))/(144.*PTO(I,K)*ANN1(I,K)
     1*CSALF1(I,K))
      CALL PRATIO(FFA1(I,K),GAM(2,K),RG,PTOPS1(I,K),PRTOL)
   16 CALL FLOW1(I)
      IF (PREVER) GO TO 25
      WGT1C(K)=WGT1C(K)+WG1(I,K)
      L=1
      IF (PTOPS1(I,K).LE.PTOPS1(IP,K)) L=I
      IF(ISECT-I)7,7,4
    4 I=I+ID
      IF(I)5,5,6
    5 ID=1
      I=IP+ID
    6 L=I-ID
      PS1(I,K)=PS1(L,K)+FLOAT(ID)*DPDR1(L,K)*(H1(I,K)+H1(L,K))/2.
      PTOPS1(I,K)=PTO(I,K)/PS1(I,K)
      GO TO 16
    7 IF(LC1(K))8,8,9
    8 LC1(K)=1
      EX=GAM(2,K)/(GAM(2,K)-1.)
      CALL PHIM(EX,ETAS(L,K),PHIX,PRCRIT)
      PRUP= PTOPS1(IP,K)*PRCRIT/PTOPS1(L,K)
     1*(1.+PRTOL)
      PRLOW=1.0
      GO TO 10
    9 LC1(K)=LC1(K)+1
   10 L = IBRC + 1
      IF(ICHOKE.EQ.L) PTOPS1(IP,K) = PRUP
      IF(WGT1(K)-WGT1C(K))12,15,11
   11 PRLOW=PTOPS1(IP,K)
      GO TO 13
   12 PRUP=PTOPS1(IP,K)
   13 WE=1.-WGT1(K)/WGT1C(K)
      J=J+1
      IF(J-32)29,22,22
   29 IF(ICHOKE-L) 30,31,30
   31 SCRIT= -WE
      GO TO 15
   30 IF(LOPIN)14,14,15
```

```
   14 PRE=(PTOPS1(IP,K)-PTRMO)/PTOPS1(IP,K)
      IF (ABS(PRE)-PRTOL)21,21,27
   21 CONTINUE
      IF (ABS(WE)-WTOL)15,15,20
   27 PTRMO=PTOPS1(IP,K)
      WGT1C(K)=0.0
      I=IP
      ID=-1
      IF (SCRIT)19,19,15
   19 PTOPS1(IP,K)=.5*(PRLOW+PRUP)
      IF (PTOPS1(IP,K).LE.PRCRIT)   PRPC=0.
      GO TO 16
   20 SCRIT= 1.
   15 IF(TRLOOP.EQ.0.) GO TO 28
   22 WRITE(6,1000)K,PRUP,PRLOW,WE,PRCRIT,J,WGT1(K),WGT1C(K),(WG1(L,K),
     1 L=1,ISECT)
      WRITE(6,1001)(PTOPS1(L,K),L=1,ISECT)
 1000 FORMAT(2X,2HK=I4,     2X,6H PRUP=F8.5,2X,6HPRLOW=F8.5,2X,6H    WE=
     1F8.5,1X,7HPRCRIT=F8.5,2X,2HJ=I4/
     22X,6H WGT1=F8.3,2X,6HWGT1C=F8.3/
     32X,6H  WG1=6F8.3)
 1001 FORMAT(1X,7HPTOPS1=6F8.5)
   28 CALL CHECK(J)
      GO TO (23,24),J
   23 CALL DIAGT(2)
      GO TO 25
   24 CALL LOOP
   25 RETURN
      END
```

```
$IBFTC OVLL    FULIST,DECK,SDD
COVRALL
      SUBROUTINE OVRALL
C     PURPOSE IS TO CALCULATE STAGE PERFORMANCE VALUES
C     AFTER FLOW ITERATION IS COMPLETED THROUGH THE LAST STAGE
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      REAL M0
      COMMON /SSTA01/CP0(8),                        PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2             DPDR1(6,8),SI(6,8),  CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)
C
      REAL MR1A
      COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),  CP1A(8),
     1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTR1A(6
     2,8),MR1A(6,8)
      COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2(8),
     1             PS2(6,8),PHI2(6,8)
C
      REAL MR2,M2     ,MF2
      COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),RU2(6
     1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)
C
      REAL M2A,MF2A
      COMMON /SSTA2A/WG2A(6,8),WGT2A(8),VU2A(6,8),VZ2A(6,8),PS2A(6,8),
     1ALF2A(6,8),TT2A(6,8),PT2A(6,8),TTBAR(8),PTBAR(8),STTO(8),SPTO(8),
     2M2A(6,8),MF2A(6,8),CP2A(8),V2A(6,8),TS2A(6,8),TAS(8),PAS(8),GAMS(8
     3),CPS(8),DELHVD(6,8)
C
      COMMON /SOVRAL/DELHT(6,8),DELHTI(6,8),DELHSI(6,8),DEHATI(6,8),
```

```
     1ETATT(6,8),ETATS(6,8),ETATAT(6,8)
C
      REAL MIS(8),MIRS(8),MR1AR(8),MR2T(8)
      DIMENSION SA0(8),SIS(8),SB1A(8),SIR(8),SA2(8),THCR(8),EPSI(8),DELT
     1(8),SETATT(8),SETATS(8),SETAAT(8),SWRTP(8),SNRT(8),SDHT(8),SETHC(8
     2),SNRTHC(8),SWRTED(8),SPTPT2(8),SPTPS2(8),ST2TT0(8),STRTT0(8),UPS(
     38),UPUPS(8),URS(8),URURS(8),VIS(8),UPVIS(8),URVIS(8),PSIPS(8),PSIR
     4S(8),RXP(8),RXR(8),DBETAR(8),DELHTS(8),DEHTIS(8),DEHSIS(8),DHATIS(
     58),PAT2A(6,8),WGT0(8)
C
      DATA RSL,TSL,PSL,GAMSL/53.35045,518.688,14.696,1.4/
C
C
      STT0(1)=TTIN
      SPT0(1)=PTIN
      TA0=0.0
      PA0=0.0
      GAM0=0.0
      OUPUP=0.0
      OURUR=0.0
      ODELHT=0.0
    5 E1=GAMSL/(GAMSL-1.)
      DO 17 K=1,KSTG
      IF (GAMF)1,1,2
    1 TA0=TA0+TAS(K)
      PA0=PA0+PAS(K)
      GO TO 3
    2 GAM0=GAM0+GAMS(K)
    3 E2=GAM(1,K)/(GAM(1,K)-1.)
      E3=GAM(5,K)/(GAM(5,K)-1.)
      E4=GAMS(K)/(GAMS(K)-1.)
      E5=1./E4
      DELHTS(K)=0.0
      DEHTIS(K)=0.0
      DEHSIS(K)=0.0
      DHATIS(K)=0.0
      DO 6 I=1,ISECT
      RW=WG2A(I,K)/WGT2A(K)
      DELHT(I,K)=DELHVD(I,K)*TFR(I,K)
      DELHTI(I,K)=CPS(K)*TT0(I,K)*(1.-(PT2A(I,K)/PTP(I,K))**E5)
      ETATT(I,K)=DELHT(I,K)/DELHTI(I,K)
      DELHSI(I,K)=CPS(K)*TT0(I,K)*(1.-(PS2A(I,K)/PTP(I,K))**E5)
      ETATS(I,K)=DELHT(I,K)/DELHSI(I,K)
      PAT2A(I,K)=PS2A(I,K)*(1.+(GAM(5,K)-1.)*MF2A(I,K)*MF2A(I,K)
     1/2.)**E3
      DEHATI(I,K)=CPS(K)*TT0(I,K)*(1.-(PAT2A(I,K)/PTP(I,K))**E5)
      ETATAT(I,K)=DELHT(I,K)/DEHATI(I,K)
      DELHTS(K)=DELHTS(K)+RW*DELHT(I,K)
      DEHTIS(K)=DEHTIS(K)+RW*DELHTI(I,K)
      DEHSIS(K)=DEHSIS(K)+RW*DELHSI(I,K)
      DHATIS(K)=DHATIS(K)+RW*DEHATI(I,K)
    6 CONTINUE
   13 SA0(K)=ALPHA0(IP,K)*57.2958
```

```
      SIS(K)=SI(IP,K)*57.2958
      SB1A(K)=BET1A(IP,K)*57.2958
      SIR(K)=RI(IP,K)*57.2958
      SA2(K)=ALF2A(IP,K)*57.2958
      THCR(K)= GAM(1,K)*(GAMSL+1.)*RG*STTO(K)/
     1(GAMSL*(GAM(1,K)+1.)*RSL*TSL)
      EPSI(K)=GAMSL*((GAM(1,K)+1.)/2.)**E2/(GAM(1,K)*((GAMSL
     1+1.)/2.)**E1)
      DELT(K)=SPTO(K)/PSL
      SETATT(K)=DELHTS(K)/DEHTIS(K)
      SETATS(K)=DELHTS(K)/DEHSIS(K)
      SETAAT(K)=DELHTS(K)/DHATIS(K)
      WGTO(K)=WGT1(K)/RWG(2,K)
      SWRTP(K)= WGTO(K)*SQRT(STTO(K))/SPTO(K)
      SNRT(K)=RPM/SQRT(STTO(K))
      SDHT(K)=DELHTS(K)/STTO(K)
      SETHC(K)=DELHTS(K)/THCR(K)
      RTHCR=SQRT(THCR(K))
      SNRTHC(K)=RPM/RTHCR
      SWRTED(K)=WGTO(K)*RTHCR*EPSI(K)/DELT(K)
      SPTPT2(K)=SPTO(K)/PTBAR(K)
      SPTPS2(K)=SPTO(K)/PS2(IP,K)
      ST2TTO(K)=TTBAR(K)/STTO(K)
      STRTTO(K)=TTR1A(IP,K)/STTO(K)
      UPS(K)=.5*(U1A(IP,K)+U2(IP,K))
      UPUPS(K)=UPS(K)*UPS(K)
      OUPUP=OUPUP+UPUPS(K)
      URS(K)=.5*(U1A(1,K)*DR(3,K)/DP1A(1,K)+U2(1,K)*DR(4,K)/DP2(1,K))
      URURS(K)=URS(K)*URS(K)
      OURUR=OURUR+URURS(K)
      ODELHT=ODELHT+DELHTS(K)
      IF (DELHSI(IP,K))14,14,15
   14 VIS(K)=1.
      GO TO 16
   15 VIS(K)=SQRT(2.*G*AJ*DELHSI(IP,K))
   16 UPVIS(K)=UPS(K)/VIS(K)
      URVIS(K)=URS(K)/VIS(K)
      PSIPS(K)=G*AJ*DELHTS(K)/(2.*UPUPS(K))
      PSIRS(K)=G*AJ*DELHTS(K)/(2.*URURS(K))
      RXP(K)=1.-(1.-(PS1(IP,K)/PTP(IP,K))**E5)/(1.-(PS2(IP,K)/
     1PTP(IP,K))**E5)
      VU1R=VU1(1,K)*DP1(1,K)/DR(2,K)
      V1R=SQRT(VU1R**2+VZ1(1,K)**2)
       PH1R=1./(1.-V1R**2/(2.*G*AJ*CP1(K)*TTO(1,K)*ETAS(1,K)))
      PTPS1R=PH1R**(GAM(2,K)/(GAM(2,K)-1.))*PTP(1,K)/PTO(1,K)
      RXR(K)=1.-(1.-(1./PTPS1R)**E5)/(1.-(PS2(1,K)/PTP(1,K))**E5)
      DBETAR(K)=(BET1A(1,K)+BET2E(1,K))*57.2958
      MIS(K)=V1(IP,K)/SQRT(GAM(2,K)*G*RG*TS1(IP,K))
      TS1R=TTO(1,K)-V1R**2/(2.*G*AJ*CP1(K))
      MIRS(K)=V1R/SQRT(GAM(2,K)*G*RG*TS1R)
      VU1AR=VU1A(1,K)*DP1A(1,K)/DR(3,K)
      V1AR=SQRT(VU1AR**2+VZ1A(1,K)**2)
      TS1AR=TTO(1,K)-V1AR**2/(2.*G*AJ*CP1A(K))
```

```
      RU1AR=VU1AR-U1A(1,K)*DR(3,K)/DP1A(1,K)
      R1AR=SQRT(RU1AR**2+VZ1A(1,K)**2)
      MR1AR(K)=R1AR/SQRT(GAM(3,K)*G*RG*TS1AR)
      VU2T=VU2(ISECT,K)*DP2(ISECT,K)/DT(4,K)
      V2T=SQRT(VU2T**2+VZ2(ISECT,K)**2)
      TS2T=TS2(ISECT,K)+(V2(ISECT,K)**2-V2T**2)/(2.*G*AJ*CP2(K))
      RU2T=VU2T+U2(ISECT,K)*DT(4,K)/DP2(ISECT,K)
      R2T=SQRT(RU2T**2+VZ2(ISECT,K)**2)
      MR2T(K)=R2T/SQRT(GAM(4,K)*G*RG*TS2T)
   17 CONTINUE
      IF (GAMF)4,4,7
    4 TAO=TAO/STG
      PAO=PAO/STG
      CALL GAMMA(PAO,TAO,FAIR,WAIR,GAMO)
      GO TO 8
    7 GAMO=GAMO/STG
    8 EO=(GAMO-1.)/GAMO
      CPO=RG/EO/AJ
      K=KSTG
      ODEHTI = 0.
      ODEHSI = 0.
      ODHATI = 0.
      DO 9 I=1,ISECT
      RW=WG2A(I,K)/WGT2A(K)
      ODEHTI = CPO*TTO(I,1)*(1.-(PT2A(I,K)/PTP(I,1))**EO)*RW+ODEHTI
      ODEHSI = CPO*TTO(I,1)*(1.-(PS2A(I,K)/PTP(I,1))**EO)*RW+ODEHSI
    9 ODHATI = CPO*TTO(I,1)*(1.-(PAT2A(I,K)/PTP(I,1))**EO)*RW+ODHATI
      OPSIP=G*AJ*ODELHT/(2.*OUPUP)
      OPSIR=G*AJ*ODELHT/(2.*OURUR)
      OWRTP=SWRTP(1)
      OWNED=SWRTED(1)*SNRTHC(1)/60.
      ONRTHC=SNRTHC(1)
      ONRT=SNRT(1)
      ODHT=ODELHT/TTIN
      OPTOT2=PTIN/PTBAR(KSTG)
      OPTOS2=PTIN/PS2(IP,KSTG)
      OPTAT2=PTIN/PAT2A(IP,KSTG)
      OETATT=ODELHT/ODEHTI
      OETATS=ODELHT/ODEHSI
      OETAAT=ODELHT/ODHATI
      OETHC=ODELHT/THCR(1)
C
C                           PRINT OUT FOR STAGE PERFORMANCE
      I=1
      WRITE(6,1000)NAME,TITLE,ICASE,ISCASE
 1000 FORMAT(1H1,21X,29HNASA TURBINE COMPUTER PROGRAM /6X,10A6/
     1  6X,10A6/ 30X,6HCASE  I3,1H.,I3/28X,17HSTAGE PERFORMANCE /16X
     27HSTAGE 1,6X,7HSTAGE 2,6X,7HSTAGE 3,6X,7HSTAGE 4/ )
      IF (KSTG-4)19,19,18
   18 KS=4
      GO TO 20
   19 KS=KSTG
   20 WRITE(6,1001)(STTO(K),K=I,KS)
```

```
 1001 FORMAT(1X,12H        TT 02X,F10.1,3X,F10.1,3X,F10.1,3X,F10.1)
      WRITE(6,1002)(SPTO(K),K=I,KS)
 1002 FORMAT(1X,12H        PT 02X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1003)(WGT1(K),K=I,KS)
 1003 FORMAT(1X,12H        WG 02X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1004)(DELHTS(K),K=I,KS)
 1004 FORMAT(1X,12H       DEL H2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1005)(SWRTP(K),K=I,KS)
 1005 FORMAT(1X,12H       WRT/P2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1006)(SDHT(K),K=I,KS)
 1006 FORMAT(1X,12H        DH/T2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1007)(SNRT(K),K=I,KS)
 1007 FORMAT(1X,12H         N/RT2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1008)(SETATT(K),K=I,KS)
 1008 FORMAT(1X,12H      ETA TT2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1009)(SETATS(K),K=I,KS)
 1009 FORMAT(1X,12H      ETA TS2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1010)(SETAAT(K),K=I,KS)
 1010 FORMAT(1X,12H      ETA AT2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1011)(PTOPS1(IP,K),K=I,KS)
 1011 FORMAT(1X,12H     PT0/PS12X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1012)(SPTPT2(K),K=I,KS)
 1012 FORMAT(1X,12H     PT0/PT2,2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1013)(SPTPS2(K),K=I,KS)
 1013 FORMAT(1X,12H     PT0/PS22X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1014)(PTRS2(IP,K),K=I,KS)
 1014 FORMAT(1X,12H   PTR1A/PS22X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1015)(ST2TTO(K),K=I,KS)
 1015 FORMAT(1X,12H     TT2/TT02X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1016)(STRTTO(K),K=I,KS)
 1016 FORMAT(1X,12H    TTR1/TT02X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1017)(PS1(IP,K),K=I,KS)
 1017 FORMAT(1X,12H        PS 12X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1018)(TTR1A(IP,K),K=I,KS)
 1018 FORMAT(1X,12H       TTR 12X,F10.1,3X,F10.1,3X,F10.1,3X,F10.1)
      WRITE(6,1019)(PTR1A(IP,K),K=I,KS)
 1019 FORMAT(1X,12H       PTR 12X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1020)(PS2(IP,K),K=I,KS)
 1020 FORMAT(1X,12H        PS 22X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1021)(TTBAR(K),K=I,KS)
 1021 FORMAT(1X,12H        TT 22X,F10.1,3X,F10.1,3X,F10.1,3X,F10.1)
      WRITE(6,1022)(PTBAR(K),K=I,KS)
 1022 FORMAT(1X,12H        PT 22X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1023)(UPVIS(K),K=I,KS)
 1023 FORMAT(1X,12H       UP/VI2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1024)(URVIS(K),K=I,KS)
 1024 FORMAT(1X,12H       UR/VI2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1025)(PSIPS(K),K=I,KS)
 1025 FORMAT(1X,12H       PSI P2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1026)(PSIRS(K),K=I,KS)
 1026 FORMAT(1X,12H       PSI R2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1027)(RXP(K),K=I,KS)
 1027 FORMAT(1X,12H        RX P2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
```

```
      WRITE(6,1028)(RXR(K),K=I,KS)
 1028 FORMAT(1X,12H        RX R2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1029)(SA0(K),K=I,KS)
 1029 FORMAT(1X,12H     ALPHA 02X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1030)(SIS(K),K=I,KS)
 1030 FORMAT(1X,12H    I STATOR2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1031)(SB1A(K),K=I,KS)
 1031 FORMAT(1X,12H     BETA 1A2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1032)(SIR(K),K=I,KS)
 1032 FORMAT(1X,12H     I ROTOR2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1033)(SA2(K),K=I,KS)
 1033 FORMAT(1X,12H     ALPHA 22X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1034)(DBETAR(K),K=I,KS)
 1034 FORMAT(1X,12H     DBETA R2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1035)(MIS(K),K=I,KS)
 1035 FORMAT(1X,12H          M 12X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1036)(MIRS(K),K=I,KS)
 1036 FORMAT(1X,12H        M1 RT2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1037)(MR1A(IP,K),K=I,KS)
 1037 FORMAT(1X,12H        MR 1A2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1038)(MR1AR(K), K=I,KS)
 1038 FORMAT(1X,12H     MR1A RT2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1039)(MR2(IP,K),K=I,KS)
 1039 FORMAT(1X,12H         MR 22X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1040)(MR2T(K),      K=I,KS)
 1040 FORMAT(1X,12H      MR2 TIP2X,F10.5,3X,F10.5,3X,F10.5,3X,F10.5)
      WRITE(6,1041)(SETHC(K),K=I,KS)
 1041 FORMAT(1X,12H      E/TH CR2X,F10.3,3X,F10.3,3X,F10.3,3X,F10.3)
      WRITE(6,1042)(SNRTHC(K),K=I,KS)
 1042 FORMAT(1X,12H     N/RTH CR2X,F10.1,3X,F10.1,3X,F10.1,3X,F10.1)
      WRITE(6,1043)(SWRTED(K),K=I,KS)
 1043 FORMAT(1X,12H   WRTHCRE/D2X,F10.1,3X,F10.1,3X,F10.1,3X,F10.1)
      IF (KSTG-KS)22,22,21
   21 WRITE(6,1045)NAME,TITLE,ICASE,ISCASE
 1045 FORMAT(1H1,21X,29HNASA TURBINE COMPUTER PROGRAM /6X,10A6/
     1  6X,10A6/ 30X,6HCASE   I3,1H.,I3/28X,17HSTAGE PERFORMANCE /16X
     27HSTAGE 5,6X,7HSTAGE 6,6X,7HSTAGE 7,6X,7HSTAGE 8/ )
      I=5
      KS=KSTG
      GO TO 20
   22 WRITE(6,1044)OPSIP,OPSIR,ODELHT,OWRTP,ONRT,ODHT,OPTOT2,
     1OPTOS2,OPTAT2,OETATT,OETATS,OETAAT,OWNED,ONRTHC,OETHC
 1044 FORMAT(//27X,18HOVRALL PERFORMANCE/6X,9HPSI P
     1F10.5, 5X,10HPSI R       F10.5, 5X9HDEL H     F10.5/6X,9HWRT/P
     2F10.5, 5X,10HN/RT        F10.5, 5X9HDELH/T    F10.5/6X,9HPTO/PT2
     3F10.5, 5X,10HPTO/PS2     F10.5, 5X9HPT/PAT2   F10.5/6X,9HETA TT
     4F10.5, 5X,10HETA TS      F10.5, 5X9HETA TAT   F10.5/6X,9HWNE/60D
     5F10.5, 5X,10HN/RTH CR    F10.5, 5X,9HE/TH CR   F10.5/)
      RETURN
      END
```

```
$IBFTC INST    FULIST,DECK,SDD                                          INST 000
CINSTG                                                                  INST 001
      SUBROUTINE INSTG                                                  INST 002
C     INTERSTAGE OUTPUT                                                 INST 003
C      NUMBER OF SECTORS IS THREE OR LESS,HUB AND CASING VALUES ARE     INST 004
C         CALCULATED AND PRINTED                                        INST 005
C     NUMBER OF SECTORS IS MORE THAN THREE,ONLY SECTOR PITCHLINE        INST 006
C         VALUES ARE PRINTED                                            INST 007
C                                                                       INST 008
      REAL MFSTOP                                                       INST 009
      LOGICAL PREVER                                                    INST 010
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,    INST 011
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,   INST 012
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC, INST 013
     3DELPR,PASS,IPC,LOPC,ISS                                           INST 014
C                                                                       INST 015
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)INST 016
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),INST 017
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),    INST 018
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)          INST 019
C                                                                       INST 020
      COMMON /SINPUT/                                                   INST 021
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,  INST 022
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),        INST 023
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),INST 024
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETARINST 025
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)INST 026
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(INST 027
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)INST 028
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)              INST 029
C                                                                       INST 030
      REAL M0                                                           INST 031
      COMMON /SSTA01/CP0(8),                    PS0(6,8),V0(6,8),TS0(6,INST 032
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8), INST 033
     2            DPDR1(6,8),SI(6,8),  CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)INST 034
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)                  INST 035
C                                                                       INST 036
      REAL MR1A                                                         INST 037
      COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),  CP1A(8),  INST 038
     1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTR1A(6INST 039
     2,8),MR1A(6,8)                                                     INST 040
C                                                                       INST 041
      COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2(8),INST 042
     1           PS2(6,8),PHI2(6,8)                                     INST 043
C                                                                       INST 044
      REAL MR2,M2      ,MF2                                             INST 045
      COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),RU2(6INST 046
     1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)        INST 047
C                                                                       INST 048
      REAL M2A,MF2A                                                     INST 049
      COMMON /SSTA2A/WG2A(6,8),WGT2A(8),VU2A(6,8),VZ2A(6,8),PS2A(6,8),  INST 050
     1ALF2A(6,8),TT2A(6,8),PT2A(6,8),TTBAR(8),PTBAR(8),STTO(8),SPTO(8), INST 051
```

```
     2M2A(6,8),MF2A(6,8),CP2A(8),V2A(6,8),TS2A(6,8),TAS(8),PAS(8),GAMS(8
     3),CPO(8),DELHVD(6,8)
C
      COMMON /SOVRAL/DELHT(6,8),DELHTI(6,8),DELHSI(6,8),DEHATI(6,8),
     1ETATT(6,8),ETATS(6,8),ETATAT(6,8)
C
      COMMON     STDPO(7),STPTO(7),STALF(7),STSI(7),STVO(7),STVUO(7),
     1STVZO(7),STTSO(7),STPSO(7),STDENO(7),STMO(7),STDP1(7),STALFE(7),
     2STDELA(7),STV1(7),STVU1(7),STVZ1(7),STTS1(7),STPS1(7),STDEN1(7),
     3STM1(7),ZWIINC(7),                            CPS(7),STDP1A(7),
     4STPTR1(7),STBET1(7),STR1(7),STR1A(7),STRU1A(7),STMR1A(7),STU1A(7),
     5STDP2(7),STBET2(7),SDBETA(7),SR2(7),SRU2(7),SMR2(7),SU2(7),RX(7),
     6STDELH(7),STPSI(7),SETATT(7),SETATS(7),SETAAT(7),RZWINC(7),
     7                  CPR(7),STPT2(7),STTT2(7),STV2(7),STVU2(7),
     8STALF2(7),STMF2(7),STTTR1(7),STVZ2(7),STTS2(7),STPS2(7),STDEN2(7),
     9STM2(7),STTTO(7),LJ,JJ,K
C
C
    1 DO 9 K=1,KSTG
      E1=(GAMS(K)-1.)/GAMS(K)
      E2=GAM(1,K)/(GAM(1,K)-1.)
     \E3=GAM(2,K)/(GAM(2,K)-1.)
      E4=GAM(3,K)/(GAM(3,K)-1.)
      E5=GAM(4,K)/(GAM(4,K)-1.)
      E6=GAM(5,K)/(GAM(5,K)-1.)
C     RELOCATE PITCHLINE VALUES
      J=ISECT+1
    4 DO 5 I=1,ISECT
      KS=J-I+1
      STTTO(KS)=TTO(KS-1,K)
      STDPO(KS)=DPO(KS-1,K)
      STPTO(KS)=PTP(KS-1,K)
      STALF(KS)=ALPHAO(KS-1,K)*57.2958
      STSI(KS)=SI(KS-1,K)*57.2958
      STVO(KS)=VO(KS-1,K)
      STVUO(KS)=VUO(KS-1,K)
      STVZO(KS)=VZO(KS-1,K)
      STTSO(KS)=TSO(KS-1,K)
      STPSO(KS)=PSO(KS-1,K)
      STDENO(KS)=144.*STPSO(KS)/STTSO(KS)/RG
      STMO(KS)=MO(KS-1,K)
      STDP1(KS)=DP1(KS-1,K)
      STALFE(KS)=ALF1E(KS-1,K)*57.2958
      STDELA(KS)=(ALPHAO(KS-1,K)+ALF1E(KS-1,K))*57.2958
      STV1(KS)=V1(KS-1,K)
      STVU1(KS)=VU1(KS-1,K)
      STVZ1(KS)=VZ1(KS-1,K)
      STTS1(KS)=TS1(KS-1,K)
      STPS1(KS)=PS1(KS-1,K)
      STDEN1(KS)=RHOS1(KS-1,K)
      STM1(KS)=V1(KS-1,K)/(SQRT(GAM(2,K)*G*RG*TS1(KS-1,K)))
       ZS =-2.*ALF1E(KS-1,K) -1.570796
      ZWIINC(KS)=COS(            ZS   )*(SIN(ALPHAO(KS-1,K))*COS(ALF1E(KS
```

```
     1-1,K))/(COS(ALPHAO(KS-1,K))*SIN(ALF1E(KS-1,K)))+1.)
      CPS(KS)=1.-(STVO(KS)/STV1(KS))**2
      STDP1A(KS)=DP1A(KS-1,K)
      STPTR1(KS)=PTR1A(KS-1,K)
      STTTR1(KS)=TTR1A(KS-1,K)
      STBET1(KS)=BET1A(KS-1,K)*57.2958
      STRI(KS)=RI(KS-1,K)*57.2958
      STR1A(KS)=R1A(KS-1,K)
      STRU1A(KS)=RU1A(KS-1,K)
      STMR1A(KS)=MR1A(KS-1,K)
      STU1A(KS)=U1A(KS-1,K)
      STDP2(KS)=DP2(KS-1,K)
      STBET2(KS)=BET2E(KS-1,K)*57.2958
      SDBETA(KS)=(BET1A(KS-1,K)+BET2E(KS-1,K))*57.2958
      SR2(KS)=R2(KS-1,K)
      SRU2(KS)=RU2(KS-1,K)
      SMR2(KS)=MR2(KS-1,K)
      SU2(KS)=U2(KS-1,K)
      RX(KS)=1.-(1.-(PS1(KS-1,K)/PTP(KS-1,K))**E1)/(1.-(PS2(KS-1,K)/
     1PTP(KS-1,K))**E1)
      STDELH(KS)=DELHT(KS-1,K)
      STPSI(KS)=2.*G*AJ*DELHT(KS-1,K)/(U1A(KS-1,K)*U1A(KS-1,K)
     1+U2(KS-1,K)*U2(KS-1,K))
      SETATT(KS)=ETATT(KS-1,K)
      SETATS(KS)=ETATS(KS-1,K)
      SETAAT(KS)=ETATAT(KS-1,K)
      ZR = -2.*BET2E(KS-1,K) -1.570796
      RZWINC(KS)=COS(          ZR )*(SIN(BET1A(KS-1,K))*COS(BET2E(KS-
     11,K))/(COS(BET1A(KS-1,K))*SIN(BET2E(KS-1,K)))+1.)
      CPR(KS)=1.-(STR1A(KS)/SR2(KS))**2
      STPT2(KS)=PT2A(KS-1,K)
      STTT2(KS)=TT2A(KS-1,K)
      STV2(KS)=V2A(KS-1,K)
      STVU2(KS)=VU2A(KS-1,K)
      STALF2(KS)=ALF2A(KS-1,K)*57.2958
      STMF2(KS)=MF2A(KS-1,K)
      STVZ2(KS)=VZ2A(KS-1,K)
      STPS2(KS)=PS2A(KS-1,K)
      STTS2(KS)=TS2A(KS-1,K)
      STM2(KS)=M2A(KS-1,K)
      STDEN2(KS)=144.*STPS2(KS)/STTS2(KS)/RG
    5 CONTINUE
      IF (ISECT-3)3,3,6
C           CALCULATE HUB VALUES
    3 LJ=1
      JJ=ISECT+2
      I=1
      L=1
      STDPO(1)=DR(1,K)
      R1=DPO(I,K)/DR(I,K)
      STDP1(1)=DR(2,K)
      R2=DP1(I,K)/DR(2,K)
      STDP1A(1)=DR(3,K)
```

```
      R3=DP1A(I,K)/DR(3,K)
      STDP2(L)=DR(4,K)
      R4=DP2(1,K)/DR(4,K)
      TALF=SIN(ALF1(I,K))*R3/COS(ALF1(I,K))
      R5=DP2A(I,K)/DR(5,K)
C        STATION    0         STATOR   INLET
   10 STTTO(L)=TTO(I,K)
      STPTO(L)=PTP(I,K)
      STVZO(L)=VZO(I,K)
      STVUO(L)=VUO(I,K)*R1
      STVO(L)=SQRT(VZO(I,K)*VZO(I,K)+STVUO(L)*STVUO(L))
      STTSO(L)=TTO(I,K)-STVO(L)*STVO(L)/(2.*G*AJ*CPO(K))
      STPSO(L)=PSO(I,K)*(STTSO(L)/TSO(I,K))**E2
      STDENO(L)=144.*STPSO(L)/(RG*STTSO(L))
       STALF(L)=ATAN2(STVUO(L),STVZO(L))*57.2958
      STSI(L)=STALF(L)-ATAN2(SIN( RADSD(I,K))*R1,COS( RADSD(I,K)))
     1*57.2958
      ASOH=SQRT(GAM(I,K)*G*RG*STTSO(L))
      STMO(L)=STVO(L)/ASOH
C        STATION    1         STATOR   EXIT
      STVZ1(L)=VZ1(I,K)
      STVU1(L)=VU1(I,K)*R2
      STV1(L)=SQRT(VZ1(I,K)*VZ1(I,K)+STVU1(L)*STVU1(L))
      STTS1(L)=TTO(I,K)-STV1(L)*STV1(L)/(2.*G*AJ*CP1(K))
      STPS1(L)=PS1(I,K)*(STTS1(L)/TS1(I,K))**E3
      STDEN1(L)=144.*STPS1(L)/STTS1(L)/RG
      STALFE(L)=ATAN2(STVU1(L),STVZ1(L))*57.2958
      STDELA(L)=STALF(L)+STALFE(L)
      AS1H=SQRT(GAM(2,K)*G*RG*STTS1(L))
      STM1(L)=STV1(L)/AS1H
      ZS =-2.*STALFE(L)/ 57.2958 -1.570796
      ZWIINC(L)= COS(    ZS)*(STVUO(L)*STVZ1(L) /(STVZO(L)*STVU1(L))+1.)
      CPS(L)=1.-(STVO(L)/STV1(L))**2
C        STATION    1A        ROTOR     INLET
      VU1AH=VU1A(I,K)*R3
      STRU1A(L)=VU1AH-U1A(I,K)/R3
      STBET1(L)=ATAN2(STRU1A(L),VZ1A(I,K))*57.2958
      T=TALF-(TALF/R3 - SIN(RADRD(I,K))/COS(RADRD(I,K)))/R3
      STRI(L)=STBET1(L)-ATAN2(     T,1.      )*57.2958
      STR1A(L)=SQRT(STRU1A(L)*STRU1A(L)+VZ1A(I,K)*VZ1A(I,K))
      V1A1AH=VZ1A(I,K)*VZ1A(I,K)+VU1AH*VU1AH
      DELTSH=(V1(I,K)*V1(I,K)-V1A1AH)/(2.*G*AJ*CP1A(K))
      TS1AH=TS1(I,K)+DELTSH
      STMR1A(L)=STR1A(L)/SQRT(GAM(3,K)*G*RG*TS1AH)
      TTRSH=1.+STMR1A(L)*STMR1A(L)*(GAM(3,K)-1.)/2.
      STTTR1(L)=TS1AH*TTRSH
      IF (RI(I,K))2,2,7
    2 EXPRI=EXPN
      GO TO 11
    7 EXPRI=EXPP
   11 PTRSH=(1.+(TTRSH-1.)*ETARR(I,K)*COS(RI(I,K))**EXPRI)**E4
      PS1AH=PS1(I,K)*(1.+DELTSH/TS1(I,K))**E4
      STPTR1(L)=PS1AH*PTRSH
```

```
      STU1A(L)=U1A(I,K)/R3
C           STATION    2          ROTOR     EXIT
      VU2H=VU2(I,K)*R4
      SRU2(L)=VU2H+U2(I,K)/R4
      STBET2(L)=ATAN2(SRU2(L),VZ2(I,K))*57.2958
      SDBETA(L)=STBET1(L)+STBET2(L)
      SR2(L)=SQRT(SRU2(L)*SRU2(L)+VZ2(I,K)*VZ2(I,K))
      V2V2H=VZ2(I,K)*VZ2(I,K)+VU2H*VU2H
      DELTSH=(V2(I,K)*V2(I,K)-V2V2H)/(2.*G*AJ*CP2(K))
      TS2H=TS2(I,K)+DELTSH
      SMR2(L)=SR2(L)/SQRT(GAM(4,K)*G*RG*TS2H)
      SU2(L)=U2(I,K)/R4
      PS2H=PS2(I,K)*(TS2H/TS2(I,K))**E5
      RX(L)=1.-(1.-(STPS1(L)/PTP(I,K))**E1)/(1.-(PS2H/PTP(I,K))**E1)
      STDELH(L)=(STU1A(L)*VU1AH+SU2(L)*VU2H)*TFR(I,K)/(    G*AJ)
      STPSI(L)=2.*G*AJ*STDELH(L)/(STU1A(L)**2+SU2(L)**2)
      SETATT(L)=STDELH(L)/DELHTI(I,K)
      SETATS(L)=STDELH(L)/DELHSI(I,K)
      SETAAT(L)=STDELH(L)/DEHATI(I,K)
      ZR =-2.*STBET2(L)/57.2958 -1.570796
      RZWINC(L)=COS(    ZR)*(STRU1A(L)*VZ2(I,K) /(VZ1A(I,K)*SRU2(L))+1.)
      CPR(L)=1.-(STR1A(L)/SR2(L))**2
      STPT2(L)=PT2A(I,K)
      STTT2(L)=TT2A(I,K)
      STVZ2(L)=VZ2A(I,K)
      STVU2(L)=VU2A(I,K)*R5
      V2A2AH=STVU2(L)**2+VZ2A(I,K)**2
      STV2(L)=SQRT(V2A2AH)
      STALF2(L)=ATAN2(STVU2(L),VZ2A(I,K))*57.2958
      DELTS2=(V2A(I,K)**2-V2A2AH)/(2.*G*AJ*CP2A(K))
      STTS2(L)=TS2A(I,K)+DELTS2
      STPS2(L)=PS2A(I,K)*(1.+DELTS2/TS2A(I,K))**E6
      STDEN2(L)=144.*STPS2(L)/(RG*STTS2(L))
      STM2(L)=STV2(L)/SQRT(GAM(5,K)*G*RG*STTS2(L))
      STMF2(L)=STM2(L)*COS(STALF2(L)/57.2958)
      IF (L.GT.1) GO TO 8
C          CALCULATE    TIP    VALUES
      I=ISECT
      L=ISECT+2
      STDPO(L)=DT(1,K)
      R1=DPO(I,K)/DT(1,K)
      STDP1(L)=DT(2,K)
      R2=DP1(I,K)/DT(2,K)
      STDP1A(L)=DT(3,K)
      R3=DP1A(I,K)/DT(3,K)
      STDP2(L)=DT(4,K)
      R4=DP2(I,K)/DT(4,K)
      TALF=SIN(ALF1(I,K))*R3/COS(ALF1(I,K))
      R5=DP2A(I,K)/DT(5,K)
      GO TO 10
    6 LJ=2
      JJ=ISECT+1
    8 CALL WOUT
```

```
    9 CONTINUE                                                          INST 264
      RETURN                                                            INST 265
      END                                                               INST 266
```

```
$IBFTC WOUT    FULIST,DECK,SDD
CWOUT
      SUBROUTINE WOUT
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      COMMON    STDP0(7),STPT0(7),STALF(7),STSI(7),STV0(7),STVU0(7),
     1STVZ0(7),STTS0(7),STPS0(7),STDEN0(7),STM0(7),STDP1(7),STALFE(7),
     2STDELA(7),STV1(7),STVU1(7),STVZ1(7),STTS1(7),STPS1(7),STDEN1(7),
     3STM1(7),ZWIINC(7),                          CPS(7),STDP1A(7),
     4STPTR1(7),STBET1(7),STR1(7),STR1A(7),STRU1A(7),STMR1A(7),STU1A(7),
     5STDP2(7),STBET2(7),SDBETA(7),SR2(7),SRU2(7),SMR2(7),SU2(7),RX(7),
     6STDELH(7),STPSI(7),SETATT(7),SETATS(7),SETAAT(7),RZWINC(7),
     7                    CPR(7),STPT2(7),STTT2(7),STV2(7),STVU2(7),
     8STALF2(7),STMF2(7),STTTR1(7),STVZ2(7),STTS2(7),STPS2(7),STDEN2(7),
     9STM2(7),STTT0(7),LJ,JJ,K
C
C          PRINT  OUT  FOR  INTERSTAGE  DATA
    8 WRITE(6,1000)NAME,TITLE,ICASE,ISCASE
 1000 FORMAT(1H1,20X29HNASA TURBINE COMPUTER PROGRAM/6X10A6/6X10A6/30X
     15HCASE I3,1H.I3/24X23HINTER-STAGE PERFORMANCE//)
      WRITE(6,1001)K,(STDP0(I),I=LJ,JJ)
 1001 FORMAT(5X5HSTA 02X12HSTATOR INLET10X5HSTAGEI3,1H./4X6HDIAM 02X,
     16F10.3)
      WRITE(6,1002)(STTT0(I),I=LJ,JJ)
 1002 FORMAT (10H      TT 0,2X,6F10.1)
      WRITE(6,1003)( STPT0(I),I=LJ,JJ)
 1003 FORMAT (10H      PT 0,2X,6F10.3)
      WRITE(6,1004)( STALF(I),I=LJ,JJ)
 1004 FORMAT (10H   ALPHA 0,2X,6F10.3)
      WRITE(6,1005)(  STSI(I),I=LJ,JJ)
 1005 FORMAT (10H  I STATOR,2X,6F10.3)
```

```
      WRITE(6,1006)(   STVO(I),I=LJ,JJ)
 1006 FORMAT (10H        V 0,2X,6F10.3)
      WRITE(6,1007)( STVUO(I),I=LJ,JJ)
 1007 FORMAT (10H       VU 0,2X,6F10.3)
      WRITE(6,1008)( STVZO(I),I=LJ,JJ)
 1008 FORMAT (10H       VZ 0,2X,6F10.3)
      WRITE(6,1009)( STTSO(I),I=LJ,JJ)
 1009 FORMAT (10H       TS 0,2X,6F10.1)
      WRITE(6,1010)( STPSO(I),I=LJ,JJ)
 1010 FORMAT (10H       PS 0,2X,6F10.3)
      WRITE(6,1011)(STDENO(I),I=LJ,JJ)
 1011 FORMAT (10H     DENS 0,2X,6F10.5)
      WRITE(6,1012)(   STMO(I),I=LJ,JJ)
 1012 FORMAT (10H        M 0,2X,6F10.3)
      WRITE(6,1013)(STDP1(I),I=LJ,JJ)
 1013 FORMAT(/5X5HSTA 12X11HSTATOR EXIT/4X6HDIAM 12X,6F10.3)
      WRITE(6,1014)(STALFE(I),I=LJ,JJ)
 1014 FORMAT (10H    ALPHA 1,2X,6F10.3)
      WRITE(6,1015)(STDELA(I),I=LJ,JJ)
 1015 FORMAT (10H      DEL A,2X,6F10.3)
      WRITE(6,1016)(   STV1(I),I=LJ,JJ)
 1016 FORMAT (10H        V 1,2X,6F10.3)
      WRITE(6,1017)( STVU1(I),I=LJ,JJ)
 1017 FORMAT (10H       VU 1,2X,6F10.3)
      WRITE(6,1018)( STVZ1(I),I=LJ,JJ)
 1018 FORMAT (10H       VZ 1,2X,6F10.3)
      WRITE(6,1019)( STTS1(I),I=LJ,JJ)
 1019 FORMAT (10H       TS 1,2X,6F10.1)
      WRITE(6,1064)( STPS1(I),I=LJ,JJ)
 1064 FORMAT (10H       PS 1,2X,6F10.3)
      WRITE(6,1020)(STDEN1(I),I=LJ,JJ)
 1020 FORMAT (10H     DENS 1,2X,6F10.5)
      WRITE(6,1021)(   STM1(I),I=LJ,JJ)
 1021 FORMAT (10H        M 1,2X,6F10.5)
      WRITE(6,1022)(ZWIINC(I),I=LJ,JJ)
 1022 FORMAT (10H    ZWI INC,2X,6F10.3)
      WRITE(6,1026)(    CPS(I),I=LJ,JJ)
 1026 FORMAT (10H       CP S,2X,6F10.3)
      WRITE(6,1000)NAME,TITLE,ICASE,ISCASE
      WRITE(6,1028)K,(STDP1A(I),I=LJ,JJ)
 1028 FORMAT(4X6HSTA 1A2X11HROTOR INLET10X5HSTAGEI3,1H./3X7HDIAM 1A2X,
     16F10.3)
      WRITE(6,1027)(STPTR1(I),I=LJ,JJ)
 1027 FORMAT (10H     PTR 1A,2X,6F10.3)
      WRITE(6,1029)(STTTR1(I),I=LJ,JJ)
 1029 FORMAT (10H     TTR 1A,2X,6F10.1)
      WRITE(6,1030)(STBET1(I),I=LJ,JJ)
 1030 FORMAT (10H    BETA 1A,2X,6F10.3)
      WRITE(6,1031)(   STRI(I),I=LJ,JJ)
 1031 FORMAT (10H    I ROTOR,2X,6F10.3)
      WRITE(6,1032)( STR1A(I),I=LJ,JJ)
 1032 FORMAT (10H       R 1A,2X,6F10.3)
      WRITE(6,1033)(STRU1A(I),I=LJ,JJ)
```

```
1033 FORMAT (10H      RU 1A,2X,6F10.3)
     WRITE(6,1034)(STMR1A(I),I=LJ,JJ)
1034 FORMAT (10H      MR 1A,2X,6F10.5)
     WRITE(6,1035)( STU1A(I),I=LJ,JJ)
1035 FORMAT (10H       U 1A,2X,6F10.3)
     WRITE(6,1037)(STDP2(I),I=LJ,JJ)
1037 FORMAT(/5X5HSTA 22X10HROTOR EXIT/4X6HDIAM 22X,6F10.3)
     WRITE(6,1036)(STBET2(I),I=LJ,JJ)
1036 FORMAT (10H    BETA 2,2X,6F10.3)
     WRITE(6,1038)(SDBETA(I),I=LJ,JJ)
1038 FORMAT (10H     DBETA,2X,6F10.3)
     WRITE(6,1039)(    SR2(I),I=LJ,JJ)
1039 FORMAT (10H       R 2,2X,6F10.3)
     WRITE(6,1040)(   SRU2(I),I=LJ,JJ)
1040 FORMAT (10H      RU 2,2X,6F10.3)
     WRITE(6,1041)(   SMR2(I),I=LJ,JJ)
1041 FORMAT (10H      MR 2,2X,6F10.5)
     WRITE(6,1042)(    SU2(I),I=LJ,JJ)
1042 FORMAT (10H       U 2,2X,6F10.3)
     WRITE(6,1043)(     RX(I),I=LJ,JJ)
1043 FORMAT (10H        RX,2X,6F10.5)
     WRITE(6,1044)(STDELH(I),I=LJ,JJ)
1044 FORMAT (10H      DELH,2X,6F10.3)
     WRITE(6,1045)( STPSI(I),I=LJ,JJ)
1045 FORMAT (10H     PSI P,2X,6F10.5)
     WRITE(6,1046)(SETATT(I),I=LJ,JJ)
1046 FORMAT (10H    ETA TT,2X,6F10.5)
     WRITE(6,1047)(SETATS(I),I=LJ,JJ)
1047 FORMAT (10H    ETA TS,2X,6F10.5)
     WRITE(6,1048)(SETAAT(I),I=LJ,JJ)
1048 FORMAT (10H    ETA AT,2X,6F10.5)
     WRITE(6,1049)(RZWINC(I),I=LJ,JJ)
1049 FORMAT (10H   ZWI INC,2X,6F10.3)
     WRITE(6,1065)(    CPR(I),I=LJ,JJ)
1065 FORMAT (10H       CPR ,2X,6F10.3)
     WRITE(6,1053)( STPT2(I),I=LJ,JJ)
1053 FORMAT (10H     PT 2A,2X,6F10.3)
     WRITE(6,1054)( STTT2(I),I=LJ,JJ)
1054 FORMAT (10H     TT 2A,2X,6F10.1)
     WRITE(6,1055)(  STV2(I),I=LJ,JJ)
1055 FORMAT (10H      V 2A,2X,6F10.3)
     WRITE(6,1056)( STVU2(I),I=LJ,JJ)
1056 FORMAT (10H     VU 2A,2X,6F10.3)
     WRITE(6,1057)(STALF2(I),I=LJ,JJ)
1057 FORMAT (10H  ALPHA 2A,2X,6F10.3)
     WRITE(6,1058)( STMF2(I),I=LJ,JJ)
1058 FORMAT (10H     MF 2A,2X,6F10.5)
     WRITE(6,1059)( STVZ2(I),I=LJ,JJ)
1059 FORMAT (10H     VZ 2A,2X,6F10.3)
     WRITE(6,1060)( STTS2(I),I=LJ,JJ)
1060 FORMAT (10H     TS 2A,2X,6F10.1)
     WRITE(6,1061)( STPS2(I),I=LJ,JJ)
1061 FORMAT (10H     PS 2A,2X,6F10.3)
```

```
      WRITE(6,1062)(STDEN2(I),I=LJ,JJ)
 1062 FORMAT (10H     DENS 2A,2X,6F10.5)
      WRITE(6,1063)(  STM2(I),I=LJ,JJ)
 1063 FORMAT (10H        M 2A,2X,6F10.5)
      RETURN
      END
```

APPENDIX 3X

```
$IBFTC DIGT    FULIST,DECK,SDD
CDIAGT
      SUBROUTINE DIAGT(M)
C
      REAL MFSTOP
      LOGICAL PREVER
      COMMON /SNTCP/G,AJ,PRPC,ICASE,PREVER,MFSTOP,JUMP,LOPIN,ISCASE,
     1KN,GAMF,IP,SCRIT,PTRN,ISECT,KSTG,WTOL,RHOTOL,PRTOL,TRLOOP,LSTG,
     2LBRC,IBRC,ICHOKE,ISORR,CHOKE,PTOPS1(6,8),PTRS2(6,8),TRDIAG,SC,RC,
     3DELPR,PASS,IPC,LOPC,ISS
C
      COMMON /SINIT/H1(6,8),H2(6,8),DP0(6,8),DP1(6,8),DP1A(6,8),DP2(6,8)
     1,DP2A(6,8),CSALF1(6,8),ALF1(6,8),CSBET2(6,8),BET2(6,8),RADSD(6,8),
     2RADRD(6,8),ANN1(6,8),ANN2(6,8),ANN2A(6,8),ANN1A(6,8),U1A(6,8),
     3U2(6,8),ANN0(6,8),PT0(6,8),TT0(6,8),ALPHA0(6,8),PTP(6,8)
C
      COMMON /SINPUT/
     1PTPS,PTIN,TTIN,WAIR,FAIR,DELC,DELL,DELA,AACS,VCTD,STG,SECT,EXPN,
     2EXPP,EXPRE,RG,RPM,PAF,SLI,STGCH,ENDJOB,NAME(10),TITLE(10),
     3PCNH(6),GAM(6,8),DR(6,8),DT(6,8),RWG(6,8),ALPHAS(6,8),ALPHA1(6,8),
     4ETARS(6,8),ETAS(6,8),CFS(6,8),ANDO(6,8),BETA1(6,8),BETA2(6,8),ETAR
     5R(6,8),ETAR(6,8),CFR(6,8),TFR(6,8),ANDOR(6,8),OMEGAS(6,8),AS0(6,8)
     6,ASMP0(6,8),ACMN0(6,8),A1(6,8),A2(6,8),A3(6,8),A4(6,8),A5(6,8),A6(
     76,8),OMEGAR(6,8),BSIA(6,8),BSMPIA(6,8),BCMNIA(6,8),B1(6,8),B2(6,8)
     8,B3(6,8),B4(6,8),B5(6,8),B6(6,8),SESTHI(8),RERTHI(8)
C
      REAL M0
      COMMON /SSTA01/CP0(8),                    PS0(6,8),V0(6,8),TS0(6,
     18),VU0(6,8),VZ0(6,8),RHOS0(6,8),PS1(6,8),WGT1(8),TA1(8),WG1(6,8),
     2              DPDR1(6,8),SI(6,8),   CP1(8),PHI1(6,8),TS1(6,8),V1(6,8)
     3,RHOS1(6,8),ALF1E(6,8),VU1(6,8),VZ1(6,8),M0(6,8)
C
      REAL MR1A
      COMMON /SSTA1A/VU1A(6,8),WG1A(6,8),WGT1A(8),VZ1A(6,8),   CP1A(8),
     1PS1A(6,8),RU1A(6,8),R1A(6,8),BET1A(6,8),RI(6,8),TTR1A(6,8),PTR1A(6
     2,8),MR1A(6,8)
C
      COMMON /SSTA2/V2(6,8),TTR2(6,8),PTR2(6,8),WG2(6,8),WGT2(8),TA2(8),
     1              PS2(6,8),PHI2(6,8)
C
      REAL MR2,M2      ,MF2
      COMMON /SFLOW2/TS2(6,8),CP2(8),R2(6,8),RHOS2(6,8),BET2E(6,8),RU2(6
     1,8),VU2(6,8),DPDR2(6,8),VZ2(6,8),MR2(6,8),MF2(6,8),M2(6,8)
C
      REAL M2A,MF2A
      COMMON /SSTA2A/WG2A(6,8),WGT2A(8),VU2A(6,8),VZ2A(6,8),PS2A(6,8),
     1ALF2A(6,8),TT2A(6,8),PT2A(6,8),TTBAR(8),PTBAR(8),STT0(8),SPT0(8),
     2M2A(6,8),MF2A(6,8),CP2A(8),V2A(6,8),TS2A(6,8),TAS(8),PAS(8),GAMS(8
     3),CPS(8),DELHVD(6,8)
C
      WRITE(6,1000)NAME,TITLE
 1000 FORMAT(1H1,5X,10A6/6X,10A6/20X,29HNASA TURBINE COMPUTER PROGRAM/
```

```
     131X,10HDIAGNOSTIC)
      IF (M.EQ.0)  GO TO 10
      GO TO (10,19,11,12,13),M
   10 DO 14 K=1,KN
      WRITE(6,1001)K,CPO(K),GAM(1,K)
 1001 FORMAT(9X,1HK,I5,9X,3HCPO,F10.3,9X,5HGAMMA,F10.5)
      WRITE(6,1002) (PTP(I,K),I=1,ISECT)
 1002 FORMAT(3X,6H   PTP,6F10.3)
      WRITE(6,1003) (PTO(I,K),I=1,ISECT)
 1003 FORMAT(3X,6H   PTO,6F10.3)
      WRITE(6,1004) (PSO(I,K),I=1,ISECT)
 1004 FORMAT(3X,6H   PSO,6F10.3)
      WRITE(6,1005) (TTO(I,K),I=1,ISECT)
 1005 FORMAT(3X,6H   TTO,6F10.1)
      WRITE(6,1006) (TSO(I,K),I=1,ISECT)
 1006 FORMAT(3X,6H   TSO,6F10.1)
      WRITE(6,1007) (VO(I,K),I=1,ISECT)
 1007 FORMAT(3X,6H    VO,6F10.3)
      WRITE(6,1008) (ALPHAO(I,K),I=1,ISECT)
 1008 FORMAT(3X,6HALPHAO,6F10.3)
   14 WRITE(6,1009) (SI(I,K),I=1,ISECT)
      IF (M.EQ.0)  GO TO 19
      GO TO 18
   19 DO 20 K=1,KN
 1009 FORMAT(3X,6H    SI,6F10.3)
      WRITE(6,1010) K,CP1(K),GAM(2,K)
 1010 FORMAT(9X,1HK,I5,9X,3HCP1,F10.3,9X,5HGAMMA,F10.5)
      WRITE(6,1011) (PS1(I,K),I=1,ISECT)
 1011 FORMAT(3X,6H   PS1,6F10.3)
      WRITE(6,1012) (DPDR1(I,K),I=1,ISECT)
 1012 FORMAT(3X,6H DPDR1,6F10.5)
      WRITE(6,1013) (TS1(I,K),I=1,ISECT)
 1013 FORMAT (3X,6H   TS1,6F10.1)
      WRITE(6,1014) (WG1(I,K),I=1,ISECT)
 1014 FORMAT(3X,6H   WG1,6F10.3)
      WRITE(6,1015) (V1(I,K),I=1,ISECT)
 1015 FORMAT(3X,6H    V1,6F10.3)
      WRITE(6,1016) (ALF1E(I,K),I=1,ISECT)
 1016 FORMAT (3X,6H ALF1E,6F10.3)
   20 WRITE(6,1017) (ALF1(I,K),I=1,ISECT)
 1017 FORMAT(3X,6H  ALF1,6F10.3)
      IF (M.EQ.0)  GO TO 11
      GO TO 18
   11 DO 15 K=1,KN
      WRITE(6,1018) K,CP1A(K),GAM(3,K)
 1018 FORMAT(9X,1HK,I5,9X,4HCP1A,F10.3,8X,5HGAMMA,F10.5)
      WRITE(6,1019) (PTR1A(I,K),I=1,ISECT)
 1019 FORMAT(3X,6H PTR1A,6F10.3)
      WRITE(6,1020) (PS1A(I,K),I=1,ISECT)
 1020 FORMAT(3X,6H  PS1A,6F10.3)
      WRITE(6,1021) (TTR1A(I,K),I=1,ISECT)
 1021 FORMAT(3X,6H TTR1A,6F10.1)
      WRITE(6,1022) (WG1A(I,K),I=1,ISECT)
```

```
1022 FORMAT (3X,6H  WG1A,6F10.3)
     WRITE(6,1023) (R1A(I,K),I=1,ISECT)
1023 FORMAT (3X,6H   R1A,6F10.3)
     WRITE(6,1024) (BET1A(I,K),I=1,ISECT)
1024 FORMAT (3X,6H BET1A,6F10.3)
  15 WRITE(6,1025) (RI(I,K),I=1,ISECT)
1025 FORMAT (3X,6H    RI,6F10.3)
     IF (M.EQ.0)  GO TO 12
     GO TO 18
  12 DO 16 K=1,KN
     WRITE(6,1026)K,CP2(K),GAM(3,K)
1026 FORMAT(9X,1HK,I5,9X,3HCP2,F10.3,9X,5HGAMMA,F10.5)
     WRITE(6,1027) (PTR2(I,K),I=1,ISECT)
1027 FORMAT (3X,6H  PTR2,6F10.3)
     WRITE(6,1028) (PS2(I,K),I=1,ISECT)
1028 FORMAT (3X,6H   PS2,6F10.3)
     WRITE(6,1029) (DPDR2(I,K),I=1,ISECT)
1029 FORMAT (3X,6H DPDR2,6F10.5)
     WRITE(6,1030) (TTR2(I,K),I=1,ISECT)
1030 FORMAT (3X,6H  TTR2,6F10.1)
     WRITE(6,1031) (TS2(I,K),I=1,ISECT)
1031 FORMAT (3X,6H   TS2,6F10.1)
     WRITE(6,1032) (WG2(I,K),I=1,ISECT)
1032 FORMAT (3X,6H   WG2,6F10.3)
     WRITE(6,1033) (R2(I,K),I=1,ISECT)
1033 FORMAT (3X,6H    R2,6F10.3)
     WRITE(6,1034) (BET2E(I,K),I=1,ISECT)
1034 FORMAT (3X,6H BET2E,6F10.3)
  16 WRITE(6,1035) (BET2(I,K),I=1,ISECT)
1035 FORMAT (3X,6H  BET2,6F10.3)
     IF (M.EQ.0)  GO TO 13
     GO TO 18
  13 DO 17 K=1,KN
     L=K +1
     WRITE(6,1036)K,CP2A(K),GAM(5,K)
1036 FORMAT(9X,1HK,I5,9X,4HCP2A,F10.3,8X,5HGAMMA,F10.5)
     WRITE(6,1037) (PT2A(I,K),I=1,ISECT)
1037 FORMAT (3X,6H  PT2A,6F10.3)
     WRITE(6,1038) (PS2A(I,K),I=1,ISECT)
1038 FORMAT (3X,6H  PS2A,6F10.3)
     WRITE(6,1039) (TT2A(I,K),I=1,ISECT)
1039 FORMAT (3X,6H  TT2A,6F10.1)
     WRITE(6,1040) (TS2A(I,K),I=1,ISECT)
1040 FORMAT (3X,6H  TS2A,6F10.1)
     WRITE(6,1041) (WG2A(I,K),I=1,ISECT)
1041 FORMAT (3X,6H  WG2A,6F10.3)
     WRITE(6,1042) (V2A(I,K),I=1,ISECT)
1042 FORMAT (3X,6H   V2A,6F10.3)
     WRITE(6,1043) (ALF2A(I,K),I=1,ISECT)
1043 FORMAT (3X,6H ALF2A,6F10.3)
     WRITE(6,1044) (SI(I,K),I=1,ISECT)
1044 FORMAT (3X,6H    SI,6F10.3)
     WRITE(6,1045) L,CPS(K),GAMS(K)
```

```
1045 FORMAT(9X,1HL,I5,9X,3HCPS,F10.3,9X,5HGAMMA,F10.5)
     WRITE(6,1046) (PTP(I,L),I=1,ISECT)
1046 FORMAT (3X,6H   PTP,6F10.3)
     WRITE(6,1047) (PTO(I,L),I=1,ISECT)
1047 FORMAT (3X,6H   PTO,6F10.3)
  17 WRITE(6,1048) (TTO(I,L),I=1,ISECT)
1048 FORMAT (3X,6H   TTO,6F10.1)
  18 CONTINUE
     RETURN
     END
```

APPENDIX 3Y

```
$IBFTC PHIM    FULIST,DECK,SDD
CPHIM
      SUBROUTINE PHIM(EXI,ETA,TR,PR)
      A = EXI-.5
      B = -(EXI+(1.-ETA)/2.)
      C = ETA/2.
      X = (-B -SQRT(B**2 -4.*A*C))/(2.*A)
      TR = ETA/(ETA-X)
      PR = TR**EXI
      RETURN
      END
```

APPENDIX 4

TURBINE COMPUTER PROGRAM
STANDARD OPTION
INPUT SHEET

START ALL INPUT CARDS IN COLUMN 2

```
NAME 1      NASA   TWO    STAGE   TURBINE
TITLE 1     1.00   5041   -8   DEG.   LOSS   PROFILE   .98   .946,   .977   .90,
$DATAIN   STAGE=   1,
STGCH=       1.0,
TTIN=      700    ,PTIN=    17.14   ,WAIR=     0    ,FAIR=     0   ,
PTPS=      1.1    ,DELC=       .1   ,DELL=    .1    ,DELA=    .1   ,
STG=         2    ,SECT=        5   ,EXPN=     3    ,EXPP=     3   ,
RG=      53.35    ,PAF=         0   ,SLI       0    ,AACS=   1.0   ,
RPM=      5041    ,VCTD=        1   ,ENDJOB=   0    ,ENDSTG=   0   ,
```

INLET RADIAL PROFILE

PCNH=	.2 ,	.2 ,	.2 ,	.2 ,	.2 ,

AXIAL STATIONS

	STA. 0	STA. 1	STA.1A	STA. 2	STA. 2A
GAMG=	1.4 ,	,	,	,	→ ,
DR=	19.11 ,	19.11 ,	18.969 ,	18.406 ,	18.265 ,
DT=	28.0 ,	28.0 ,	28.141 ,	28.704 ,	28.845 ,
RWG=	1.0 ,	,	,	,	→ ,

STATOR RADIAL DISTRIBUTIONS

	ROOT	PITCH	TIP	(FOR THREE SECTORS)	
SDIA=	0. ,	,	,	,	→ ,
SDEA=	,	,	,	,	,
SREC=	1.0 ,	,	,	,	→ ,
SETA=	.97 ,	.98 ,	.98 ,	.98 ,	.97 ,
SCF=	.977 ,	,	,	,	→ ,
SPA=	22.140 ,	26.035 ,	30.135 ,	34.194 ,	38.499 ,
SESTH=	1.0 ,				→

ROTOR RADIAL DISTRIBUTIONS

	ROOT	PITCH	TIP	(FOR THREE SECTORS)	
RDIA=	50.6 ,	44.9 ,	38.1 ,	30.2 ,	20.9 ,
RDEA=	,	,	,	,	,
RREC=	1.0 ,	,	,	,	→ ,
RETA=	.919 ,	.946 ,	.946 ,	.946 ,	.919 ,
RCF=	.95 ,	,	,	,	→ ,
RPA=	33.408 ,	36.352 ,	38.976 ,	41.280 ,	43.008 ,
RTF=	1.0 ,	,	,	,	→ ,
RERTH=	1.01 ,				

```
ENDJOB=    0   ,      ENDJOB=1.0 IF LAST CASE
ENDSTG=    0   $      ENDSTG=1.0 IF LAST STAGE
```

TURBINE COMPUTER PROGRAM
STANDARD OPTION
INPUT SHEET

START ALL INPUT CARDS IN COLUMN 2

```
NAME 1
TITLE 1
$DATAIN  STAGE= 2 ,
STGCH=          ,
TTIN=           ,PTIN=          ,WAIR=          ,FAIR=          ,
PTPS=           ,DELC=          ,DELL=          ,DELA=          ,
STG=            ,SECT=          ,EXPN=          ,EXPP=          ,
RG=             ,PAF=           ,SLI            ,AACS=          ,
RPM=            ,VCTD=          ,ENDJOB=        ,ENDSTG=        ,

                      INLET RADIAL PROFILE
PCNH=           ,         ,          ,          ,

                         AXIAL STATIONS
          STA. 0     STA. 1     STA.1A    STA. 2     STA. 2A

GAMG=     1.4     ,          ,          ,          ,          ,
DR=       18.265  , 17.814   , 17.673   , 17.110   , 17.110   ,
DT=       28.845  , 29.296   , 29.437   , 30.00    , 30.00    ,
RWG=      1.0     ,          ,          ,          ,          ,

                    STATOR RADIAL DISTRIBUTIONS
          ROOT      PITCH         TIP          (FOR THREE SECTORS)

SDIA=     25.0    ,  22.4    ,  20.2    ,  18.3    ,  16.6    ,
SDEA=             ,          ,          ,          ,          ,
SREC=     1.0     ,          ,          ,          ,          ,
SETA=     .97     ,  .98     ,  .98     ,  .98     ,  .97     ,
SCF=      .925    ,          ,          ,          ,          ,
SPA=      30.42   ,  36.855  ,  43.485  ,  50.765  ,  58.240  ,
SESTH=    1.01    ,

                    ROTOR RADIAL DISTRIBUTIONS
          ROOT      PITCH         TIP          (FOR THREE SECTORS)

RDIA=     36.6    ,  26.9    ,  16.1    ,  4.6     ,  -6.7    ,
RDEA=             ,          ,          ,          ,          ,
RREC=     1.0     ,          ,          ,          ,          ,
RETA=     .919    ,  .946    ,  .946    ,  .946    ,  .919    ,
RCF=      .90     ,          ,          ,          ,          ,
RPA=      43.35   ,  48.15   ,  52.35   ,  55.75   ,  58.55   ,
RTF=      1.0     ,          ,          ,          ,          ,
RERTH=    1.01    ,

ENDJOB=   0       ,        ENDJOB=1.0 IF LAST CASE

ENDSTG=   1       $        ENDSTG=1.0 IF LAST STAGE
```

APPENDIX 5

```
                 TURBINE COMPUTER PROGRAM
   NAME-    NASA  TWO STAGE TURBINE
   TITLE-   1.00 5041 -8 DEG. LOSS PROFILE .98 .946, .977 .90,
*DATAIN
  TTIN=    700.000    PTIN=      17.140  WAIR=       0.     FAIR=        0.
  PTPS=      1.100    DELC=       0.100  DELL=       0.100 DELA=         0.100
   STG=      2.000    SECT=       5.000  EXPN=       3.000 EXPP=         3.000
    RG=     53.350     PAF=       0.      SLI=       0.     AACS=        1.000
   RPM=   5041.000    VCTD=       1.    EXPRE=       0.
ENDSTG=      0.     ENDJOB=      -0.000
                  INLET RADIAL  PROFILES
PCNH=    0.200       0.200       0.200       0.200       0.200       -0.000
```

STANDARD OPTION

STAGE= 1

AXIAL STATIONS

	STA. 0	STA. 1	STA.1A	STA. 2	STA. 2A	
GAMG=	1.400	1.400	1.400	1.400	1.400	0.
DR=	19.110	19.110	18.969	18.406	18.265	0.
DT=	28.000	28.000	28.141	28.704	28.845	0.
RWG=	1.000	1.000	1.000	1.000	1.000	0.

STATOR RADIAL DISTRIBUTIONS

	ROOT	PITCH	TIP	(FOR THREE	SECTORS)	
SDIA=	0.	0.	0.	0.	0.	0.
SDEA=	0.	0.	0.	0.	0.	0.
SREC=	1.000	1.000	1.000	1.000	1.000	0.
SETA=	0.970	0.980	0.980	0.980	0.970	0.
SCF=	0.977	0.977	0.977	0.977	0.977	0.
SPA=	22.140	26.035	30.135	34.194	38.499	0.
SESTH=	1.000					

ROTOR RADIAL DISTRIBUTIONS

RDIA=	50.600	44.900	38.100	30.200	20.900	0.
RDEA=	0.	0.	0.	0.	0.	0.
RREC=	1.000	1.000	1.000	1.000	1.000	0.
RETA=	0.919	0.946	0.946	0.946	0.919	0.
RCF=	0.950	0.950	0.950	0.950	0.950	0.
RTF=	1.000	1.000	1.000	1.000	1.000	0.
RPA=	33.408	36.352	38.976	41.280	43.008	0.
RERTH=	1.010					

STANDARD OPTION

STAGE= 2

AXIAL STATIONS

	STA. 0	STA. 1	STA.1A	STA. 2	STA. 2A	
GAMG=	1.400	1.400	1.400	1.400	1.400	0.
DR=	18.265	17.814	17.673	17.110	17.110	0.
DT=	28.845	29.296	29.437	30.000	30.000	0.
RWG=	1.000	1.000	1.000	1.000	1.000	0.

STATOR RADIAL DISTRIBUTIONS

	ROOT	PITCH	TIP	(FOR THREE	SECTORS)	
SDIA=	25.000	22.400	20.200	18.300	16.600	0.
SDEA=	0.	0.	0.	0.	0.	0.
SREC=	1.000	1.000	1.000	1.000	1.000	0.
SETA=	0.970	0.980	0.980	0.980	0.970	0.
SCF=	0.925	0.925	0.925	0.925	0.925	0.
SPA=	30.420	36.855	43.485	50.765	58.240	0.
SESTH=	1.010					

ROTOR RADIAL DISTRIBUTIONS

RDIA=	36.600	26.900	16.100	4.600	-6.700	0.
RDEA=	0.	0.	0.	0.	0.	0.
RREC=	1.000	1.000	1.000	1.000	1.000	0.
RETA=	0.919	0.946	0.946	0.946	0.919	0.
RCF=	0.900	0.900	0.900	0.900	0.900	0.
RTF=	1.000	1.000	1.000	1.000	1.000	0.
RPA=	43.350	48.150	52.350	55.750	58.550	0.
RERTH=	1.010					

NASA TURBINE COMPUTER PROGRAM

NAME- NASA TWO STAGE TURBINE

TITLE- 1.00 5041 -8 DEG. LOSS PROFILE .98 .946, .977 .90,

CASE 9. 0

STAGE PERFORMANCE

	STAGE 1	STAGE 2	STAGE 3	STAGE 4
TT 0	700.0	608.5		
PT 0	17.140	10.120		
WG 0	43.618	43.618		
DEL H	21.960	11.360		
WRT/P	67.330	106.318		
DH/T	0.03137	0.01867		
N/RT	190.532	204.358		
ETA TT	0.93545	0.93099		
ETA TS	0.82313	0.74145		
ETA AT	0.92064	0.92442		
PTO/PS1	1.600	1.328		
PTO/PT2	1.694	1.357		
PTO/PS2	1.840	1.474		
PTR1A/PS2	1.340	1.216		
TT2/TTO	0.86926	0.92220		
TTR1/TTO	0.91710	0.94789		
PS 1	10.712	7.627		
TTR 1	642.0	576.8		
PTR 1	12.478	8.347		
PS 2	9.314	6.865		
TT 2	608.5	561.1		
PT 2	10.120	7.457		
UP/VI	0.44821	0.59140		
UR/VI	0.35559	0.43665		
PSI P	1.02409	0.52977		
PSI R	1.62705	0.97180		
RX P	0.21419	0.26033		
RX R	-0.08794	-0.07414		
ALPHA 0	0.	20.327		
I STATOR	0.	0.127		
BETA 1A	46.326	15.372		
I ROTOR	8.226	-0.728		
ALPHA 2	20.327	-9.327		
DBETA R	116.221	86.347		
M 1	0.83798	0.64248		
M1 RT	1.01118	0.78437		
MR 1A	0.47073	0.35231		
MR1A RT	0.69178	0.50428		
MR 2	0.64049	0.52057		
MR2 TIP	0.69786	0.61740		
E/TH CR	16.272	9.684		
N/RTH CR	4339.3	4654.2		
WRTHCRE/D	43.4	68.6		

OVRALL PERFORMANCE

PSI P	0.77693	PSI R	1.32293	DEL H	33.31947
WRT/P	67.32999	N/RT	190.53189	DELH/T	0.04760
PTO/PT2	2.29861	PTO/PS2	2.49689	PT/PAT2	2.30771
ETA TT	0.93726	ETA TS	0.86238	ETA TAT	0.93500
WNE/60D	3142.12964	N/RTH CR	4339.32922	E/TH CR	24.68936

NASA TURBINE COMPUTER PROGRAM

NAME- NASA TWO STAGE TURBINE

TITLE- 1.00 5041 -8 DEG. LOSS PROFILE .98 .946, .977 .90,

CASE 9. 0

INTER-STAGE PERFORMANCE

STA 0	STATOR INLET		STAGE 1.		
DIAM 0	19.999	21.777	23.555	25.333	27.111
TT 0	700.0	700.0	700.0	700.0	700.0
PT 0	17.140	17.140	17.140	17.140	17.140
ALPHA 0	0.	0.	0.	0.	0.
I STATOR	0.	0.	0.	0.	0.
V 0	299.462	299.462	299.462	299.462	299.462
VU 0	0.	0.	0.	0.	0.
VZ 0	299.462	299.462	299.462	299.462	299.462
TS 0	692.5	692.5	692.5	692.5	692.5
PS 0	16.509	16.509	16.509	16.509	16.509
DENS 0	0.06434	0.06434	0.06434	0.06434	0.06434
M 0	0.232	0.232	0.232	0.232	0.232

STA 1	STATOR EXIT				
DIAM 1	19.999	21.777	23.555	25.333	27.111
ALPHA 1	69.539	67.940	66.303	64.911	63.359
DEL A	69.539	67.940	66.303	64.911	63.359
V 1	1147.973	1080.203	1017.726	954.147	895.216
VU 1	1075.550	1001.125	931.914	864.122	800.175
VZ 1	401.292	405.692	409.026	404.586	401.412
TS 1	590.3	602.9	613.8	624.2	633.3
PS 1	9.252	10.046	10.712	11.379	11.936
DENS 1	0.04230	0.04498	0.04711	0.04920	0.05087
M 1	0.96384	0.89743	0.83798	0.77904	0.72567
ZWI INC	-0.655	-0.696	-0.736	-0.768	-0.802
CP S	0.932	0.923	0.913	0.901	0.888

NASA TURBINE COMPUTER PROGRAM

NAME- NASA TWO STAGE TURBINE

TITLE- 1.00 5041 -8 DEG. LOSS PROFILE .98 .946, .977 .90,

CASE 9. 0

INTER-STAGE PERFORMANCE

STA 1A	ROTOR INLET		STAGE 1.		
DIAM 1A	19.886	21.721	23.555	25.389	27.224
PTR 1A	11.994	12.251	12.478	12.778	13.083
TTR 1A	637.2	639.2	642.0	645.8	650.4
BETA 1A	58.689	53.184	46.326	37.951	27.199
I ROTOR	8.090	8.285	8.226	7.752	6.300
R 1A	754.063	656.998	572.134	493.912	433.305
RU 1A	644.244	525.969	413.810	303.750	198.058
MR 1A	0.63336	0.54567	0.47073	0.40286	0.35083
U 1A	437.407	477.755	518.104	558.452	598.801
STA 2	ROTOR EXIT				
DIAM 2	19.436	21.495	23.555	25.615	27.674
BETA 2	57.531	58.629	59.379	60.258	60.964
DBETA	116.221	111.813	105.705	98.210	88.163
R 2	700.037	738.945	764.768	797.558	818.483
RU 2	590.611	630.919	658.125	692.497	715.609
MR 2	0.58510	0.61884	0.64049	0.66793	0.68411
U 2	427.500	472.802	518.104	563.406	608.708
RX	-0.00404	0.11698	0.21419	0.30756	0.38295
DELH	21.682	22.139	22.182	22.137	21.658
PSI P	2.90232	2.45375	2.06896	1.76143	1.48743
ETA TT	0.91417	0.94185	0.94519	0.94753	0.92554
ETA TS	0.80596	0.82681	0.83128	0.83237	0.81640
ETA AT	0.89418	0.92234	0.92982	0.93439	0.91678
ZWI INC	-1.853	-1.613	-1.420	-1.245	-1.091
CPR	-0.160	0.209	0.440	0.616	0.720
PT 2A	10.061	10.112	10.122	10.148	10.139
TT 2A	609.6	607.7	607.6	607.7	609.7
V 2A	402.129	406.587	403.081	404.008	398.167
VU 2A	164.063	158.533	140.021	128.808	106.467
ALPHA 2A	24.078	22.949	20.327	18.592	15.509
MF 2A	0.30673	0.31338	0.31636	0.32046	0.32044
VZ 2A	367.139	374.407	377.979	382.925	383.669
TS 2A	596.2	594.0	594.0	594.2	596.5
PS 2A	9.304	9.334	9.355	9.376	9.391
DENS 2A	0.04213	0.04241	0.04251	0.04259	0.04249
M 2A	0.33596	0.34032	0.33737	0.33811	0.33255

NASA TURBINE COMPUTER PROGRAM

NAME- NASA TWO STAGE TURBINE

TITLE- 1.00 5041 -8 DEG. LOSS PROFILE .98 .946, .977 .90,

CASE 9. 0

INTER-STAGE PERFORMANCE

STA 0	STATOR INLET		STAGE 2.		
DIAM 0	19.323	21.439	23.555	25.671	27.787
TT 0	608.5	608.5	608.5	608.5	608.5
PT 0	10.120	10.120	10.120	10.120	10.120
ALPHA 0	24.078	22.949	20.327	18.592	15.509
I STATOR	-0.921	0.549	0.127	0.292	-1.090
V 0	402.129	406.587	403.081	404.008	398.167
VU 0	164.063	158.533	140.021	128.808	106.467
VZ 0	367.139	374.407	377.979	382.925	383.669
TS 0	596.2	594.0	594.0	594.2	596.5
PS 0	9.304	9.334	9.355	9.376	9.391
DENS 0	0.04213	0.04241	0.04251	0.04259	0.04249
M 0	0.336	0.340	0.337	0.338	0.333

STA 1	STATOR EXIT				
DIAM 1	18.962	21.259	23.555	25.851	28.148
ALPHA 1	61.651	59.301	57.068	54.670	52.234
DEL A	85.729	82.250	77.395	73.262	67.743
V 1	852.183	797.562	746.698	695.931	647.721
VU 1	749.978	685.793	626.716	567.767	512.036
VZ 1	404.658	407.178	405.938	402.443	396.689
TS 1	548.0	555.5	562.1	568.2	573.6
PS 1	6.934	7.324	7.627	7.929	8.162
DENS 1	0.03415	0.03558	0.03662	0.03767	0.03841
M 1	0.74258	0.69027	0.64248	0.59558	0.55171
ZWI INC	-1.037	-1.099	-1.132	-1.169	-1.176
CP S	0.777	0.740	0.709	0.663	0.622

NASA TURBINE COMPUTER PROGRAM
NAME- NASA TWO STAGE TURBINE
TITLE- 1.00 5041 -8 DEG. LOSS PROFILE .98 .946, .977 .90,
CASE 9. 0
INTER-STAGE PERFORMANCE

STA 1A	ROTOR INLET		STAGE 2.		
DIAM 1A	18.849	21.202	23.555	25.908	28.261
PTR 1A	7.983	8.170	8.347	8.606	8.869
TTR 1A	570.7	573.2	576.8	581.8	587.9
BETA 1A	40.543	29.084	15.372	-0.488	-16.202
I ROTOR	3.944	2.184	-0.728	-5.088	-9.502
R 1A	522.850	455.198	409.722	390.560	400.023
RU 1A	339.865	221.264	108.612	-3.324	-111.614
MR 1A	0.45563	0.39382	0.35231	0.33398	0.34043
U 1A	414.602	466.353	518.104	569.855	621.606

STA 2	ROTOR EXIT				
DIAM 2	18.399	20.977	23.555	26.133	28.711
BETA 2	45.803	47.980	49.600	51.528	52.888
DBETA	86.347	77.064	64.972	51.040	36.686
R 2	509.758	559.144	596.911	646.636	683.989
RU 2	365.472	415.395	454.567	506.260	545.449
MR 2	0.44404	0.48778	0.52057	0.56377	0.59497
U 2	404.695	461.399	518.104	574.808	631.513
RX	0.02621	0.16021	0.26033	0.35784	0.43101
DELH	11.860	11.960	11.654	11.321	10.491
PSI P	1.76916	1.39156	1.08700	0.86528	0.66904
ETA TT	0.92504	0.95555	0.94969	0.94336	0.89093
ETA TS	0.77236	0.77943	0.76036	0.73946	0.68609
ETA AT	0.92285	0.95234	0.94352	0.93608	0.87998
ZWI INC	-1.831	-1.493	-1.218	-0.968	-0.751
CPR	-0.052	0.337	0.529	0.635	0.658
PT 2A	7.336	7.395	7.443	7.496	7.540
TT 2A	559.1	558.6	559.9	561.3	564.8
V 2A	357.521	377.102	392.056	408.091	421.585
VU 2A	-39.223	-46.005	-63.537	-68.548	-86.064
ALPHA 2A	-6.298	-7.007	-9.327	-9.670	-11.779
MF 2A	0.30955	0.32651	0.33740	0.35074	0.35899
VZ 2A	355.363	374.286	386.874	402.292	412.707
TS 2A	548.4	546.8	547.1	547.4	550.0
PS 2A	6.859	6.861	6.865	6.868	6.871
DENS 2A	0.03376	0.03387	0.03387	0.03386	0.03372
M 2A	0.31143	0.32897	0.34192	0.35580	0.36672

APPENDIX 6

```
NAME-  NASA TWO STAGE TURBINE
 TITLE-  1.00 5041  8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN  STGCH=1.0, PTPS=1.1, STAGE=1.0,
PTIN= 17.14,
TTIN= 700.,
WAIR= 0.,
FAIR= 0.,
AACS= 1.,
STG= 2.,
SECT= 5.,
DELC=.1,
DELL=.1,
DELA=.1,
EXPN= 3.0,
EXPP= 3.0,
RG= 53.35,
RPM= 5041.,
PCNH= .2,.2,.2,.2,.2,
GAMG=  1.4,1.4,1.4,1.4,1.4,
DR=  19.110,19.110,18.969,18.406,18.265,
DT=  28.0,28.0,28.141,28.704,28.845,
RWG=  1.,1.,1.,1.,1.,
SDIA=  0.,0.,0.,0.,0.,
SREC=  1.,1.,1.,1.,1.,
SETA=  .97,.98,.98,.98,.97,
SCF=  .977,.977,.977,.977,.977,
SDEA=  0.,
SPA=  22.140,26.035,30.135,34.194,38.499,
SESTH= 1.,
RDIA=  50.6,44.9,38.1,30.2,20.9,
RREC=  1.,1.,1.,1.,1.,
RETA= .919,.946,.946,.946,.919,
RCF= .95,.95,.95,.95,.95,
RTF= 1.,1.,1.,1.,1.,
RDEA=  0.,
RPA=  33.408,36.352,38.976,41.280,43.008,
RERTH=  1.01,
ENDSTG= 0.$
$DATAIN  STAGE=2.0,
GAMG=  1.4,1.4,1.4,1.4,1.4,
DR=  18.265,17.814,17.673,17.110,17.110,
DT=  28.845,29.296,29.437,30.000,30.000,
RWG=  1.,1.,1.,1.,1.,
SDIA=  25.0,22.4,20.2,18.3,16.6,
SREC=  1.,1.,1.,1.,1.,
SETA=  .97,.98,.98,.98,.97,
SCF= .925,.925,.925,.925,.925,
SPA=  30.420,36.855,43.485,50.765,58.240,
SESTH=  1.01,
RDIA=  36.6,26.9,16.1,4.6,-6.7,
RREC=  1.,1.,1.,1.,1.,
RETA= .919,.946,.946,.946,.919,
RCF= .90,.90,.90,.90,.90,
RTF= 1.,1.,1.,1.,1.,
RPA=  43.350,48.150,52.350,55.750,58.550,
RERTH=  1.01,
ENDSTG= 1.$
```

```
NAME-   NASA TWO STAGE TURBINE
 TITLE-   0.60 3025   8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN   STGCH=0.0, PTPS=1.1, STAGE=1.0,
RPM= 3025,
ENDSTG= 0.$
$DATAIN   STAGE=2.0,
ENDSTG=1.$
NAME-   NASA TWO STAGE TURBINE
 TITLE-   0.70 3529   8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN   STGCH=0.0, PTPS=1.1, STAGE=1.0,
RPM= 3529,
ENDSTG= 0.$
$DATAIN   STAGE=2.0,
ENDSTG=1.$
NAME-   NASA TWO STAGE TURBINE
 TITLE-   0.80 4033   8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN   STGCH=0.0, PTPS=1.1, STAGE=1.0,
                   RPM=4033.,
ENDSTG= 0.$
$DATAIN   STAGE=2.0,
ENDSTG=1.$
NAME-   NASA TWO STAGE TURBINE
 TITLE-   0.90 4537   8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN   STGCH=0.0, PTPS=1.1, STAGE=1.0,
                   RPM=4537.,
ENDSTG= 0.$
$DATAIN   STAGE=2.0,
ENDSTG=1.$
NAME-   NASA TWO STAGE TURBINE
 TITLE-   1.10 5545   8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN   STGCH=0.0, PTPS=1.1, STAGE=1.0,
                   RPM=5545.,
ENDSTG= 0.$
$DATAIN   STAGE=2.0,
ENDSTG=1.$
NAME-   NASA TWO STAGE TURBINE
 TITLE-   1.20 6049   8 DEG. LOSS PROFILE .98 .946, .977 .90, 3.0 3.0,
$DATAIN   STGCH=0.0, PTPS=1.1, STAGE=1.0,
                   RPM=6049.,
ENDSTG= 0.$
$DATAIN   STAGE=2.0,
ENDSTG=1.$
```

NASA-Langley, 1967 E-3734

04435
274 cup
6-3-67

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

—National Aeronautics and Space Act of 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

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

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

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

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

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

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

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

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

SCIENTIFIC AND TECHNICAL INFORMATION DIVISION

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546