

AD-A124 987

FINITE ELEMENT PROGRAM FOR CALCULATING FLOWS IN
TURBOMACHINES WITH RESULTS FOR NASA TASK-1 COMPRESSOR
(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA J A FERGUSON

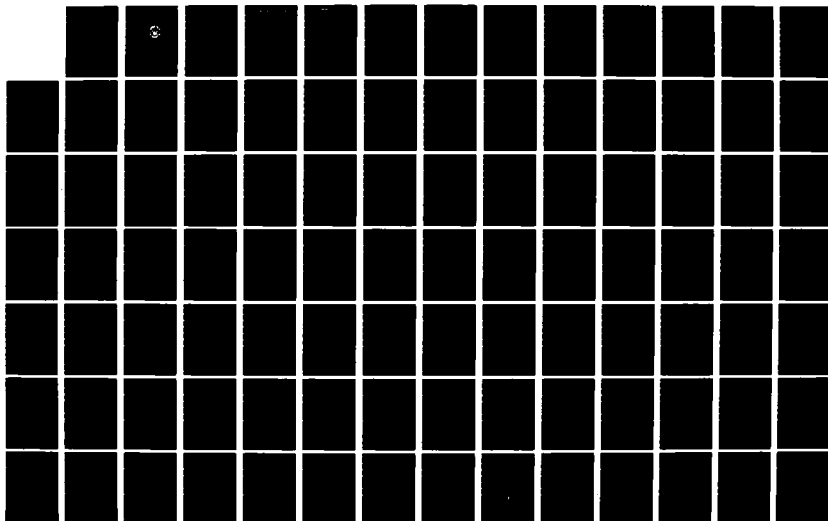
1/3

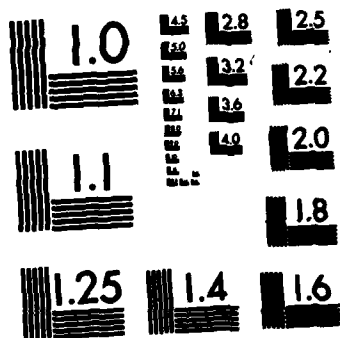
UNCLASSIFIED

OCT 82

F/G 28/4

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A124587

NAVAL POSTGRADUATE SCHOOL

Monterey, California



Copy available to DTIC does not
permit fully legible reproduction

THESIS

FINITE ELEMENT PROGRAM FOR CALCULATING
FLOWS IN TURBOMACHINES WITH RESULTS
FOR NASA TASK-1 COMPRESSOR

BY

Julian A. Ferguson III

October 1982

Thesis Advisor:

Raymond P. Shreeve

Approved for public release; distribution unlimited

DTIC FILE COPY

88 028

092

DTIC
ELECTED
01 1983
E

DISCLAIMER NOTICE

**THIS DOCUMENT IS BEST QUALITY
PRACTICABLE. THE COPY FURNISHED
TO DTIC CONTAINED A SIGNIFICANT
NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.**

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A124987	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Finite Element Program for Calculating Flows in Turbomachines with Results for NASA Task-1 Compressor		5. TYPE OF REPORT & PERIOD COVERED Engineer's Thesis October 1982
7. AUTHOR(s) Julian A. Ferguson III		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
12. REPORT DATE October 1982		13. NUMBER OF PAGES 247
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Finite Element Method Flow in Turbomachines Compressor Flow Field		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A general mesh generation code (MESHGEN) and finite element flow solver (TURBO) for calculating the flow development through axial turbomachines are fully documented. The finite element approach followed Hirsch and Warzee. Excellent results were obtained for the NASA Task-1 compressor operating with subsonic flow conditions. Construction of the code will allow		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-0001

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE/When Data Entered

straightforward extension to transonic flows, turbine stages
and multiple stage machines.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	33 CP



Approved for public release; distribution unlimited

Finite Element Program for Calculating Flows in
Turbomachines with Results for NASA Task-1 Compressor

by

Julian A. Ferguson III
Lieutenant, United States Navy
B.S., Auburn University, 1975

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

and

AERONAUTICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
October 1982

Author:

Julian A. Ferguson III

Approved by:

Max F. Platzer Thesis Advisor

Max F. Platzer Second Reader

Max F. Platzer
Chairman, Department of Aeronautics

William M. Lallen
Dean of Science and Engineering

ABSTRACT

A general mesh generation code (MESHGEN) and finite element flow solver (TURBO) for calculating the flow development through axial turbomachines are fully documented. The finite element approach followed Hirsch and Warzee. Excellent results were obtained for the NASA Task-1 compressor operating with subsonic flow conditions. Construction of the code will allow straightforward extension to transonic flows, turbine stages and multiple stage machines.

TABLE OF CONTENTS

I.	INTRODUCTION	15
	A. STATEMENT OF TASK	15
	B. DESCRIPTION OF THE PROBLEM	16
II.	MATHEMATICAL MODEL	19
III.	FINITE ELEMENT METHOD	24
	A. INTRODUCTION	24
	B. APPLICATION OF THE WEIGHTED RESIDUAL PROCESS	25
	C. APPLICATION OF THE FINITE ELEMENT METHOD . .	27
	D. NUMERICAL INTEGRATION TECHNIQUE	30
	1. Stiffness Matrix Evaluation	31
	2. Right-hand Side Vector Evaluation . . .	35
IV.	DESCRIPTION OF THE PROGRAM MESHGEN	36
	A. MAIN PROGRAM DESCRIPTION	36
	B. SUBROUTINE DESCRIPTIONS	38
	1. Subroutine INIT1	38
	2. Subroutine INPUT	39
	3. Subroutine TMESH	40
	4. Subroutine CONEC	41
	5. Subroutine INIT2	41
	6. Subroutine FLOFCT	42
	7. Subroutine TASK1	43
	8. Subroutine FILGEN	44
	9. Subroutine MPLOT	44

10.	Subroutine ERR1	45
11.	Subroutine ERR2	45
12.	Subroutine ERR3	45
V.	DESCRIPTION OF THE PROGRAM TURBO	47
A.	MAIN PROGRAM DESCRIPTION	47
B.	SUBROUTINE DESCRIPTIONS	49
1.	Subroutine RDATA	49
2.	Subroutine INIT1	50
3.	Subroutine ZERO1	50
4.	Subroutine INPUT	50
5.	Subroutine DIST	51
6.	Subroutine SLINE	53
7.	Subroutine DUCT	57
8.	Subroutine ROTO	58
9.	Subroutine STAT	60
10.	Subroutine FCAL	61
11.	Subroutine STIFF	64
12.	Subroutine DSIMQ	66
13.	Subroutine REPLA	66
14.	Subroutine TEST	67
15.	Subroutine RELAX	67
16.	Subroutine NOCON	67
17.	Subroutine OUTPUT	68
18.	Subroutine MPLOT	68
19.	Subroutine SHAPE	69
20.	Subroutine JACOB	69

21. Subroutine ERR1	70
22. Subroutine ERR2	70
23. Subroutine ERR3	71
VI. RESULTS AND DISCUSSION	72
A. PROGRAM VERIFICATION	72
1. Test Case 1	72
2. Test Case 2	74
3. Test Case 3	74
4. Test Case 4	75
B. POINTS OF INTEREST	76
VII. CONCLUSIONS AND RECOMMENDATIONS	78
A. PROGRAM MESHGEN	78
B. PROGRAM TURBO	78
LIST OF REFERENCES	80
TABLES	83
FIGURES	87
APPENDIX A. STORAGE ALLOCATION FOR THE PROGRAM MESHGEN	163
APPENDIX B. LISTING OF THE OUTPUT VARIABLES FOR PROGRAM MESHGEN	165
APPENDIX C. STORAGE ALLOCATION FOR THE PROGRAM TURBO	167
APPENDIX D. LISTING OF THE OUTPUT VARIABLES FOR PROGRAM TURBO	170
APPENDIX E. LISTING OF THE PROGRAM MESHGEN	171
APPENDIX F. LISTING OF THE PROGRAM TURBO	183
APPENDIX G. SAMPLE PROGRAM OUTPUT	216
INITIAL DISTRIBUTION LIST	247

LIST OF TABLES

1.	Connectivity Relationships for Figure 3	83
2.	Comparison of Program Predictions with NASA Task-1 Compressor Measurements at 50% Design Speed . . .	84
3.	Comparison of Program Predictions with NASA Task-1 Compressor Measurements at 70% Design Speed . . .	85
4.	Comparison of Program Predictions with NASA Task-1 Compressor Measurements at 80% Design Speed . . .	86

LIST OF FIGURES

1.	S1 and S2 Stream Surfaces	87
2.	Nomenclature of an Eight-Node Element	88
3.	Example of a Three Element Mesh	89
4.	Mapping Relationships from the ξ, η to the z, r Plane	90
5.	Mesh Generation	91
6.	MESHGEN Generated Plot of the 222 Node Mesh Used in the Calculations of the Meridional Through- Flow	92
7.	A TURBO Generated Tektonix 618 Plot of the Rotor Inlet Axial Velocity at 50% Design Speed	93
8.	A TURBO Generated Tektonix 618 Plot of the Rotor Inlet Relative Flow Angles	94
9.	A TURBO Generated Tektonix 618 Plot of the Total Pressure Ratio of the Rotor vs Inlet Radius	95
10.	A TURBO Generated Tektonix 618 Plot of the Adiabatic Efficiency of the Rotor vs Inlet Radius	96
11.	A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Axial Velocity	97
12.	A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Relative Flow Angles	98
13.	A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Absolute Flow Angles	99
14.	A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Deviation Flow Angles	100
15.	A TURBO Generated Tektonix 618 Plot of the Stator Inlet Axial Velocity	101
16.	A TURBO Generated Tektonix 618 Plot of the Stator Inlet Absolute Flow Angles	102

17.	A TURBO Generated Tektonix 618 Plot of the Stator Total Pressure Ratio vs Inlet Radius	103
18.	A TURBO Generated Tektonix 618 Plot of the Stator Outlet Axial Velocity	104
19.	A TURBO Generated Tektonix 618 Plot of the Stator Outlet Absolute Flow Angles	105
20.	A TURBO Generated Tektonix 618 Plot of the Stator Outlet Deviation Flow Angles	106
21.	Comparison of Predictions to Gavito and Observations for Rotor Inlet Axial Velocity, 50% Design	107
22.	Comparison of Predictions to Gavito and Observations for Rotor Outlet Axial Velocity, 50% Design	108
23.	Comparison of Predictions to Gavito and Observations for Stator Inlet Axial Velocity, 50% Design	109
24.	Comparison of Predictions to Gavito and Observations for Stator Outlet Axial Velocity, 50% Design	110
25.	Comparison of Predictions to Hirsch and Observations for Rotor Inlet Axial Velocity, 50% Design	111
26.	Comparison of Predictions to Hirsch and Observations for Rotor Outlet Axial Velocity, 50% Design	112
27.	Comparison of Predictions to Hirsch and Observations for Rotor Outlet Relative Flow Angles, 50% Design	113
28.	Comparison of Predictions to Hirsch and Observations for Rotor Outlet Absolute Flow Angles, 50% Design	114
29.	Comparison of Predictions to Hirsch and Observations for Stator Inlet Axial Velocity, 50% Design	115
30.	Comparison of Predictions to Hirsch and Observations for Stator Outlet Axial Velocity, 50% Design	116
31.	Axial Velocity at the Rotor Inlet, 50% Design . .	117
32.	Relative Flow Angles at Rotor Inlet, 50% Design .	118
33.	Total Pressure Ratio of the Rotor, 50% Design . .	119

34.	Adiabatic Efficiency of the Rotor, 50% Design . .	120
35.	Axial Velocity at the Rotor Outlet, 50% Design .	121
36.	Relative Flow Angles at Rotor Outlet, 50% Design	122
37.	Absolute Flow Angles at Rotor Outlet, 50% Design	123
38.	Deviation Flow Angles at Rotor Outlet, 50% Design	124
39.	Axial Velocity at the Stator Inlet, 50% Design .	125
40.	Absolute Flow Angles at Stator Inlet, 50% Design	126
41.	Total Pressure Ratio at Stator Inlet, 50% Design	127
42.	Axial Velocity at the Stator Outlet, 50% Design .	128
43.	Absolute Flow Angles at Stator Outlet, 50% Design	129
44.	Deviation Flow Angles at Stator Outlet, 50% Design	130
45.	Axial Velocity at the Rotor Inlet, 70% Design . .	131
46.	Relative Flow Angles at Rotor Inlet, 70% Design .	132
47.	Total Pressure Ratio of the Rotor, 70% Design . .	133
48.	Adiabatic Efficiency of the Rotor, 70% Design . .	134
49.	Axial Velocity at the Rotor Outlet, 70% Design .	135
50.	Relative Flow Angles at Rotor Outlet, 70% Design	136
51.	Absolute Flow Angles at Rotor Outlet, 70% Design	137
52.	Deviation Flow Angles at Rotor Outlet, 70% Design	138
53.	Axial Velocity at the Stator Inlet, 70% Design .	139
54.	Absolute Flow Angles at Stator Inlet, 70% Design	140
55.	Total Pressure Ratio at Stator Inlet, 70% Design	141
56.	Axial Velocity at the Stator Outlet, 70% Design .	142
57.	Absolute Flow Angles at Stator Outlet, 70% Design	143
58.	Deviation Flow Angles at Stator Outlet, 70% Design	144

59.	Axial Velocity at the Rotor Inlet, 80% Design . .	145
60.	Relative Flow Angles at Rotor Inlet, 80% Design .	146
61.	Total Pressure Ratio of the Rotor, 80% Design . .	147
62.	Adiabatic Efficiency of the Rotor, 80% Design . .	148
63.	Axial Velocity at the Rotor Outlet, 80% Design .	149
64.	Relative Flow Angles at Rotor Outlet, 80% Design	150
65.	Absolute Flow Angles at Rotor Outlet, 80% Design	151
66.	Deviation Flow Angles at Rotor Outlet, 80% Design	152
67.	Axial Velocity at the Stator Inlet, 80% Design .	153
68.	Absolute Flow Angles at Stator Inlet, 80% Design	154
69.	Total Pressure Ratio at Stator Inlet, 80% Design	155
70.	Axial Velocity at the Stator Outlet, 80% Design .	156
71.	Absolute Flow Angles at Stator Outlet, 80% Design	157
72.	Deviation Flow Angles at Stator Outlet, 80% Design	158
73.	Axial Velocity at the Rotor Inlet, 80% Design, 9% Blockage	159
74.	Axial Velocity at the Rotor Outlet, 80% Design, 9% Blockage	160
75.	Axial Velocity at the Stator Inlet, 80% Design, 9% Blockage	161
76.	Axial Velocity at the Stator Outlet, 80% Design, 9% Blockage	162

LIST OF SYMBOLS

<u>Symbol</u>	<u>Description</u>	<u>Units</u>
ψ	Stream function	lbm/sec
α	Absolute flow angle whose tangent is the ratio of the absolute tangential-to-meridional velocity	degrees
β	Relative flow angle whose tangent is the ratio of the absolute tangential-to-meridional velocity	degrees
δ	Deviation angle, difference in flow angle and camber-line angle at trailing edge in cascade projection tangential-to-meridional velocity	degrees
κ_1	Angle between tangent to the blade meanline and the axial direction	degrees
ϕ	Camber angle, difference between angles in cascade projection of tangents to camber line at extremes of camber-line arc	degrees
σ	Solidity, ratio of chord to blade spacing	dimensionless
$\tilde{\omega}$	Total-pressure-loss coefficient	dimensionless
T	Temperature	$^{\circ}\text{R}$
P	Pressure	psia
ρ	Density	lbm/ft ³

Subscripts

t	Total conditions
1	blade leading edge
2	blade trailing edge
E	Over an element

Superscripts

e	For an element
---	----------------

I. INTRODUCTION

A. STATEMENT OF TASK

The original task of this research project was to continue the work of Macchi [Ref. 1] and Cirone [Ref. 2] in the development of a turbine prediction computer code for the Turbopropulsion Laboratory at the Naval Postgraduate School. An analysis of the referenced code by Ferguson [Ref. 3] indicated that a significant amount of work remained to be done in order to make the program operational. In the author's opinion the task could be better accomplished through the use of a different solution technique. After additional study and review of similar work [Refs. 4, 5, 6 and 7] it was decided that a finite element approach to the problem would be adopted. A program developed by Gavito [Ref. 8], which followed the work of Hirsch and Warzee [Ref. 4], was selected as the basis for development of the computer code described in the sections that follow. However, Gavito's program was formulated as a compressor performance prediction which, as it was reported, had not given results similar to those obtained by Hirsch and Warzee. Thus the first goal of the project became the development of an axial compressor prediction code that could produce results comparable to those published by Hirsch and Warzee. The second goal was to revise and document the program so that its

application to any compressor and its extension to turbine analysis could be carried out without excessive difficulty.

B. DESCRIPTION OF THE PROBLEM

One purpose of conducting a through-flow analysis is to predict the performance of a turbomachine under design and off-design operating conditions. Through the combination of a mathematical model and empirically determined correlations, it is possible to provide the engineer with a tool that will determine the effects of variations in design parameters and analyze the performance of an existing machine.

The first problem addressed in the formulation of a performance prediction code is that of expressing the analysis in a form that can be accurately and efficiently solved by computer methods. Most methods for through-flow calculations are based on the classic work of Wu [Ref. 9] which stated that the equations of fluid flow in turbomachines can be solved on the intersecting sets of stream surfaces known as the S1 family and S2 family of stream surfaces (Fig. 1). In general the intersection of a S1 surface and a S2 surface is a twisting line with three dimensional variations. However, if an axisymmetric assumption is made, the S2 surface will lie on a meridional plane and the directional derivatives on the S2 surface become the $\partial(\)/\partial r$ and $\partial(\)/\partial z$ in cylindrical coordinates. As shown by Smith [Ref. 10], circumferentially averaged equations with an axisymmetric

flow assumption can be used to a good first approximation for the through-flow analysis.

Three general methods of solving the so-called radial equilibrium equation of flows in turbomachines which results from the axisymmetric assumption can be found in the literature. The first is called the streamline curvature method [Refs. 1, 2 and 11]. The method derived its name from the fact that the radius of curvature of the streamline is an integral part of the formulation. The second, a matrix method, applies a finite differences technique to the radial equilibrium equation, normally after it has been reduced to a Poisson form [Refs. 12 and 13]. The third method is the finite element method which was used in the present work.

In the mid-1970's, Hirsch and Warzee [Ref. 4] first reported the application of the finite element method to solution of the axisymmetric radial equilibrium equation. They applied the finite element technique to solve the equation expressed in quasi-harmonic form in terms of the stress function. They published extensive comparisons of the predictions obtained using their method with measurements obtained on several machines under various operating conditions. In general, the method produced excellent results for compressors and turbines of single and multi-stage configurations. It was this demonstrated ability of the method over

such a wide range of parameters that let to its selection for use in the present project.

In the sections which follow, the development of programs MESHGEN and TURBO, which are based on the work of Hirsch and Warzee [Ref. 4], is documented. Comparisons are given of the results obtained when the program was applied to analyze the NASA Task-1 compressor with results obtained by the referenced authors.

II. MATHEMATICAL MODEL

The equation of motion for a fluid has the general form (Vavra [Ref. 14])

$$(\partial \vec{V} / \partial t) + (\vec{V} \cdot \nabla) \vec{V} = -\nabla p / \rho + \vec{F}_f - \nabla(gz) \quad (1)$$

Using the vector identity

$$(\vec{V} \cdot \nabla) \vec{V} = \nabla(V^2/2) - (\vec{V} \times \nabla \times \vec{V}) \quad (2)$$

Eq. (1) can be written as

$$\partial \vec{V} / \partial t + \nabla(V^2/2 + gz) = -\nabla p / \rho + \vec{F}_f + (\vec{V} \times \nabla \times \vec{V}) \quad (3)$$

The first law of thermodynamics for a fluid particle can be written as

$$Tds = dh - dp/\rho \quad (4)$$

which, along an elemental path length $d\vec{r}$ in a fluid field implies that

$$T(d\vec{r} \cdot \nabla s) = (d\vec{r} \cdot \nabla)h - (d\vec{r} \cdot \nabla)p/\rho \quad (4a)$$

or

$$d\vec{r} \cdot (\nabla h - T\nabla s - \nabla p/\rho) = 0 \quad (5)$$

Since $d\vec{r}$ is not equal to zero in general, in a homogeneous fluid flow the first law may be expressed as

$$\nabla h - T\nabla s - \nabla p/\rho = 0 \quad (6)$$

Substituting Eq. (6) into Eq. (3) yields

$$\partial \vec{V} / \partial t + \nabla (h + V^2/2 + gz) = T \nabla s + \vec{F}_f + \vec{V} \times \nabla \times \vec{V} \quad (7)$$

For steady, inviscid flow Eq. (7) reduces to

$$\nabla H = T \nabla s + (\vec{V} \times \nabla \times \vec{V}) \quad (8)$$

As shown by Hirsch and Warzee [Ref. 15], Eq. (8) can be revised to describe the flow through blade rows by introducing a circumferential averaging process and assuming that the flow is axisymmetric at the averaged condition. The averaged equation can be expressed as

$$-(\vec{V} \times \nabla \times \vec{V}) = T \nabla s - \nabla H + \vec{F}_b + \vec{F}_d \quad (9)$$

where F_b is the body force representing the action of the blades on the flow and F_d represents the dissipative force whose work generates the irreversible entropies. The F_d forces are considered to be uniformly distributed in the tangential direction and proportional to the loss coefficients. Equation (9) leads to the following three equations in cylindrical coordinates (with $\partial(\)/\partial\theta = 0$)

$$\begin{aligned} (V_u/r) (\partial(rV_u)/\partial r) - V_z ((\partial V_r/\partial z) - (\partial V_z/\partial r)) = \\ \partial H/\partial r - T(\partial s/\partial r) - F_{b,r} - F_{d,r} \end{aligned} \quad (10a)$$

$$(V_z/r) (\partial(rV_u)/\partial z) + (V_r/r) (\partial(rV_u)/\partial r) = F_u \quad (10b)$$

$$\begin{aligned} V_r ((\partial V_r/\partial z) - (\partial V_z/\partial r)) - (V_u/r) (\partial(rV_u)/\partial z) = \\ \partial H/\partial z - T(\partial s/\partial z) - F_z \end{aligned} \quad (10c)$$

Equation (10a) expresses the radial equilibrium of the meridional through-flow and it is the governing equation to be solved for the velocity components. Equations (10b) and (10c) determine the tangential and axial components of the forces once Eq. (10a) has been solved.

For the solution of Eq. (10a) to have physical meaning, care must be taken to ensure that continuity is satisfied throughout the field. In general, the continuity equation can be expressed as

$$\partial \rho / \partial t + \nabla \cdot (\rho \vec{V}) = 0 \quad (11)$$

which for steady, circumferentially averaged flow can be written as

$$(\partial / \partial r) (\rho r b V_r) + (\partial / \partial z) (\rho r b V_z) = 0 \quad (12)$$

where b is a blockage factor describing the reduction in the flow area in the tangential direction due to the presence of rotor and stator blades. The tangential blockage is approximated by

$$b = 1 - t/s \quad (12a)$$

where t is the blade thickness and s is the blade spacing. As will be discussed in the description of subroutine INPUT, this factor will be modified to account for the end-wall boundary layer contractions. A stream function, ψ , can be introduced and defined so that Eq. (12) is automatically satisfied as follows:

$$\partial\psi/\partial r = \rho r b V_z \quad (13a)$$

$$\partial\psi/\partial z = -\rho r b V_r \quad (13b)$$

After the substitution of Eqs. (13a) and (13b) into Eq. (10a) one is assured of the implicit satisfaction of continuity as the radial equilibrium equation is solved explicitly. Equation (10a) may now be written as

$$\begin{aligned} (\partial/\partial r) \left((1/\rho r b) (\partial\psi/\partial r) \right) + (\partial/\partial z) \left((1/\rho r b) (\partial\psi/\partial z) \right) = \\ (1/V_z) \left((\partial H/\partial r) - T(\partial s/\partial r) + \right. \\ \left. (V_u/r) (\partial(rV_u)/\partial r) \right) - F_{b,r} - F_{d,r} \end{aligned} \quad (14)$$

The equation is written in a slightly different form for solution in rotor regions where rothalpy remains constant along a streamline. The definition of rothalpy, H_R , given by

$$H_R = H - UV_u \quad (15)$$

is substituted into Eq. (14) and the terms in brackets on the right-hand side become

$$\left[\partial H_R/\partial r - T(\partial s/\partial r) - (W_u/r) (\partial(rV_u)/\partial r) - F_{b,r} - F_{d,r} \right] \quad (16)$$

The significance of the $F_{b,r}$ and $F_{d,r}$ terms can be analyzed in the following manner. The body force F_b acts in a direction normal to the mean blade surface, which for radial blading is in the circumferential direction. The term $F_{b,r}$ accounts for the body force component in the radial direction that results when blade lean is present. For most blading this is a small term that may be neglected.

Similarly, $F_{d,r}$ is the contribution of the dissipative forces in the radial direction for non-cylindrical stream surfaces. This contribution can normally be neglected for axial-flow machines, which is the case treated here. (Note that the two body force terms are included in the analysis for machines in which the magnitudes of these forces are significant.) With these simplifications the radial equilibrium equation may be written in the form

$$\begin{aligned} (\partial/\partial r) \left((1/\rho r b) (\partial \psi / \partial r) \right) + (\partial/\partial z) \left((1/\rho r b) (\partial \psi / \partial z) \right) = \\ (1/V_z) \left((\partial H / \partial r) - T (\partial s / \partial r) \right) + (V_u / r) \left(\partial (r V_u) / \partial r \right) \end{aligned} \quad (17)$$

with the appropriate modifications for solution in a rotor region.

III. FINITE ELEMENT METHOD

A. INTRODUCTION

The finite element method is a numerical procedure that is particularly well suited to solving problems in continuum mechanics, which invariably involve equations expressed in differential form. As stated by Cook [Ref. 16], the essence of the finite element method is the "piecewise approximation of a function ϕ , by means of polynomials, each defined over a small region (element) and expressed in terms of nodal values of the function."

In order to understand the finite element method and the solution techniques employed in the computer program reported herein, one must first have a complete understanding of the basic element, the terminology used to describe the element, and the relationship between the element and the remainder of the solution domain. The complete problem is solved in a piecewise manner, in which the solution of the derived governing relationship over the discrete region of an element is sought to determine the contribution of the element to the overall solution. Figure 2 illustrates the single element as it is used in the present work and the nomenclature for the element on what is referred to as the "local domain". The number scheme to employ is arbitrary, limited only by the requirement that the system remain

consistent from element to element. Figure 3 shows a vertical stack of three elements to show how elements are connected in what is known as a "global domain". Table 1 lists the relationship between the two reference systems, known as the connectivity. The connectivity is important because the solution of a problem over a computational region involves a careful summation of the local contributions of each element to the global equations. The summation process is tracked by the connectivity. Again the global numbering scheme is arbitrary, influenced mainly by considerations of computer storage and computational efficiency.

The key concept to be grasped is that the finite element method is a series of local solutions that are coupled together through the connectivity relationships to form a solution for the entire computational domain. A more detailed description of the finite element method is contained in Refs. 16 through 18.

B. APPLICATION OF THE WEIGHTED RESIDUAL PROCESS

A standard weighted residuals process was used to transform Eq. (17) into a form that can be solved by numerical techniques. As a first step, the equation was written in a more compact form as

$$(\partial/\partial r)[k(\partial\psi/\partial r)] + (\partial/\partial z)[k(\partial\psi/\partial z)] + f = 0 \quad (18)$$

where

$$k(r,z) = (1/\rho r b) \quad (19)$$

$$f(r,z) = (1/V_z) [T(\partial s/\partial r) - \partial H/\partial r + (V_u/r)(\partial(rV_u)/\partial r)] \quad (20)$$

with the boundary conditions

$$k(\partial\psi/\partial n) + \alpha_1(\psi - \psi_0) = 0 \quad (21)$$

on the associated exterior surface S . Equations (18) and (21) may be rewritten as

$$(1/r) \{ (\partial/\partial r) [k(\partial\psi/\partial r)] + (\partial/\partial z) [k(\partial\psi/\partial z)] + f \} = 0 \quad (22)$$

in the volume, V , and

$$(1/r) [k(\partial\psi/\partial n) + \alpha_1(\psi - \psi_0)] = 0 \quad (23)$$

on the surface, S . An approximation, $\tilde{\psi}(r,z)$, of the unknown solution is searched for such that the corresponding weighted residual, $\bar{R}$, is equal to zero. Analytically this is expressed as

$$\bar{R} = \int_V W(r,z) R_V(r,z) dV + \int_S W(r,z) R_S(r,z) dS = 0 \quad (24)$$

where $W(r,z)$ is the (known) weight function and R_V and R_S are the so-called "residuals" in the volume and on the surface, respectively. As the sum of R_V and R_S approaches zero, the approximation, $\tilde{\psi}$, approaches the exact solution, ψ , with R_V and R_S defined to be identically zero if $\tilde{\psi} = \psi$. By defining R_V to be equal to the left-hand side of Eq. (22) and R_S to be equal to the left-hand side of Eq. (23), Eq. (24) can be written as

$$\begin{aligned} \int_{\Omega} -W(r,z) [(\partial/\partial r) \{ k(\partial\psi/\partial r) \} + (\partial/\partial z) \{ k(\partial\psi/\partial z) \} + f] 2\pi d\Omega \\ + \int_C Wk(\partial\psi/\partial n) 2\pi dC = 0 \end{aligned} \quad (25)$$

Integration of the first term of Eq. (25) by parts yields

$$\int_{\Omega} [k\{(\partial\psi/\partial r)(\partial W/\partial r) + (\partial\psi/\partial z)(\partial W/\partial z)\} - Wf] d\Omega = 0 \quad (26)$$

if ψ is selected to equal ψ_0 along the corresponding part of the boundary. The second term of Eq. (26) reduces to zero through the proper application of the boundary conditions.

The boundary conditions must be satisfied in different ways for different portions of the boundary. Along the inlet where $\alpha_1 = 0$ the conditions may be satisfied by specifying $(\partial\psi/\partial n)$ to be zero or by specifying the nodal values of ψ if the inlet conditions are conducive to calculating ψ for each node. Along the shroud and along the hub the value of ψ must be specified as $\psi = (m/2\pi)$ at the shroud and $\psi = 0$ at the hub. For nodes at the exit plane, the condition that $(\partial\psi/\partial n) = 0$ is required.

C. APPLICATION OF THE FINITE ELEMENT METHOD

The first step is to discretize the region into subregions or elements. Within each element the unknown stream function, ψ , and the coordinates r and z are assumed to have the following polynomial variations:

$$\psi(r, z) = \sum_{i=1}^n \psi_i N_i(\xi, \eta) \quad (27a)$$

$$r = \sum_{i=1}^n r_i N_i(\xi, \eta) \quad (27b)$$

$$z = \sum_{i=1}^n z_i N_i(\xi, \eta) \quad (27c)$$

where n = number of nodes in the element

N_i = the shape or (trial) function for node i

ψ_i = value of ψ at node i

r_i = value of r at node i

z_i = value of z at node i

Equations (27b) and (27c) imply a geometrical as well as functional transformation or mapping, as shown for the present case in Fig. 4.

The second step in the process is the selection of the weight and shape functions. The shape functions are defined when the particular type of finite element is selected for the computational grid [Ref. 16]. The eight-noded quadrilateral was used in the present program and its associated shape functions were entered in a subroutine. The weight function is independent of the shape function and may be chosen at the discretion of the individual. In the present case the standard Galerkin technique was employed and therefore, the weight functions were defined as being equal to the shape functions, so that

$$W(r,z) = N(r,z) \quad (28)$$

Equation (26) may now be expressed in the following form:

$$\int_E \left\{ k \left[(\partial N_j / \partial r) \sum_{i=1}^n (\partial N_i / \partial r) + (\partial N_j / \partial z) \sum_{i=1}^n (\partial N_j / \partial z) \sum_{i=1}^n (\partial N_i / \partial z) \right] - N_j f(r,z) \right\} d\Omega = 0 \quad (29)$$

where $\int_E$ represents the integral over an element. In matrix notation this becomes

$$[K]^e \{\delta\}^e = \{f\}^e \quad (30)$$

where

$$k_{ij}^e = \int_E k(r,z) [(\partial N_j / \partial r)(\partial N_i / \partial r) + (\partial N_j / \partial z)(\partial N_i / \partial z)] d\Omega \quad (31a)$$

$$f_i^e = \int_E N_i f(r,z) d\Omega \quad (31b)$$

and

$$\delta_i = \psi_i \quad (31c)$$

The summation of the elemental contributions over the entire region yields the global system of equations needed to solve the problem. In matrix notation the global system of equations is expressed as

$$[K] \{\delta\} = \{f\} \quad (32)$$

where

$$[K] = \sum_{i=1}^m [K]_i^e \quad (33a)$$

$$\{\delta\} = \sum_{i=1}^m \{\delta\}_i^e \quad (33b)$$

$$\{f\} = \sum_{i=1}^m \{f\}_i^e \quad (33c)$$

and m = number of elements in the mesh

$$\delta_i = \psi_i$$

$[K]$ = system's stiffness matrix

$\{f\}$ = system's right-hand side vector

Since k_{ij} and f_i depend on the unknown solution ψ , Eq. (32) is a nonlinear differential equation to be solved by an iterative procedure. The details of the procedure are contained in section V.

D. NUMERICAL INTEGRATION TECHNIQUE

In general, problems are analyzed using a coordinate system in which the boundary conditions can be written and satisfied in the simplest possible way. For problems with irregularly shaped boundaries and/or mixed conditions along different portions of the boundary, obtaining numerical solutions in the original coordinate system can be a formidable task. Very often a scheme must be found to transform the derived equations into a new coordinate system that conforms to the requirements of standard numerical techniques. Traditional transformation techniques tend to be complicated exercises in algebra that require extensive reformulation for each geometry or type of boundary condition. The power of the finite element method is the automatic inclusion of a transformation of the geometry and the function to a rectangular domain where a variety of integration techniques may be employed. This can be seen in Fig. 4, which illustrates what is implied by Eqs. (27a), (27b), and (27c). The method can handle extremely complicated boundary conditions

and shapes with ease and is limited only by the type of element selected by the individual.

The quadratic properties of the eight-node element allows curved boundaries in the physical domain so long as the section of the boundary included within a single element does not have a point of inflection. The use of a quadratic element also ensures continuity of the approximated function along the elemental boundaries regardless of the direction of approach from within the mesh. The specific numerical technique used in the program is discussed in the following section.

1. Stiffness Matrix Evaluation

In section C the following relationship was derived for the individual elements of the stiffness matrix, $[K]$:

$$k_{ij}^e = \int_E k(r,z) [(\partial N_j / \partial r)(\partial N_i / \partial r) + (\partial N_j / \partial z)(\partial N_i / \partial z)] d\Omega \quad (31a)$$

In order to take advantage of well established numerical integration techniques, Eq. (31a) must be transformed from the (z,r) domain and its irregular elemental boundaries to the rectangular (ξ,η) domain. Equations (27a) through (27c) describe the variation of the function and the (r,z) coordinate values in the (ξ,η) plane. In order to transform Eq. (31a) it is necessary to determine the relationship of the variations of the derivatives in the two domains. These relationships can be derived in a straightforward manner through the use of the chain rule as follows:

$$(\partial N_i / \partial \xi) = (\partial N_i / \partial z) (\partial z / \partial \xi) + (\partial N_i / \partial r) (\partial r / \partial \xi) \quad (34)$$

and

$$(\partial N_i / \partial \eta) = (\partial N_i / \partial z) (\partial z / \partial \eta) + (\partial N_i / \partial r) (\partial r / \partial \eta) \quad (35)$$

Equations (34) and (35) can be combined in matrix form as

$$\begin{Bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{Bmatrix} = \begin{bmatrix} \frac{\partial z}{\partial \xi} & \frac{\partial r}{\partial \xi} \\ \frac{\partial z}{\partial \eta} & \frac{\partial r}{\partial \eta} \end{bmatrix} \begin{Bmatrix} \frac{\partial N_i}{\partial z} \\ \frac{\partial N_i}{\partial r} \end{Bmatrix} \quad (36)$$

Through the selection of the type element to be used in the mesh, $N_i(\xi, \eta)$ is a known function [Ref. 16], which makes possible the computation of the left-hand side vector for any point within the element boundaries. Similarly, by taking the appropriate derivatives of Eqs. (27b) and (27c) the 2x2 matrix, known as the Jacobian matrix $[J]$, can be determined. It follows that the r and z derivatives of the shape/weight functions can be determined for any point of an element from

$$\begin{Bmatrix} \frac{\partial N_i}{\partial z} \\ \frac{\partial N_i}{\partial r} \end{Bmatrix} = \begin{bmatrix} \frac{\partial z}{\partial \xi} & \frac{\partial r}{\partial \xi} \\ \frac{\partial z}{\partial \eta} & \frac{\partial r}{\partial \eta} \end{bmatrix}^{-1} \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{Bmatrix} \quad (37)$$

An examination of Eqs. (27a) through (27c) shows that once the derivatives of the shape/weight functions are known for a point then it is a simple procedure to determine the

derivatives of any other elemental property that has a value specified at the nodes.

The final relationship that is needed for the transformation is the relationship between the differential change in the coordinate directions of the (r,z) plane and the (ξ,η) plane. As shown by Kaplan [Ref. 19], the required relationship is

$$dzdr = |J|d\xi d\eta \quad (38)$$

Equation (31a) can now be transformed for integration in the (ξ,η) plane to the form

$$k_{ij}^e = \int_{-1}^1 \int_{-1}^1 \sum_k^n N_k k_k ([B]^T [B]) |J| d\xi d\eta \quad (39)$$

where

$$[B] = [J]^{-1} \begin{Bmatrix} \frac{\partial N}{\partial \xi} \\ \frac{\partial N}{\partial \eta} \end{Bmatrix} \quad (40)$$

$$\text{and} \quad k = (1/\rho r b) \quad (41)$$

The Gauss-Legendre method of numerical integration was used to obtain a solution to Eq. (39). It was selected because its determined accuracy was easily determined and the simple summing procedure used in the solution could be efficiently coded. A one dimensional example is used here to illustrate the use of the method.

The definite integral

$$I = \int_{-1}^1 \phi(\xi) d\xi \quad (42)$$

may be written in the form

$$I \approx W_1\phi_1 + W_2\phi_2 + W_3\phi_3 + \dots + W_n\phi_n \quad (43)$$

where $\phi_i = \phi(\xi_i)$

W_i = Gaussian weight function for ξ_i

The values of the points, called abscissas, and their corresponding weighting function values are catalogued for use. The number of points to be used is determined by the order of the function to be approximated. In general, a polynomial of $(2n - 1)$ is integrated exactly by the use of n Gauss points. In two dimensions the Gauss method can be written as

$$I = \int_{-1}^1 \int_{-1}^1 \phi(\xi, \eta) d\xi d\eta \quad (44)$$

which can be written as

$$I \approx \sum_{i=1}^n \sum_{j=1}^m W_i W_j \phi(\xi_i, \eta_j) \quad (45)$$

Equation (39) can now be written in a form that can be coded for solution by the computer as

$$k_{ij}^e = \sum_k \sum_l W_k W_l \sum_m N_m k_m ([B]^T [B]) |J| \quad (46)$$

The scheme used in the program is a two dimensional, three-point Gauss integration. A detailed description of the evaluation of $[B]^T[B]$ and the method used to obtain k_m is contained in the description of subroutine STIFF.

2. Right-hand Side Vector Evaluation

In Section C the following relationship for f was derived:

$$f_i^e = \int_E N_i f(r,z) d\Omega \quad (31b)$$

By using the techniques of Section D1, Eq. (31b) can be replaced by

$$f_i^e = \sum_j \sum_k W_j W_k \sum_i N_i \sum_\ell N_\ell F_\ell |J| \quad (47)$$

A more detailed description of the specific methods used to solve Eq. (47) is contained in the description of subroutine FCAL.

IV. DESCRIPTION OF PROGRAM MESHGEN

A. MAIN PROGRAM DESCRIPTION

MESHGEN was developed in order that the required inputs for the program TURBO could be generated in a fast, accurate, and conceptually correct manner. The program generates an eight-node isoparametric element mesh, computes the related connectivity matrix, defines the type of region enclosed by each element, computes the tangential blockage factor and an estimate of the stream function for each node in the mesh, and computes the thermodynamic conditions and velocity at the inlet. The inputs required to operate the program are the mass flow rate, total temperature, and total pressure at the inlet, the blading and machine geometries, RPM, C_p , γ , R , and scaling constants for the plot of the mesh. The blading geometries must be coded in the program as a subroutine that uses approximations or design information to express the blade variables as functions of radius. The user provides the other information in response to interactive prompts. The program has two modes of operation, one which generates a complete mesh and all of the associated information and another which uses a previously generated mesh to compute the changes in specific arrays that result from a change in the inlet conditions.

The program is completely general and may be used for either single-stage axial compressor or turbine, and, with very minor modifications, can be expanded for use with multi-stage machines. The mesh size that can be generated by the program is limited only by computer storage considerations. To use the program for another machine, the user is required to replace subroutine TASK1 with a subroutine that can compute the tangential blockage factor, b , for the desired blading. The functioning of the program and its subroutines for both modes of operation is described in the section B.

The program's algorithm in outline form is as follows:

Algorithm:

Determine the value of the appropriate operating conditions and whether a new mesh is desired (Subroutine INIT1).

Obtain the coordinates of the super element corners and a description of the division of the super elements into the final mesh. Compute the storage allocation parameters (Subroutine INPUT).

Compute the (r,z) coordinates for all nodes in the mesh (Subroutine TMESH).

Compute the connectivity relationships for the mesh and determine the beginning and ending node numbers for the rotor and stator (Subroutine CONEC).

Compute the array of node numbers where the value of ψ is to be specified. Compute an initial estimate of the nodal stream function distribution and call subroutine FLOFCT to determine the inlet conditions (Subroutine INIT2).

Compute the nodal tangential blockage factor, b , for the rotor and stator nodes (Subroutine TASK1).

Place the computed values in disk storage (Subroutine FILGEN).

Plot the generated mesh on the Tektronix 618 terminal for inspection (Subroutine MPLOT).

END

B. SUBROUTINE DESCRIPTIONS

1. Subroutine INIT1

Subroutine INIT1 obtains the value of thermodynamic variables and plotting parameters that are required for the program in either mode of operation. The user is required to provide the values of the mass flow (lbm/sec), total temperature ($^{\circ}\text{R}$), and total pressure (psia), ratio of specific heats (γ), gas constant (ft-lbf/lbm- $^{\circ}\text{R}$), and the specific heat at constant pressure (BTU/lbm- $^{\circ}\text{R}$). The scaling constants are convenient values of r and z used to frame the plot of the mesh. If a new mesh is not desired, the program exits the subroutine.

Subroutine INIT1 determines if a new mesh is to be generated by the response of the user to an interactive question. If a mesh is to be generated, the subroutine obtains some preliminary information about the flow region. Figure 5 shows how the user must first divide the flow region into a coarse mesh known as super elements, recording the (z,r) coordinates of the corner points. The minimum number of super elements for a single-stage compressor is five so that the three duct regions, the rotor, and the stator may be represented. The maximum number of super elements and the subsequent division into mesh elements is limited only by the

storage limitations of the machine. In practice, the maximum number of elements is limited by the number of equations that may be solved by the program TURBO. It is also limited by the fact that only one super element may be used to describe a rotor or duct region and that the super elements for these regions may only be further subdivided into a single column of elements. The latter restrictions are for compatibility with the computation procedures used in the program TURBO. A decision must then be reached on what subdivision of the super elements will provide a mesh that is sufficiently fine to yield accurate results efficiently. Once the flow region has been divided into super elements and a determination as to the total number of rows and columns of mesh elements has been made, the user can input the appropriate values in response to the prompts provided by the program.

2. Subroutine INPUT

Subroutine INPUT uses interactive prompts to obtain a description of the turbomachine's flow passage geometry and the desired mesh characteristics. The user must provide the program with the coordinate values of the super element nodes as shown in Fig. 5. The values are entered as nodal pairs on a station-by-station basis. The first (z,r) coordinate entered is the node on the shroud and the second lies on the hub. The program then asks the user to identify the type of region enclosed in each super element and into how many columns each super element is to be divided.

Enough information is now available for the program to compute and store the information required for the program TURBO. Subroutine INPUT stores the responses to the interactive prompts, determines the values of the storage location pointers, and determines if any storage limitation has been exceeded. If any storage limitation is exceeded, the subroutine calls the appropriate error subroutine to halt execution. The interactive portion of the subroutine is omitted if a new mesh is not desired. However, the values of the pointers are calculated and storage requirements are evaluated as before. A listing of the pointers and the corresponding variables is given in Appendix A.

3. Subroutine TMESH

Subroutine TMESH computes the nodal coordinate values from the information obtained in subroutines INIT1 and INPUT. The subroutine uses linear interpolation in the axial direction and a quadratic scheme in the radial direction. The radial interpolation scheme maintains the difference in the squares of the radius of the nodes as a constant. This allows the assumption of equal mass flow between the nodes for uniform axial velocity which is used to determine the initial estimate of the nodal stream function distribution. The nodes are numbered with the assumption that the fluid flow is from left-to-right in the mesh and that the number of columns of elements is greater than or equal to the number of rows of elements. The node at the inlet shroud is labeled

1 and the node at the outlet hub is labeled n for an n node mesh. The numbering proceeds on a column-by-column basis from top-to-bottom. The total number of rows, columns, and mesh elements is displayed to the user. The mesh computations are omitted if a new mesh is not created and elemental count information is displayed as before.

4. Subroutine CONEC

Subroutine CONEC generates the connectivity matrix for the mesh. The connectivity matrix is used to keep track of which nodes are assigned to which elements and the arrangement of the assigned nodes within the element. The connectivity relationships for a three element stack is shown in Fig. 3. Additionally, the subroutine determines the first and last nodes of the rotor and stator.

5. Subroutine INIT2

Subroutine INIT2 stores the node numbers for nodes where the value of ψ is specified as a known quantity. The array is used in the program TURBO to apply the boundary conditions. The array is not computed if a new mesh is not desired. For either mode of operation, subroutine INIT2 computes the values of the inlet thermodynamic variables, the inlet axial velocity, and an initial estimate of the nodal stream function distribution. Subroutine FLOFCT is used to calculate the inlet conditions and is described in the next section. The initial stream function is computed from the boundary conditions at the shroud and hub. Along

the shroud, ψ is specified to be equal to $(\dot{m}/2\pi)$ and along the hub to be zero. The value of ψ along the inlet is determined by a linear interpolation because of the quadratic node spacing in the radial direction. The remaining nodal values of ψ are obtained by an assumption that ψ is a constant along the streamwise boundaries of the elements. The assumption is obviously in error, but it observes the boundary conditions and provides a reasonable first estimate to begin the iteration scheme used in the program TURBO.

6. Subroutine FLOFCT

Subroutine FLOFCT computes the inlet conditions for a passage with a specified geometry, mass flow rate, total temperature, total pressure, and an assumed uniform inlet velocity. The method followed is the "total flow function" formulation proposed by Shreeve [Ref. 20]. The total flow function is defined as the ratio of the mass flux to the limiting or stagnation mass flux. The following definitions and equations are required for the method:

$$V_t = [2H]^{0.5} \quad (48)$$

$$X = V/V_t \quad (49)$$

$$T/T_t = 1 - X^2 \quad (50)$$

$$p/p_t = (1 - X^2)^{\frac{\gamma}{(\gamma-1)}} \quad (51)$$

$$\rho/\rho_t = (1 - X^2)^{\frac{1}{(\gamma-1)}} \quad (52)$$

From the definition of the total flow function, ϕ , it follows that

$$\phi = \rho V / \rho_t V_t = X(1 - X^2)^{\frac{1}{\gamma-1}} \quad (53)$$

The value of ϕ at the inlet can be calculated at the inlet from the assumed uniform conditions by the expression

$$\phi_1 = \dot{m} / (\rho_t V_t A) \quad (54)$$

The value of X at the inlet is found through the following Newtonian iteration:

$$\phi_1 = \dot{m} / (\rho_t V_t A) \quad (54)$$

Assume $X = 0.1$ to assure the selection of the subsonic root.

Calculate: $\phi = X(1 - X^2)^{\frac{1}{\gamma-1}} \quad (53)$

and $d\phi/dX = \{1/X - 2X/[(\gamma - 1)(1 - X^2)]\}$

Test $|\phi_1 - \phi| < \epsilon$

If the test fails then calculate

$$X = X + (\phi_1 - \phi) (d\phi/dX)$$

and recalculate ϕ until convergence is reached. Once convergence is reached the inlet conditions are computed by equations (50) through (52).

7. Subroutine TASK1

Subroutine TASK1 computes the nodal tangential blockage factor for the blading of the NASA TASK1 transonic compressor. The value of the blockage factor is determined by

$$b = 1 - t/s$$

(12a)

The values of t and s are obtained by approximations to the known blading geometry that are expressed as functions of radius. The maximum thickness of the blade is artificially defined to be at the mid-point of the chordline to ensure that the factor is accounted for in the calculations in the program TURBO. This artificiality could easily be removed through a modification to the axial interpolation scheme used in rotor and stator super elements. Subroutine TASK1 is the only machine dependent subroutine in use in the program and would need to be replaced with an appropriate substitute in order for the program to be used on another machine. The use of the subroutine is omitted if the user does not desire a new mesh.

8. Subroutine FILGEN

Subroutine FILGEN places the computed mesh parameters on disk storage for use in the program TURBO. If the limited mode of operation was selected by the user, the subroutine only updates the values of the parameters that change for a new inlet condition. A listing of the output parameters and their corresponding storage location is given in Appendix B.

9. Subroutine MPLOT

Subroutine MPLOT provides an on-line plot of the computed mesh on the Tektonix 618 graphics terminal. Figure 6 shows the 63 element, 222 node mesh used in the computations of the test cases. The subroutine displayed the mesh through

direct calls to the subroutines of the library plotting package, GRAFF. The subroutine's algorithm is as follows:

Algorithm:

Form two Real*4 arrays from the information in the r coordinate and the z coordinate arrays for plotting compatibility.

Sort the arrays and plot the streamwise boundaries of the mesh elements.

Sort the arrays and plot the transverse boundaries of the mesh elements.

END

10. Subroutine ERR1

Subroutine ERR1 is called by subroutine INPUT if the storage limitation for Real*8 variables has been exceeded. The subroutine displays the amount by which the limitation was exceeded and terminates the program's execution. The user response would be to increase the value of LIMR if possible or reduce the size of the mesh.

11. Subroutine ERR2

Subroutine ERR2 is called by subroutine INPUT if the storage limitation for Real*4 variables has been exceeded. The subroutine displays the amount by which the limitation was exceeded and terminates the program's execution. The user response would be to increase the value of LIM4 if possible or reduce the size of the mesh.

12. Subroutine ERR3

Subroutine ERR3 is called by subroutine INPUT if the storage limitation for Integer*4 variables has been exceeded.

The subroutine displays the amount by which the limitation was exceeded and terminates the program's execution. The user response would be to increase the value of LIM1 if possible or reduce the size of the mesh.

V. DESCRIPTION OF PROGRAM TURBO

Program TURBO solves the quasi-harmonic stream function radial equilibrium equation for flow in an axial compressor. The program uses the information computed by the program MESHGEN to calculate the desired thermodynamic information at all nodal points and displays selected values on a graphics terminal for inspection.

A. MAIN PROGRAM DESCRIPTION

The program obtains a solution of the equation

$$[K]\{\psi\} = \{f\} \quad (32)$$

An iterative scheme was adopted for this nonlinear problem, whereby an estimate of the stream function distribution is used to calculate values of the velocity components and thermodynamic variables at the nodes of the mesh. These computed values are then substituted into Eq. (32) and a new value of the stream function distribution is calculated. The estimate of the distribution is compared to the calculated value to determine if the solution has reached convergence according to the following criterion:

$$\epsilon > \left| \frac{\psi_i^n - \tilde{\psi}_i^{n+1}}{\tilde{\psi}_i^{n+1}} \right| \quad (47)$$

where ψ_i^n = estimate of ψ at node i

$\tilde{\psi}_i^{n+1}$ = solution for ψ at node i

If the maximum difference for all nodes is less than ϵ , the procedure is terminated. If the maximum difference exceeds some specified value of ϵ , the new estimate of ψ to be used for the next iteration is determined using

$$\psi_i^{n+1} = \psi_i^n + \alpha [\tilde{\psi}_i^{n+1} - \psi_i^n] \quad (48)$$

where ψ_i^{n+1} = new estimate of ψ for the next iteration

α = under relaxation factor required for convergence because of the strong nonlinear properties of Eq. (32).

The process of constructing the inputs required for Eq. (32) is repeated until convergence is obtained. The details of calculating the inputs and constructing the stiffness matrix and the right-hand side vector are contained in descriptions of the program's subroutines.

The program's algorithm follows in outline form:

Obtain the computational constants (SUBROUTINE RDATA).

Determine the values of the pointers used to partition the storage arrays (SUBROUTINE INIT1).

Set the initial values for all storage locations to 0.0 or 0 as appropriate (SUBROUTINE ZERO1).

Recall from storage the externally computed input values and initialize the inlet conditions (SUBROUTINE INPUT).

Calculate a velocity and thermodynamic variable distribution based on the assumed stream function distribution and the inlet conditions (SUBROUTINE DIST).

From the distributions obtained in DIST, calculate the right-hand side vector {f} (SUBROUTINE FCAL).

Using the density and blockage factor distributions, form the stiffness matrix [K] (SUBROUTINE STIFF).

Solve the system of linear equations to obtain a new stream function distribution (SUBROUTINE DSIMQ).

Place the solution vector in its proper storage location (SUBROUTINE REPLA).

Compare the original stream function distribution to the solution vector to determine the maximum difference in the distributions for all nodes (SUBROUTINE TEST).

Determine if the convergence criterion has been satisfied.

If convergence has not been obtained, perform the relaxation iteration to update the estimate of the stream function distribution (SUBROUTINE RELAX), prepare for another cycle (SUBROUTINE NOCON), and then return to SUBROUTINE DIST for further calculations.

If the convergence criterion has been satisfied, print the results (SUBROUTINE OUTPUT) and display selected information on the graphics terminal (SUBROUTINE MPLOT).

END

B. SUBROUTINE DESCRIPTIONS

Sections 1 through 23 provide a detailed description of the subroutines of program TURBO.

1. Subroutine RDATA

Subroutine RDATA is used to store the following computational constants:

- (a) Logical Input/Output variable NREAD and NWRITE.
- (b) Relaxation factor.
- (c) Limits for the storage arrays.

- (d) Three-point Gaussian abscissas and weighting values.
- (e) Constants used for conversions between different units.

2. Subroutine INIT1

Subroutine INIT1 determines the values of the pointers used to partition the Real*8, Real*4, and Integer*4 arrays and determines whether the storage limitations for any of the arrays has been exceeded. If the storage limitations have been exceeded the subroutine will halt execution by calling the appropriate error subroutine. A listing of the pointers and their corresponding array names is contained in Appendix C.

3. Subroutine ZEROI

Subroutine ZEROI sets the initial value of all arrays equal to 0.0 or 0 as appropriate.

4. Subroutine INPUT

Subroutine INPUT retrieves required input information from its corresponding disk storage location. The information must be placed in storage before running program TURBO. The usual method of generating the information and placing it in storage is through the use of the program MESHGEN.

Subroutine INPUT also initializes the inlet conditions to their proper values, and modifies the nodal blockage factors to account for end-wall boundary layer effects. No attempt was made to include a global method for calculating the blockage factors; rather, a method similar to the one used by Hirsch and Warzee [Ref. 4] was used. The full method

used by Hirsch and Warzee was to artificially reduce the size of the flow passage by reducing the boundaries of the mesh, followed by the application of a general blockage factor to the nodes of the mesh. In the program TURBO, no mesh modifications are made. The procedure followed was to apply a general blockage factor to all nodes, followed by the application of an additional blockage factor to nodes in the outer elements in the rotor, stator, and the passage in between. Though reasonable results were obtained by this method, the handling of the end-wall boundary layers remains the most obvious weakness in the code. This is addressed specifically in section VII.

5. Subroutine DIST

Subroutine DIST calculates the distributions of velocity, density, temperature, pressure, fluid flow angles, entropy, and enthalpy using the known blade and machine geometry, inlet conditions, and the assumed distribution of the stream function. Properties of nodes at the mid-line of the rotor or stator blades were assumed to have a value equal to the average of the inlet and exit conditions of the blade. The elemental calculations are accomplished through the control of subroutines SLINE, DUCT, ROTO, and STAT.

The following is the subroutine DIST's algorithm in outline form:

For each element in the mesh.

Determine type element for appropriate computations.

Duct Elements

For each node at stations 2 and 3:

Determine the location of the streamline and thermodynamic conditions at station one (Subroutine SLINE).

(If the element is along the machine exit plane, ensure that $(\partial\psi/\partial z) = 0$.)

Compute the thermodynamic conditions (Subroutine DUCT).

Assign the appropriate values to the proper storage location.

Rotor Elements

For each node at stations 1:

Determine the location of the streamline at station 3 and the $\partial\psi/\partial z$ and the $\partial\psi/\partial r$ at stations 1 and 3 (Subroutine SLINE).

Determine the inlet and outlet relative flow angles and the outlet absolute flow angle.

Compute the total-to-total pressure ratio and the adiabatic efficiency for the streamline (Subroutine ROTO).

Assign the appropriate values to the proper storage location.

For each node at stations 2:

Determine the location of the streamline and the thermodynamic conditions and the $\partial\psi/\partial z$ and the $\partial\psi/\partial r$ at stations 1 (Subroutine SLINE).

Determine the location of the streamline and the $\partial\psi/\partial z$ and the $\partial\psi/\partial r$ at station 3 (Subroutine SLINE).

Determine the inlet and outlet relative flow angles and the outlet absolute flow angle and the relative deviation angle.

Compute the thermodynamic conditions at station 3 (Subroutine ROTO).

Compute the value of all properties for the node as being the average of the values at station 1 and station 3.

Assign the appropriate values to the proper storage location.

For each node at stations 3:

Determine the location of the streamline and thermodynamic conditions at station 1 and the value of $\partial\psi/\partial z$ and $\partial\psi/\partial r$ at stations 1 and 3 (Subroutine SLINE).

Determine the inlet and outlet relative flow angles and the outlet absolute flow angle and compute the thermodynamic conditions (Subroutine ROTO).

Assign the appropriate values to the proper storage location.

Stator Elements

The stator algorithm is the same as the rotor algorithm except the outlet absolute flow angle is the only angle calculated. The inlet absolute flow angle is determined through interpolation.

6. Subroutine SLINE

In order to understand the functioning of this subroutine and others to follow, one must refer to the nomenclature used to describe the eight-node element. Figures 2 and 3 show the nomenclature clearly and Table 1 demonstrates the connectivity. All of the calculations in the program for the distributions of velocity, flow angles, and thermodynamic properties are founded on the assumption that the points in question lie on the same stream surface. Thus the objective of the subroutine is to obtain the location of a given value of the stream function at a specified station in the flow region. The location of the streamline is required in order to compute the variables used in $\{K\}$ and $\{f\}$.

Through the application of the boundary conditions, the nodes along the shroud are defined to lie on one stream surface and the nodes along the hub are on another. It is possible for all other nodes in the mesh to be on different stream surfaces. For these nodes an interpolation scheme must be followed to find the (ξ, η) coordinates of a specified stream surface at a given station in the mesh.

The solution sequence that the program follows starts at the top element of the first column of elements in the mesh and solves the thermodynamic and velocity conditions for all nodes in the element using the assumed stream function distribution and the specified inlet conditions. When the calculations for the first element are complete, the program continues down the column until the calculations are complete for the element along the hub. The program then sequences to the top element for the next column and continues until the calculations are complete for the last element in the mesh. In this sequence it is always possible to calculate the conditions at station 1 of an element for any interim nodal values of the stream function.

The process will be described by way of an example for one node as shown in Fig. 3.

Example: Find the (ξ, η) coordinates of the streamline that passes through node 7 of element 1, Node (1,7).

From the connectivity relationships it is known that

$$(\text{Node } (1,7)) = (14)$$

The value of (14) is known and the search is begun to find two nodal values of ξ at station 1 that bracket the desired value, (14). The program first tests to see if (Node (1,7)) is greater than (Node (1,5)). In this case it is not and the program would automatically shift and test to see if (Node (1,7)) is greater than (Node (2,5)). In this example the value is larger and the same test would be applied to (node (2,4)). Again the answer would be true. The program would then test to see if the value of (Node (2,3)) were larger than (Node (1,7)). The answer being true would signal the program that the location of the streamline had been bracketed and a half-interval technique would be applied to find the location. As shown in Fig. 4, the value of ξ for all locations along station 1 is -1. This fact is important for two reasons. One, with ξ known the program is only required to iterate on η to obtain convergence. Two, the Kronecker delta property of the shape functions means that only the shape functions at station 1 have nonzero values [Ref. 16]. For the half interval method, the program uses the average η of the most recent bracketing as its estimate for η . In this example the first estimate of η is equal to 0.5 and the solution for ψ at (-1., 0.5) can be written as

$$\psi(-1., 0.5) = N_3(-1., 0.5)\psi_3 + N_4(-1., 0.5)\psi_4 + N_5(-1., 0.5)\psi_5$$

The solution is then compared to $\psi(\text{Node } (1,7))$ to determine if the difference is less than some ϵ . If the difference exceeds ϵ , the new estimate for η becomes 0.25 or 0.75 depending on whether the solution is larger than or less than the value of $\psi(\text{Node } (1,7))$. The process is continued until convergence is reached. Once ξ and η for the streamline location at station 1 are known, all of the properties for station 1 can be determined. (The same method is used to find the location of the streamline at station 3 for rotor and stator elements.) Having determined the coordinates of streamlines at all desired locations it is possible to calculate the required $(\partial N_i / \partial r)$ and $(\partial N_i / \partial z)$. The computed inlet and exit coordinates and conditions are then passed to subroutine DIST for use in subroutines DUCT, ROTO, and STAT as appropriate for the calculation of the conditions at Node (1,7).

The following is the subroutine's algorithm in outline form:

Duct Element

If the node being investigated is on station one, exit the subroutine.

For stations two and three, determine the streamline coordinates and the thermodynamic conditions at station one and compute the $\partial\psi/\partial z$ and $\partial\psi/\partial r$ at the node. Exit the subroutine.

Rotor/Stator Element

If the node being investigated is on station one, set all inlet thermodynamic variables equal to the corresponding

nodal value and compute the $\partial\psi/\partial z$ and the $\partial\psi/\partial r$ at station three. Exit the subroutine.

For station two, determine the streamline coordinates and the thermodynamic conditions at station one and compute the streamline location and the $\partial\psi/\partial z$ and the $\partial\psi/\partial r$ at station three. Exit the subroutine.

For station three, determine the streamline coordinates and the thermodynamic conditions at station one and compute the $\partial\psi/\partial z$ and the $\partial\psi/\partial r$ at station three. Exit the subroutine.

7. Subroutine DUCT

Subroutine DUCT determines the values of temperature, pressure, and density for the elemental nodes at stations two and three. An iterative procedure is used with the knowledge that angular momentum is a constant in a duct. The initial estimate of the velocity at station 1 is made using the computed values of $\partial\psi/\partial z$, $\partial\psi/\partial r$, r , and b at the node (Subroutine SLINE) and by choosing the estimate of the density at the node to be equal to the density at station 1. The following sequence of calculations is repeated until convergence on a value of the exit velocity:

$$V_{m2} = (1/(\rho_2 r_2 b_2)) * [(\partial\psi/\partial z)_2^2 + (\partial\psi/\partial r)_2^2]^{0.5}$$

$$\alpha_2 = \tan^{-1} [(r_1 V_{m1} \tan \alpha_1) / (r_2 V_{m2})]$$

$$T_2 = T_{T1} - (\gamma - 1) / 2 * (V_{m2}^2 (1 + \tan^2 \alpha_2)) / (\gamma R G_c)$$

$$P_2 = P_{T1} (T_2 / T_{T1})^{**(\gamma / (\gamma - 1))}$$

$$\rho_2 = P_2 / (RT_2)$$

$$V_{m2n} = (1/(\rho_2 r_2 b_2)) * [(\partial\psi/\partial z)_2^2 + (\partial\psi/\partial r)_2^2]^{0.5}$$

$$\text{Test if } |V_{m2n} - V_{m2}| < \epsilon$$

The total conditions are then calculated from the static conditions and the computed velocity.

8. Subroutine ROTO

Subroutine ROTO calculates the change in the relative flow angles, the velocity, and the thermodynamic properties along a streamline across a (compressor) rotor element. The program uses the conditions at station one and the location of the streamline and the partial derivatives of the stream function at station 3, all of which were calculated in subroutine SLINE. The relationships in subroutine ROTO are derived from cascade correlations and known property relationships for a streamline in a rotor.

The first step is the calculation of the inlet and exit relative flow angles. For inlet flow without swirl, the relative inlet flow angle can be calculated using

$$\beta_1 = \tan^{-1}(U_1/V_{m1})$$

The incidence angle is calculated using the known blade geometry and inlet angle, since

$$i = \beta_1 - \kappa_1$$

where κ_1 is the angle formed between the tangent to the blade chordline and the axial direction. In order to calculate the exit relative flow angle one must determine the deviation angle, δ . The program uses the correlations and equations

derived by NASA [Ref. 21]. The specific sequence of equations, using the notation of Ref. 21, is as follows:

$$i_C - i_{2D} = f(M, r)$$

$$i_{2D} = (K_i)_t (K_i)_{SH} (i_0)_{10} + n\phi$$

$$i_{ref} = i_{2D} + (i_C - i_{2D})$$

$$\delta_{2D} = (K_\delta)_t (K_\delta)_{SH} (\delta_0)_{10} + (m/\sigma^b) + (i_C - i_{2D}) (d\delta/di)_{2D}$$

$$\delta_C - \delta_{2D} = f(M_R, r)$$

$$\delta_{ref} = \delta_{2D} + (\delta_C - \delta_{2D})$$

$$\delta = \delta_{ref} + (i - i_{ref}) (d\delta/di)_{2D}$$

$$\beta_2 = \beta_1 - \phi - i + \delta$$

The expressions used to approximate the NASA correlation curves were those obtained by Crouse of NASA Lewis and were provided to the author by Okiishi [Ref. 22].

When the inlet conditions and relative flow angles are known it is possible to determine the conditions at the rotor element exit. The initial exit velocity is obtained in the same way as in subroutine DUCT. The following sequence of equations is solved iteratively until convergence for the exit velocity is reached:

$$\alpha_2 = \tan^{-1} [(\omega r_2 - V_{m2} \tan \beta_2) / V_{m2}]$$

$$D = 1 - (W_2/W_1) + (r_1 W_{u1} - r_2 W_{u2}) / (2 W_1 \bar{r})$$

$$\tilde{\omega} = \text{curve fit to } \tilde{\omega}(D, \cos \beta, \sigma)$$

$$T_{E1} = T_{R1} + \omega^2(r_2^2 - r_1^2)/\text{constant}$$

$$P_{E1} = P_{T1} (T_{E1}/T_{T1}) * (\gamma/\gamma-1)$$

$$P_{R1} = P_{T1} (T_{R1}/T_{T1}) * (\gamma/\gamma-1)$$

$$P_{E2} = P_{E1} - \tilde{\omega}(P_{R1} - P_1)$$

$$W_2 = V_{m2}/\cos \beta_2$$

$$T_2 = T_{E1} - W_2^2/\text{constant}$$

$$P_2 = P_{E2} (T_2/T_{E1}) * (\gamma/\gamma-1)$$

$$\rho_2 = P_2/(RT_2)$$

$$V_{\min} = (1/(\rho_2 r_2 b_2)) * [(\partial\psi/\partial z)_2^2 + (\partial\psi/\partial r)_2^2]^{0.5}$$

$$\text{Test if } |V_{\min} - V_{m2}| < \epsilon$$

When convergence is reached the total conditions are calculated from the static conditions and the computed velocity. The value of the entropy change is calculated by:

$$S = R \ln(P_{T2}/P_{T1}) * \text{constant}$$

9. Subroutine STAT

Subroutine STAT calculates the stator element exit absolute flow angle and thermodynamic conditions using the knowledge that the total enthalpy is a constant across the stator. The initial estimate of the exit velocity is obtained in the same way as in subroutine DUCT. The following sequence of equations is used until convergence for the exit velocity is reached:

$$V_1 = [V_{m1}^2 (1 + \tan^2 \alpha_1)]^{0.5}$$

$$D = 1 - (V_2/V_1) + (r_1 V_{u1} - r_2 V_{u2}) / (2 \sigma V \bar{r})$$

$$\tilde{\omega} = \text{curve fit to } \omega(D, \cos \beta, \sigma)$$

$$P_{T2} = P_{T1} - \tilde{\omega}(P_{T1} - P_1)$$

$$T_2 = T_{T1} - (\gamma - 1) / 2 * (V_{m2}^2 (1 + \tan^2 \alpha_2)) / (\gamma R G)$$

$$P_2 = P_{T2} (T_2 / T_{T2})^{**(\gamma / \gamma - 1)}$$

$$\rho_2 = P_2 / (R T_2)$$

$$V_{\min} = (1 / (\rho_2 r_2 b_2)) * [(\partial \psi / \partial z)_2^2 + (\partial \psi / \partial r)_2^2]^{0.5}$$

$$\text{Test if } |V_{\min} - V_{m2}| < \epsilon$$

When convergence is reached the total conditions are calculated from the static conditions and the computed velocity. The value of the entropy change is calculated by the same method used by subroutine ROTO.

10. Subroutine FCAL

Subroutine FCAL uses the previously computed distributions of total temperature, entropy, enthalpy, axial velocity, and tangential velocity to compute the right-hand side vector for the global system of equations. In the absolute frame of reference, $f(r, z)$ can be expressed in the form

$$f(r, z) = (1/V_z) T(\partial s / \partial r) - \partial H / \partial r + (V_u / r) (\partial (r V_u) / \partial r) \quad (20)$$

The value of $f(r, z)$ within an element is determined using the following relationships:

$$T(z,r) = \sum_i^n N_i(\xi,\eta) T_i$$

$$H(z,r) = \sum_i^n N_i(\xi,\eta) H_i$$

$$s(z,r) = \sum_i^n N_i(\xi,\eta) s_i$$

$$V(z,r) = \sum_i^n N_i(\xi,\eta) V_i$$

$$V(z,r) = \sum_i^n N_i(\xi,\eta) V_i$$

$$r = \sum_i^n N_i(\xi,\eta) r_i$$

where the value of $N_i(\xi,\eta)$ is determined by the value of ξ and η for a specified Gaussian integration point within the element. The required partial derivatives are found in a simple and direct way. To illustrate, the $(\partial H/\partial r)$ is derived as follows:

$$H(z,r) = \sum_i^n N_i H_i$$

$$(\partial H/\partial r) = \sum_i^n (\partial N_i/\partial r) H_i + \sum_i^n N_i (\partial H_i/\partial r)$$

and since H_i is a constant then

$$\partial H/\partial r = \sum_i^n (\partial N_i/\partial r) H_i$$

where $(\partial N_i/\partial r)$ is found by Eq. (37), and H_i is the appropriate nodal value of H . The radial variations in entropy and angular momentum are found in the same manner. The same procedure is followed for the values of rothalpy and relative tangential velocity for rotor elements.

It is now possible to calculate the quantities in the expression for the right-hand side vector at a point. All that remains is to apply an integration technique to

obtain the value over an element and to assemble the resulting local contributions into the global equations. As shown in section III.D.2, the local contribution for node i in an element can be expressed as:

$$f_i^e = \sum_{j,k} W_j W_k \sum_i N_i \sum_{\ell} N_{\ell} f_{\ell} |J| \quad (47)$$

In the program Eq. (47) is modified to

$$f_i^e = \sum_m^9 W_m \sum_i^n N_i \sum_k^n N_k f_k |J|$$

where the Gaussian abscissas and the corresponding product of the weight functions are grouped into three one-dimensional arrays. At the completion of the summing process, the local contribution for f has been calculated for each node in the element. The global system is then updated by adding the local contributions to the global values through the use of the connectivity relationships.

The following is the subroutine's algorithm in outline form:

Algorithm:

Iterate for each element in the mesh.

Iterate for each Gaussian point.

Find shape functions, $|J|$, and $[J]^{-1}$.

Find V_z , T_T , V_u , r , rV_u , $(\partial s / \partial r)$, and $(\partial H / \partial r)$.

Compute the contributions of the value f at the Gauss point to the value of f at each node of the element.

Upon completion of the Gaussian integration, add the local contribution to the global system.

END

The same algorithm is followed for rotor elements with the appropriate substitutions of H_R and W_u .

11. Subroutine STIFF

Subroutine STIFF uses the computed distributions of density and blockage factors to form the stiffness matrix for the global system of equations. It was shown earlier that the contribution to the elemental stiffness is expressed as

$$k_{ij}^e = \int_E k(r,z) [(\partial N_j / \partial r)(\partial N_i / \partial r) + (\partial N_j / \partial z)(\partial N_i / \partial z)] d\Omega \quad (31a)$$

$$\text{where} \quad k(r,z) = (1/\rho r b) \quad (19)$$

Again, the elemental properties are considered to have a polynomial variation of the form

$$\rho(z,r) = \sum_i^n N_i \rho_i, \quad r_i = \sum_i^n N_i r_i, \quad \text{and} \quad b(z,r) = \sum_i^n N_i b_i$$

As shown earlier, Eq. (31a) can be converted by the Gauss-Legendre method to

$$k_{ij}^e = \sum_{k\lambda} W_k W_\lambda \sum_m N_m k_m ([B]^T [B]) |J| \quad (46)$$

The value of k_m is determined from the definitions of ρ , b , and r by the same methods used in subroutine FCAL. The evaluation of $[B]^T [B]$ is a simple matter to perform. The

matrix [B] is simply the column vector $\{(\partial N_i / \partial z), (\partial N_i / \partial r)\}$. Therefore, $[B]^T [B]$ can be written $\{(\partial N_i / \partial r)(\partial N_j / \partial r) + (\partial N_i / \partial z)(\partial N_j / \partial z)\}$. The value of $(\partial N_i / \partial z)$ and $(\partial N_i / \partial r)$ is found for all nodes in the element in one step through the use of subroutine JACOB. The matrix product can be evaluated at a point (z,r) as

$$\sum_i^n \sum_j^n [(\partial N_i / \partial r)(\partial N_j / \partial r) + (\partial N_i / \partial z)(\partial N_j / \partial z)]$$

By using the same Gaussian weighting scheme used in subroutine FCAL, Eq. (46) may be written in the form

$$k_{ij}^e = \sum_k^9 W_k \sum_\ell^n N_\ell k_\ell \sum_{ij}^{nn} [(\partial N_i / \partial r)(\partial N_j / \partial r) + (\partial N_i / \partial z)(\partial N_j / \partial z)] |J|$$

The resulting 8x8 elemental matrix is then added to the global stiffness matrix through the connectivity relationships.

Up to this point [K] and {f} have been assembled without regard to the boundary conditions except at the exit plane where the $(\partial \psi / \partial n) = 0$ was enforced explicitly during the procedures used by Subroutine DIST. Care must be taken to ensure that the boundary conditions for the other three segments of the boundary are not violated. As shown in section III.B, the boundary condition for the nodes along the shroud, along the hub and at the inlet plane of the machine is that the value of ψ is specified. If the value of ψ is specified at these locations the Eq. (32) must be modified so that ψ is no longer free at these nodes. A standard technique is

employed to remove individual equations from a system of equations when the degree of freedom represented by the individual equations has been removed.

The following is the subroutine's algorithm in outline form:

Algorithm:

Iterate for each element in the mesh.

Iterate for each Gaussian point.

Find shape functions, $|J|$, and $[J]^{-1}$.

Find the value of k at the Gauss point.

Compute the elemental stiffness matrix.

Upon completion of the Gaussian integration, add the local contribution to the global system.

Upon completion of the addition of the last element's contribution, modify the system of equations to include the boundary conditions.

END

12. Subroutine DSIMQ

Subroutine DSIMQ is a non-IMSL library, double precision subroutine that solves a set of n simultaneous equations of the form

$$[A]\{X\} = \{B\}$$

where $[A]$ is an $n \times n$ matrix

$\{X\}$ and $\{B\}$ are $n \times 1$ vectors.

13. Subroutine REPLA

Subroutine REPLA places the solution vector obtained from subroutine DSIMQ into its proper storage location.

14. Subroutine TEST

Subroutine TEST determines the maximum difference in the assumed nodal distribution of the stream function at the beginning of an iteration to the solution of the radial equilibrium equation calculated using the assumed distribution. The difference in the distributions at a node is defined as

$$\text{Diff} = \left| \left[(\psi_i^n - \tilde{\psi}_i^{n+1}) / \tilde{\psi}_i^{n+1} \right] \right|$$

where ψ_i^n is the assumed value at node i and $\tilde{\psi}^{n+1}$ is the calculated value for node i . Convergence is considered to be reached when the maximum difference for any node is less than a selected reference value, ϵ .

15. Subroutine RELAX

Subroutine RELAX performs the relaxation scheme to obtain an updated estimate of the stream function distribution for the next program iteration. The new estimate for stream function distribution is calculated as follows:

$$\psi_i^{n+1} = \psi_i^n + \alpha [\tilde{\psi}_i^{n+1} - \psi_i^n] \quad (48)$$

16. Subroutine NOCON

Subroutine NOCON prepares the program for the next iteration by setting the elements of the right-hand side vector, $\{f\}$, and the stiffness matrix, $[K]$, equal to zero.

17. Subroutine OUTPUT

Subroutine OUTPUT prints the computed nodal values of a majority of the velocities and thermodynamic properties. A listing of the values that are printed and the corresponding units is contained in Appendix D. A sample output listing is contained in Appendix G.

18. Subroutine MPLOT

Subroutine MPLOT uses the Tektronix 618 terminal to make an online graphical presentation of selected variables at the rotor inlet, rotor outlet, stator inlet and the stator outlet. Figures 7 through 20 provide examples of the plots available for display to the individual on request. The user is given the option of terminating the plotting sequence at any stage of the presentation through the use of interactive prompts.

The following is the subroutine's algorithm in outline form:

Algorithm:

Convert the appropriate variables to Real 4 for compatibility with the library plotting package GRAFF.

Determine the values of axial velocity, relative flow angles, total-to-total pressure ratio, and the adiabatic efficiency for the rotor inlet, display if requested.

Determine the values of axial velocity, and relative, absolute, and deviation angles for the rotor exit, display if requested.

Determine the values of axial velocity, absolute flow angles, and total-to-total pressure ratio for the stator inlet, display if requested.

Determine the values of axial velocity, and absolute and deviation angles for the stator exit, display if requested.

END

19. Subroutine SHAPE

Subroutine SHAPE calculates the eight nodal shape functions for a given point (ξ, η) . The equations for the nodal shape functions are:

$$N(1) = (\xi\eta + \xi^2 + \eta^2 + \xi^2\eta + \xi\eta^2 - 1)/4$$

$$N(2) = (1 + \eta - \xi^2 - \xi^2\eta)/2$$

$$N(3) = (-\xi\eta + \xi^2 + \eta^2 + \xi^2\eta - \xi\eta^2 - 1)/4$$

$$N(4) = (1 - \eta^2 - \xi + \xi\eta^2)/2$$

$$N(5) = (\xi\eta + \xi^2 + \eta^2 - \xi^2\eta - \xi\eta^2 - 1)/4$$

$$N(6) = (1 - \eta - \xi^2 + \xi\eta^2)/2$$

$$N(7) = (-\xi\eta + \xi^2 + \eta^2 - \xi^2\eta + \xi\eta^2 - 1)/4$$

$$N(8) = (1 - \eta^2 + \xi - \xi\eta^2 - 1)/2$$

The values of the shape functions are stored in the array SF, and are returned to the calling portion of the program.

20. Subroutine JACOB

Subroutine JACOB computes the partial derivatives of the shape functions with respect to ξ and η and computes the elements of the Jacobian matrix, [J], for a specific point (ξ, η) . The equations for the partial derivatives were obtained directly from the differentiation of the functions shown in the description of subroutine SHAPE. The arrays D

and E store the values of the $(\partial N/\partial \xi)$ and $(\partial N/\partial \eta)$ respectively. The Jacobian matrix is calculated by the following sequence of equations:

$$J(1,1) = (\partial z/\partial \xi) = \sum_{i=1}^n (\partial N_i/\partial \xi) z_i$$

$$J(1,2) = (\partial r/\partial \xi) = \sum_{i=1}^n (\partial N_i/\partial \xi) r_i$$

$$J(2,1) = (\partial z/\partial \eta) = \sum_{i=1}^n (\partial N_i/\partial \eta) z_i$$

$$J(2,2) = (\partial r/\partial \eta) = \sum_{i=1}^n (\partial N_i/\partial \eta) r_i$$

Arrays D and E and the Jacobian matrix are returned to the calling location in the program.

21. Subroutine ERR1

Subroutine ERR1 is called by subroutine INPUT if the storage limitation for Real 8 variables has been exceeded. The subroutine displays the amount by which the limitation was exceeded and terminates the program's execution. The user response would be to increase the value of LIMR if possible or reduce the size of the mesh.

22. Subroutine ERR2

Subroutine ERR2 is called by subroutine INPUT if the storage limitation for Real 4 variables has been exceeded. The subroutine displays the amount by which the limitation was exceeded and terminates the program's execution. The

user response would be to increase the value of LIM4 if possible or reduce the size of the mesh.

23. Subroutine ERR3

Subroutine ERR3 is called by subroutine INPUT if the storage limitation for Integer 4 variables has been exceeded. The subroutine displays the amount by which the limitation was exceeded and terminates the program's execution. The user response would be to increase the value of LIM1 if possible or reduce the size of the mesh.

VI. RESULTS AND DISCUSSION

A. PROGRAM VERIFICATION

Four operating conditions of the NASA TASK-1 compressor were used to test the capabilities of the programs MESHGEN and TURBO. In all cases the 63 element, 222 node mesh with an under-relaxation factor of 0.24 as recommended by Hirsch and Warzee [Ref. 4] were used. Selected portions of the results obtained are presented in Figs. 7 through 76. The points annotated as "observed values" were obtained from the material published in Refs. 23 and 24. The values attributed to Gavito were obtained from Ref. 8 and those attributed to Hirsch from Ref. 4. A discussion of the predictions for the various conditions are presented in the sections that follow.

1. Test Case 1

For test case 1 the operating point was defined as a rotor speed of 50% design speed and an inlet mass flow rate of 107.6 lbm/sec. The author was unable to locate this specific operating point in Ref. 23 or 24 and must assume that in Ref. 4 the mass flow rate was modified to conform to the end-wall boundary layer scheme described in that reference. Therefore, it was necessary to use the observed values published by Hirsch and Warzee in Figs. 21 through 30. The relative differences found between the present predictions and the reported observations are given in Table 1.

a. Comparison to the Work of Gavito

Figures 21, 22, 23 and 24 compare the results obtained by Gavito with the predictions of the present program. A significant improvement has been obtained at all locations. That this was possible was due in large measure to the solid foundation to the present work provided by Gavito's program and to the excellent documentation given in Ref. 8.

b. Comparison to the Work of Hirsch and Warzee

The predictions of the program TURBO compare quite favorably with those of Hirsch and Warzee. As shown in Figs. 25, 26, 29 and 30 the predictions of the two programs have almost identical average relative errors for the velocity profiles. The predictions of Hirsch and Warzee tend to have better agreement in the rotor and stator tip regions, while the program TURBO has slightly better agreement near the hub. The program TURBO's predictions had a 3.7% and 2.5% average error at the rotor inlet and exit respectively with a maximum error of 4.6% at the inlet and 7.0% at the outlet. The stator inlet velocity predictions had an average relative error of 2.6% and maximum error of 5.8% and the outlet predictions had a 2.0% average error with a maximum error of 6.0%. The prediction of both programs for the velocity profiles show excellent agreement with the observed values.

Hirsch and Warzee's program consistently produced flow angle predictions with closer agreement to the observed

values for the published rotor outlet angles. Though the two programs had the same average errors of 3.3° for the relative flow angles and 4.8° for the absolute flow angles, Hirsch and Warzee's program provided better qualitative distributions. The difference is clearly shown in Figs. 27 and 28.

Hirsch and Warzee did not publish predictions of total pressure ratios or adiabatic efficiencies, so the predictions made by TURBO for these parameters were not presented. It is assumed that no significant differences could have occurred because of the similarity of the results for the velocity profile and flow angles discussed earlier.

2. Test Case 2

Because of the apparent modification in the mass flow rate which was assumed in test case 1 and the lack of comparative data for all quantities predicted by the program TURBO, another operating point at 50% design speed was compared. The operating condition for case 2 was defined as a mass flow rate of 114.7 lbm/sec at a speed equal to 50% of design, which corresponded to reading 38 of Ref. 24. The results for case 2 are presented in Figs. 31 through 44, and a summary of the relative differences between the predictions and observations is contained in Table 2.

3. Test Case 3

Test case 3 corresponded to reading 45 of Ref. 24, which was defined as a speed equal to 70% design and a mass flow rate of 151.55 lbm/sec. When reviewing the results

presented for this case, one should note the decline in the agreement between the program's predictions and the observed values. It is the author's opinion that the degradation is primarily caused by two factors. The first is the formulation used to compute the meridional velocity change across the rotor and the other is the application of a single blockage factor to all nodes to account for the end-wall boundary layers. Both factors are much more significant at 70% design speed than they were at 50%. At 70% design speed the rotor tip relative Mach number is about 0.94. This would require the program to account for transonic effects at the tip. Second, the mass flow and absolute Mach number of the flow in all regions is significantly higher at 70% design speed. Therefore, it is unlikely that a single blockage factor will work satisfactorily in all regions of the machine. It is hoped that both areas will be addressed in any future work on the program.

The results for test case 3 are presented in Figs. 45 through 58 with the corresponding differences between the predictions and observations summarized in Table 3.

4. Test Case 4

The operating condition of test case 4 was at 80% design speed point with a mass flow rate of 174.54 lbm/sec, corresponding to reading 50 of Ref. 24. Figures 59 through 72 and Table 4 present the results of the program's predictions and comparisons to the observed values. As expected,

and for similar reasons to those cited in test case 3, the agreement between the program's predictions and observed values is significantly poorer than any of the three previous cases.

B. POINTS OF INTEREST

The results produced by the program are highly dependent on the value of the general blockage factor used to account for the end-wall boundary layers. This factor influences both the quality, in terms of agreement with observations, and the stability of the solution. Figures 73 through 76 show the axial velocity distributions for the rotor and stator for test case 4 with a blockage factor of 9% instead of the 6% factor used to obtain the results shown in Figs. 59 through 72. A comparison of corresponding velocity profiles clearly demonstrates the factor's pronounced influence on the program's solution. The general blockage factor also has a strong influence on the program's convergence rate. In some cases the factor can cause the program to become oscillatory or even divergent.

For the low speed cases of 50% and 70% design, additional blockage factors had to be applied to the duct element between the rotor and stator tips to obtain accurate results. As shown in the program listing for subroutine INPUT of program TURBO, the factors used for 50% design speed were much higher than the factors used for 70% design speed and that

no additional factors were used for 80% design speed. A global method of calculating the end-wall blockage factor must be incorporated if the program is to become independent of inputs other than physical constants.

The deterioration of the program's predictions with increasing Mach number and its failure to run for test cases with strong supersonic relative velocities at the tip demonstrate the need to provide the program with a method of handling supersonic relative velocities. Hirsch and Warzee [Ref. 15] showed a method of extending the radial equilibrium formulation used in the present code to supersonic flow. They presented comparisons of predictions obtained by this method to observations of the NASA TASK-1 transonic compressor at 100% design speed. The results were impressive and clearly showed that the method is valid for relative velocities in excess of Mach 1.4. It is strongly recommended that the first effort at improving the code be an effort to modify TURBO to include the method shown in Ref. 15.

VII. CONCLUSIONS AND RECOMMENDATIONS

A computer program based on the finite element technique has been developed and has been verified satisfactorily for computing flows through subsonic axial flow compressor stages. Minor modifications have been suggested to allow transonic stages to be calculated.

The code was written in such a way that it could be readily adapted to compute either turbines or compressors with multiple stages. Before such extensions are attempted however, the following specific recommendations are made to improve the present compressor code.

A. PROGRAM MESHGEN

1. Incorporate some of the two-dimensional techniques of Adamek [Ref. 25] to improve the efficiency of the code.
2. Review the code to find improvements in storage allocations and computational efficiencies.
3. Modify the program to track the first and last nodes of the rotor and stator as subscripted variables so that the program can be used to generate the appropriate mesh parameters for a multi-stage machine.
4. Convert subroutine MPLOT to the DISPLA system.

B. PROGRAM TURBO

1. Incorporate a method for the global calculation of the blockage and losses created by the end-wall boundary layers.

2. Convert the storage of [K] and the solution technique for the program to at least a symmetric banded scheme or if possible to a skyline equivalent scheme.
3. Test the program on a variety of machines and operating conditions.
4. Obtain expressions that approximate the NASA correlation curves for 65-series blading.
5. Incorporate methods for the prediction of stall/surge.
6. Take advantage of the modular form of the program and include a variety of correlation techniques as a user selected option.
7. Review the program for improved storage and computational techniques. Specifically, determine ways to take fuller advantage of the dynamic dimensioning scheme [Ref. 16] used by the program.
8. Convert the rotor inlet calculations to allow the value of β_1 to have nonzero values so that the program can be extended to multi-stage analysis.
9. Modify the use of the values of the beginning nodes of the rotor and stator to subscripted variables so that the analysis can be extended to multi-stage machines.
10. Convert subroutine MPLOT to the DISPLA system.
11. Develop iterative schemes for calculating the flow angle, the velocity distribution changes and the thermodynamic property changes across a turbine rotor and stator for inclusion in subroutines ROTO and STAT.

LIST OF REFERENCES

1. Naval Postgraduate School Technical Report NPS-57MA70081A, Computer Program for Prediction of Axial Flow Turbine Performance, by E. Macchi, August 1970.
2. Cirone, R., Computer Evaluation of the On- and Off-Design Performance of an Axial Air Turbine, M.S. Thesis, Naval Postgraduate School, Monterey, California, March 1981.
3. Naval Postgraduate School TPL Technical Note 82-02, Verification of a Single Stage Axial Turbine Performance Prediction Program for the HP 21-MX Computer System, by J. A. Ferguson, May 1982.
4. Hirsch, C., and Warzee, G., "A Finite Element Method for the Axisymmetric Flow Computation in a Turbomachine," International Journal for Numerical Methods in Engineering, v. 10, pp. 93-113, 1976.
5. Adler, D., and Krimerman, Y., "The Numerical Calculation of the Meridional Flow Field in Turbomachines Using the Finite Element Method," Israel Journal of Technology, v. 12, pp. 268-274, 1974.
6. Hirsch, C., and Deconinck, H., Finite Element Methods for Transonic Blade-to-Blade Calculation in Turbomachines, ASME Paper 81-GT-5.
7. Bettencourt, J., Finite Element Analysis Program (FEAP) for Conduction Heat Transfer, Engineer's Thesis, Naval Postgraduate School, Monterey, California, December 1979.
8. Gavito, F. V., The Finite Element Method Applied to Flows in Turbomachines, M.S. Thesis, Naval Postgraduate School, Monterey, California, December 1976.
9. NACA TN2604, A General Theory of Three-dimensional Flow in Subsonic and Supersonic Turbomachines of Axial, Radial, and Mixed Flow Types, by C. H. Wu, 1952.
10. Smith, H. L., "The Radial Equilibrium Equation of Turbomachinery," Transactions of the ASME, Journal of Engineering for Power, v. 88A, pp. 1-12, 1966.

11. Novak, R. A., "Streamline Curvature Computing Procedures for Fluid Flow Problems," Transactions of the ASME, Journal of Engineering for Power, v. 89A, p. 478, 1967.
12. Aeronautical Research Council R&M 3509, A Digital Computer Program for the Through-flow Fluid Mechanics in an Arbitrary Turbomachine Using a Matrix Method, by H. Marsh, 1966.
13. Carleton University, Division of Aerothermodynamics, ME/A 73-1, Axial Flow Compressor Analysis Using a Matrix Method, by W. R. Davis and D. A. J. Miller, 1973.
14. Vavra, M. H., Aero-Thermodynamics and Flow in Turbomachines, Kriger, 1981.
15. Hirsch, C., and Warzee, G., "A Finite Element Method for Through-flow Calculations in Turbomachines," Transactions of the ASME, Journal of Fluids Engineering, v. 98, pp. 403-422, 1976.
16. Cook, R. D., Concepts and Applications of Finite Element Analysis, pp. 2, 399-401, Wiley, 1981.
17. Huebner, K. H., The Finite Element Method for Engineers, Wiley, 1975.
18. Zienkiewicz, O. C., The Finite Element Method, 3rd Ed., McGraw-Hill, 1977.
19. Kaplan, W., Advanced Calculus, p. 93, Addison-Wesley, 1952.
20. Naval Postgraduate School Technical Report NPS-57Sf74081, Flow Into a Transonic Compressor Rotor, Part 1, Analysis, by R. P. Shreeve, August 1974.
21. NASA SP-36, Aerodynamic Design of Axial Flow Compressors, by NASA Staff, 1965.
22. Okiishi, T. H., private communication, June 1982.
23. NASA CR-72806, Evaluation of Range and Distortion Tolerance for High Mach Number Transonic Fan Stages, Volume 1, by C. C. Koch, K. R. Bilwakesh, and V. L. Doyle, August 1971.
24. NASA CR-72964, Evaluation of Range and Distortion Tolerance for High Mach Number Transonic Fan Stages, Volume 2, by C. C. Koch, K. R. Bilwakesh, and V. L. Doyle, August 1971.

25. Adamek, J. M. An Automatic Mesh Generator Using Two- and Three-dimensional Isoparametric Finite Elements, M.S. Thesis, Naval Postgraduate School, Monterey, California, 1973.

TABLE 1

Connectivity Relationships for Figure Three

<u>Element Number</u>	<u>Local Node Number</u>	<u>Global Number</u>
1	1	12
1	2	8
1	3	1
1	4	2
1	5	3
1	6	9
1	7	14
1	8	13
2	1	14
2	2	9
2	3	3
2	4	4
2	5	5
2	6	10
2	7	16
2	8	15
3	1	16
3	2	10
3	3	5
3	4	6
3	5	7
3	6	11
3	7	18
3	8	17

TABLE 2

Comparison of Program Predictions with NASA Task-1
Compressor Measurements at 50% Design Speed

	<u>Average</u> <u>Difference</u>	<u>Maximum</u> <u>Difference</u>
Rotor Inlet		
Axial Velocity	4.6%	7.2%
Relative Angles	1.5°	3.1°
Total Pressure Ratio	0.7%	1.4%
Efficiencies	4.3%	11.4%
Rotor Outlet		
Axial Velocity	3.2%	6.7%
Relative Angles	1.6°	2.8°
Absolute Angles	1.5°	4.2°
Deviation Angles	2.2°	3.0°
Stator Inlet		
Axial Velocity	3.4%	6.8%
Absolute Angles	2.5°	4.2°
Total Pressure Ratio	0.3%	1.2%
Stator Outlet		
Axial Velocity	1.7%	5.7%
Absolute Angles	0.6°	1.5°
Deviation Angles	1.4°	2.6°

TABLE 3

Comparison of Program Predictions with NASA Task-1
Compressor Measurements at 70% Design Speed

	<u>Average Difference</u>	<u>Maximum Difference</u>
Rotor Inlet		
Axial Velocity	4.8%	7.4%
Relative Angles	1.14°	1.8°
Total Pressure Ratio	1.1%	2.0%
Efficiencies	3.2%	4.8%
Rotor Outlet		
Axial Velocity	4.1%	7.4%
Relative Angles	2.0°	3.2°
Absolute Angles	3.6°	5.5°
Deviation Angles	1.6°	3.5°
Stator Inlet		
Axial Velocity	4.0%	8.8%
Absolute Angles	3.9°	5.5°
Total Pressure Ratio	0.5%	2.1%
Stator Outlet		
Axial Velocity	3.6%	7.4%
Absolute Angles	0.6°	1.1°
Deviation Angles	1.5°	2.4°

TABLE 4

Comparison of Program Predictions with NASA Task-1
Compressor Measurements at 80% Design Speed

	<u>Average Difference</u>	<u>Maximum Difference</u>
Rotor Inlet		
Axial Velocity	7.9%	11.4%
Relative Angles	1.3°	1.5°
Total Pressure Ratio	1.5%	2.5%
Efficiencies	3.1%	7.9%
Rotor Outlet		
Axial Velocity	6.6%	8.9%
Relative Angles	2.9°	5.5°
Absolute Angles	6.8°	8.5°
Deviation Angles	2.4°	4.7°
Stator Inlet		
Axial Velocity	6.9%	13.9%
Absolute Angles	5.6°	7.0°
Total Pressure Ratio	0.9%	3.1%
Stator Outlet		
Axial Velocity	8.0%	20.0%
Absolute Angles	0.8°	1.4°
Deviation Angles	1.5°	2.3°

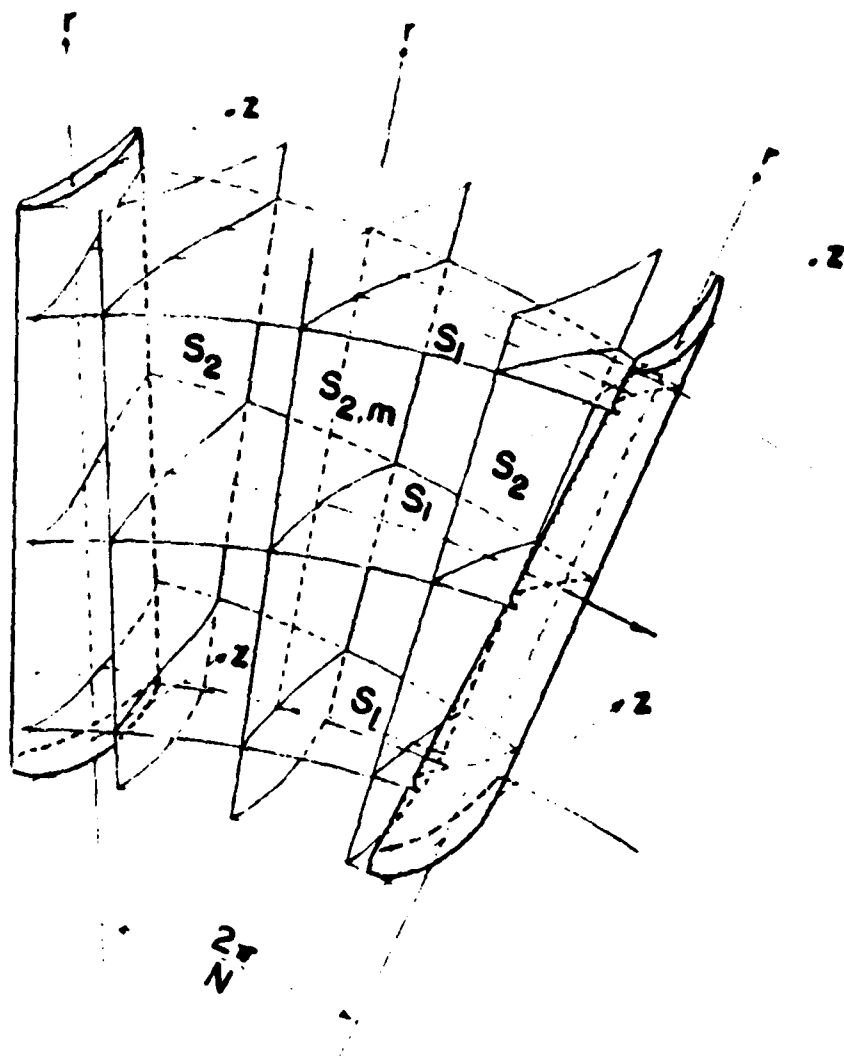


Figure 1. S_1 and S_2 Stream Surfaces

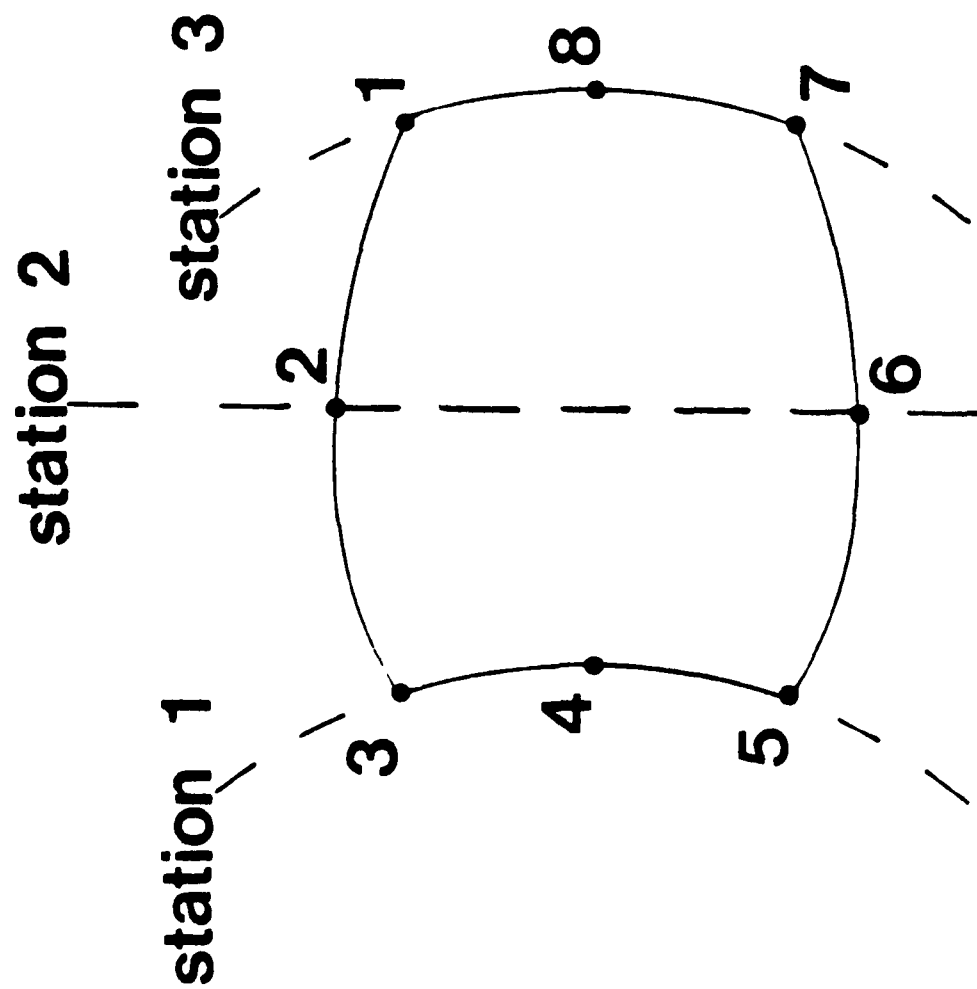


Figure 2. Nomenclature of an Eight-Node Element

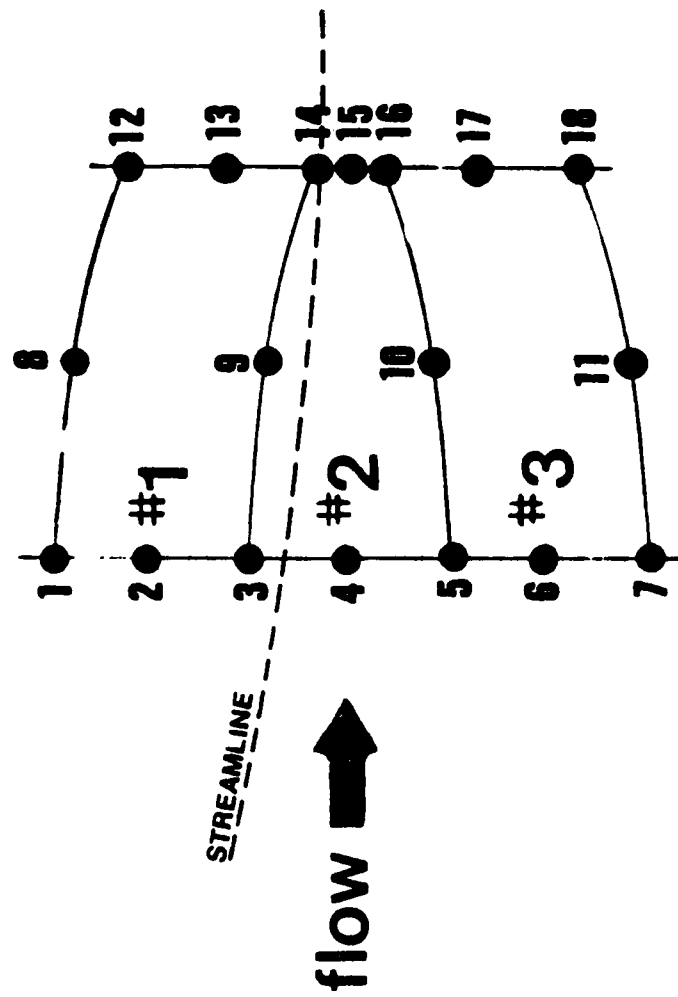


Figure 3. Example of a Three Element Mesh

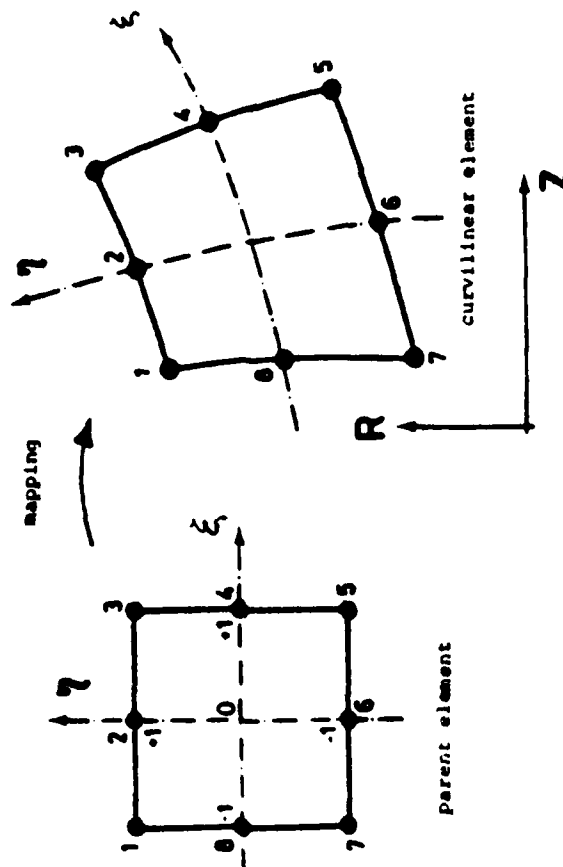


Figure 4. Mapping Relationships from the ξ, η to the z, r Plane

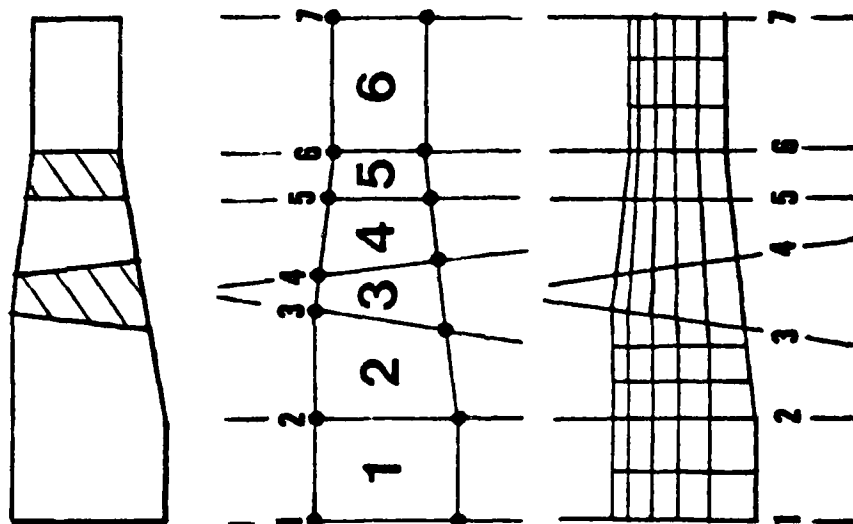


Figure 5. Mesh Generation

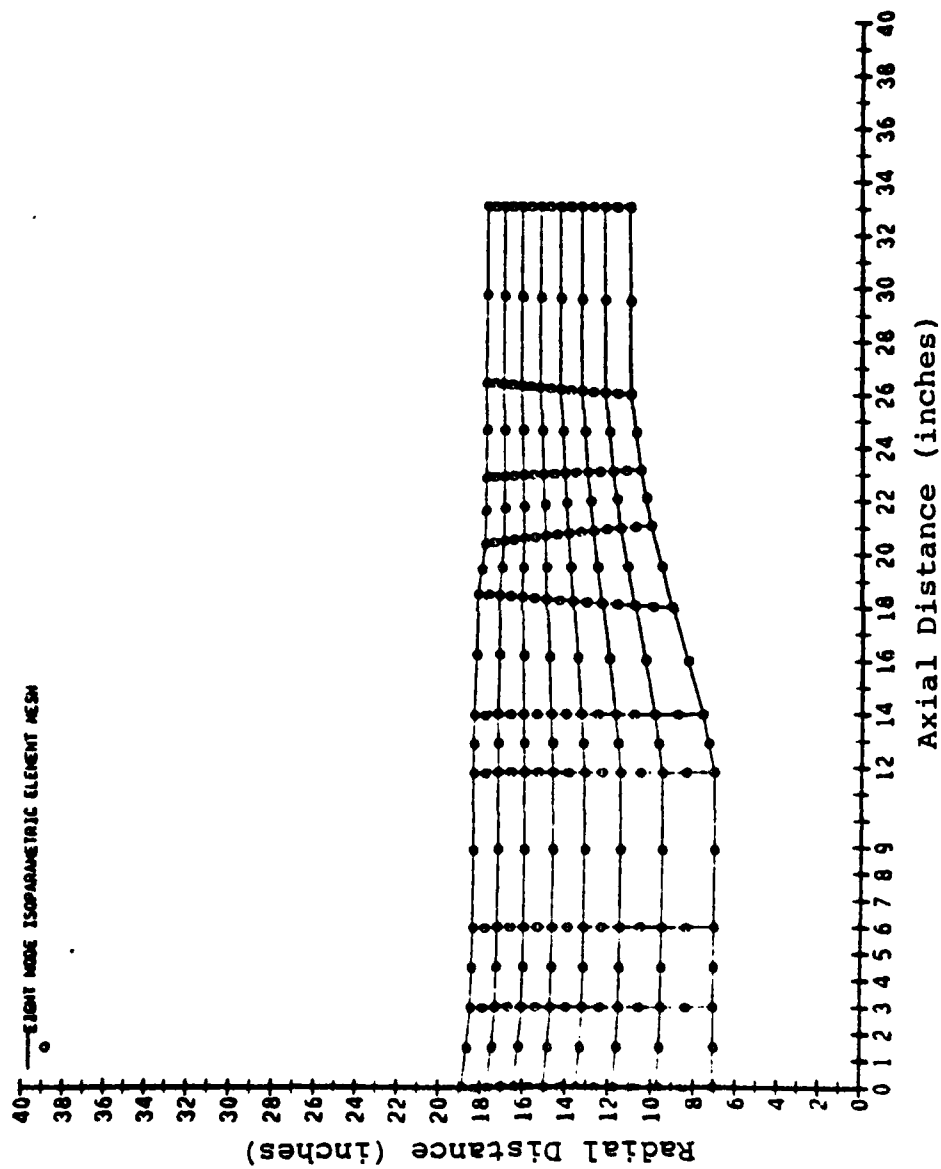


Figure 6. MESHGEN Generated Plot of the 222 Node Mesh Used in the Calculations of the Meridional Through-Flow

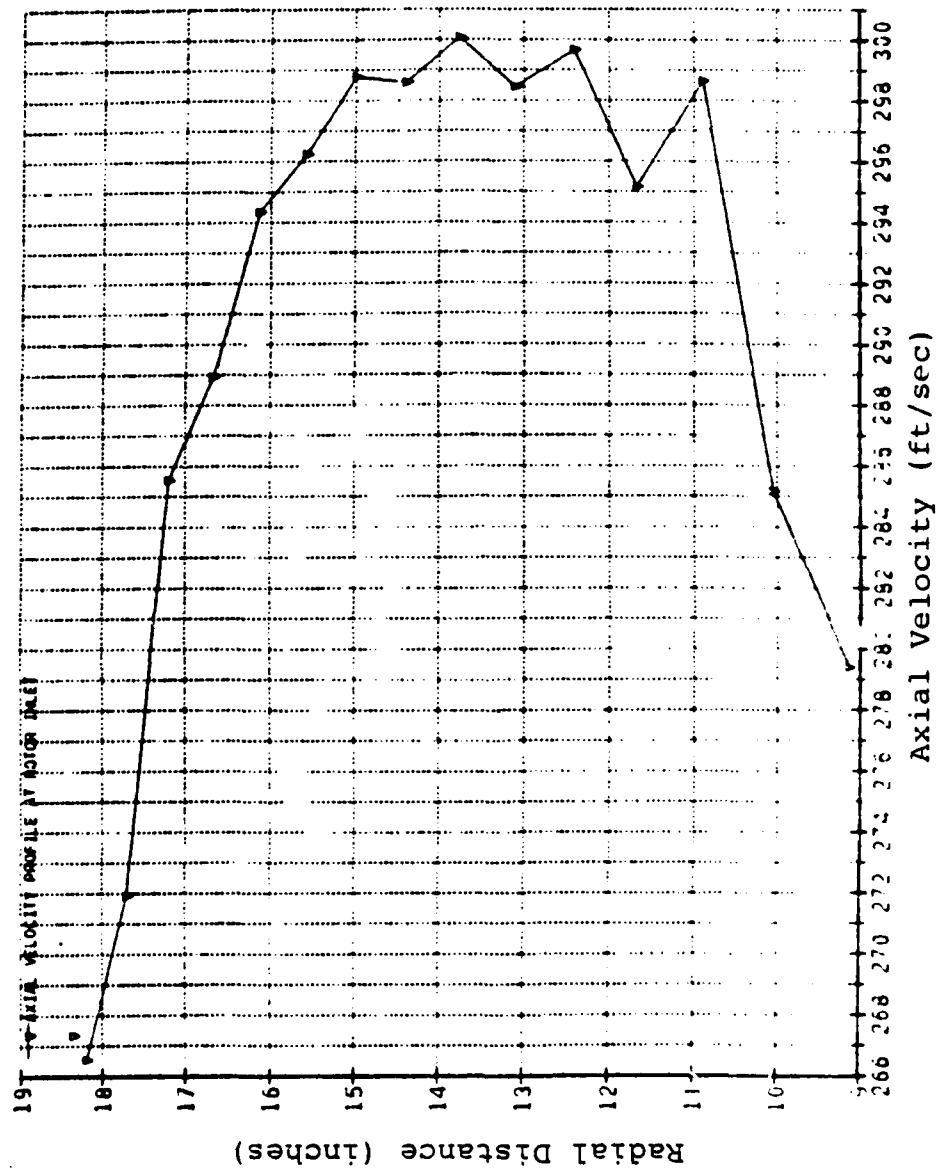


Figure 7. A TURBO Generated Tektonix 618 Plot of the Rotor inlet Axial Velocity at 50% Design Speed

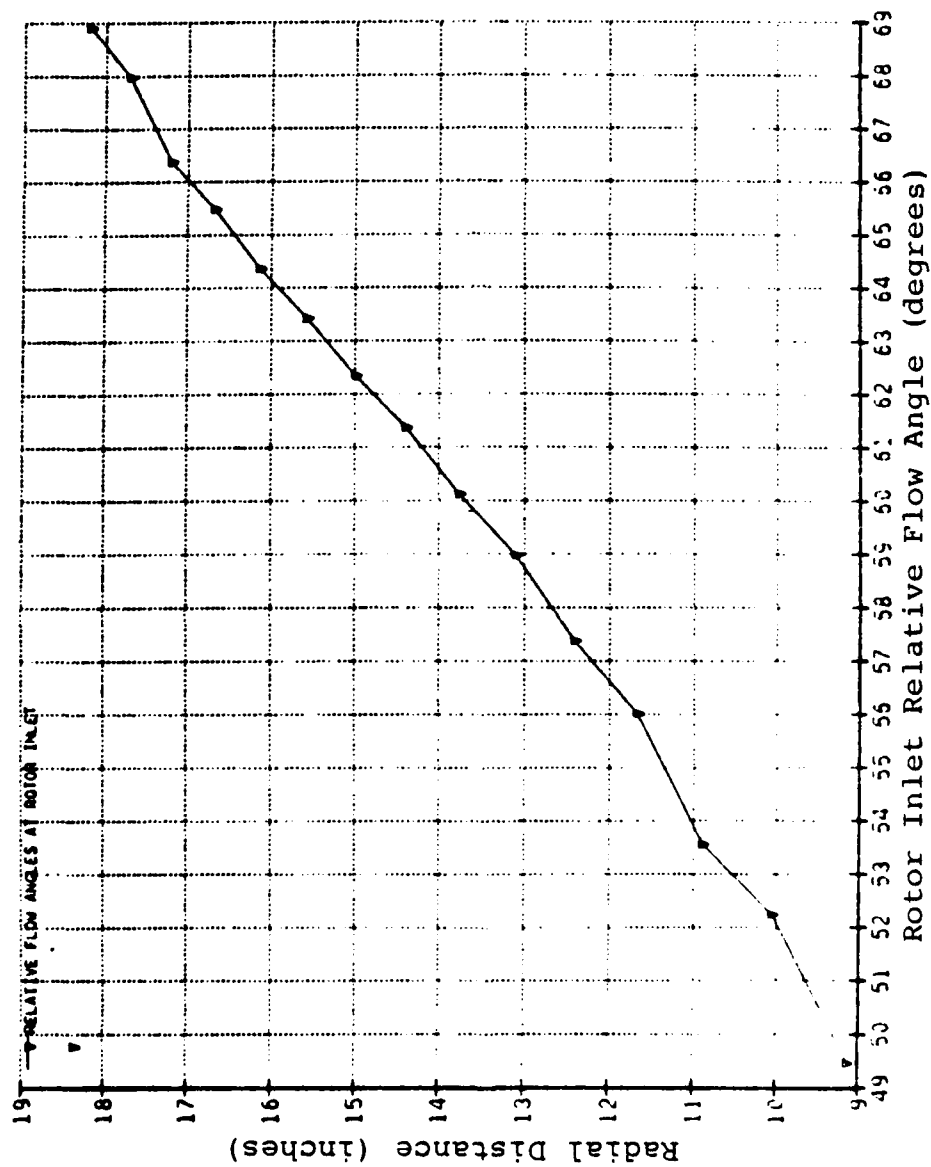


Figure 8. A TURBO Generated Tektonix 618 Plot of the Rotor Inlet Relative Flow Angles

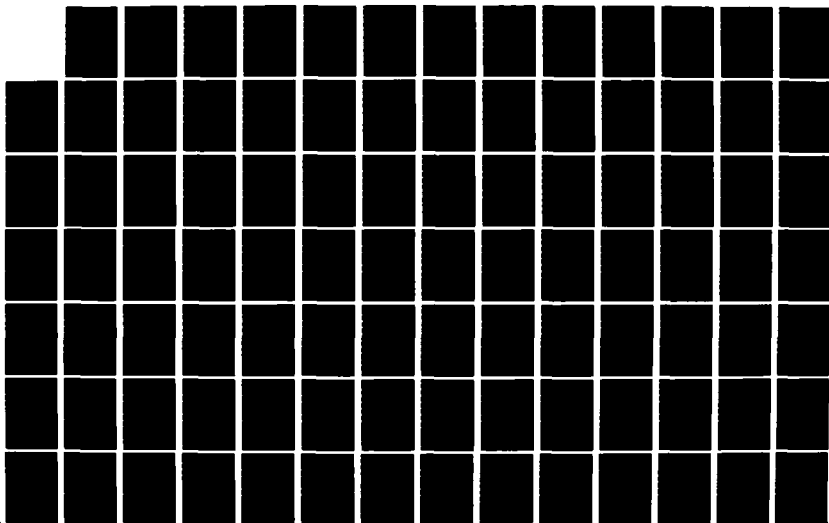
AD-A124 987

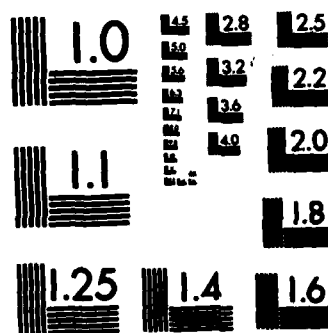
FINITE ELEMENT PROGRAM FOR CALCULATING FLOWS IN
TURBOMACHINES WITH RESULTS FOR NASA TASK-1 COMPRESSOR
(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA J A FERGUSON
OCT 82 F/G 28/4

2/3

UNCLASSIFIED

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

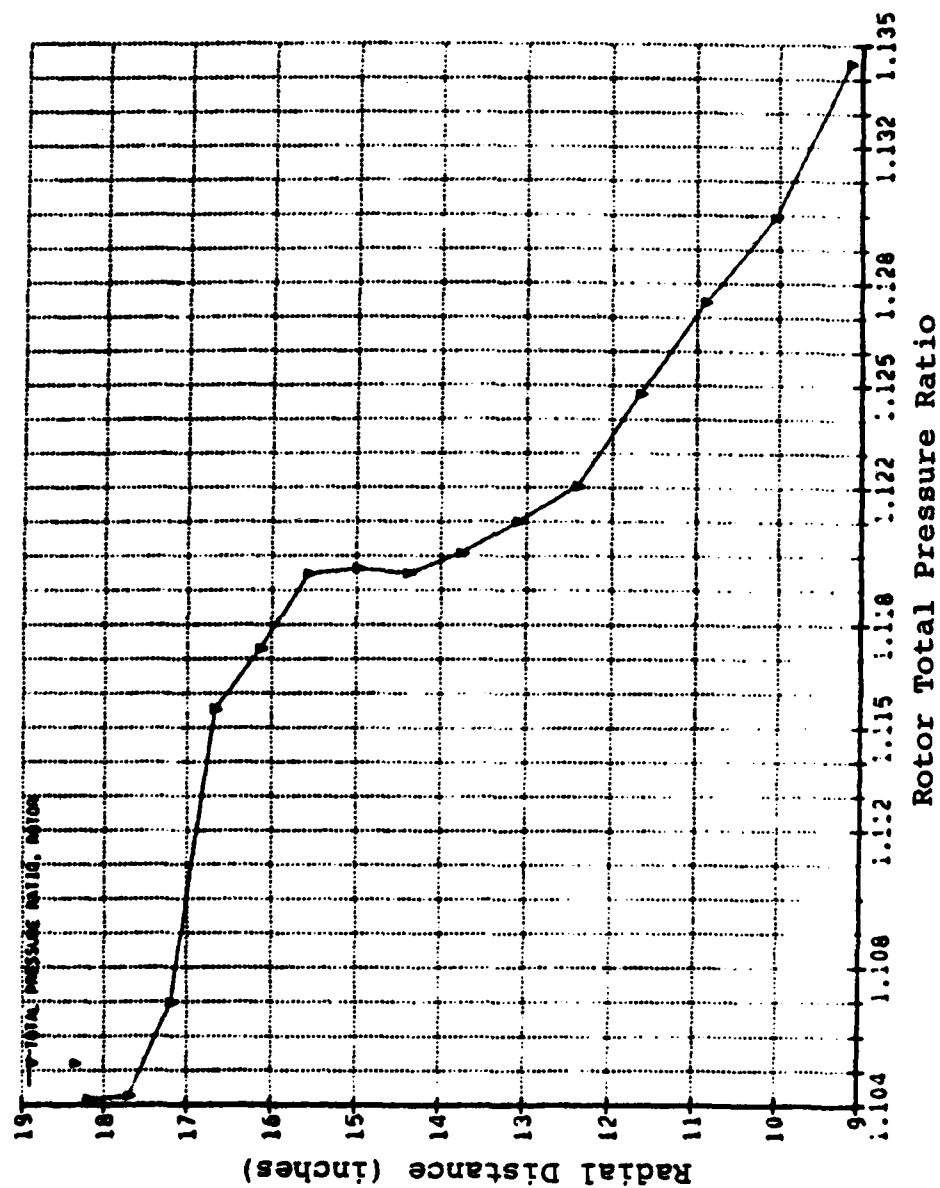


Figure 9. A TURBO Generated Tektonix 618 Plot of the Total Pressure Ratio of the Rotor vs Inlet Radius

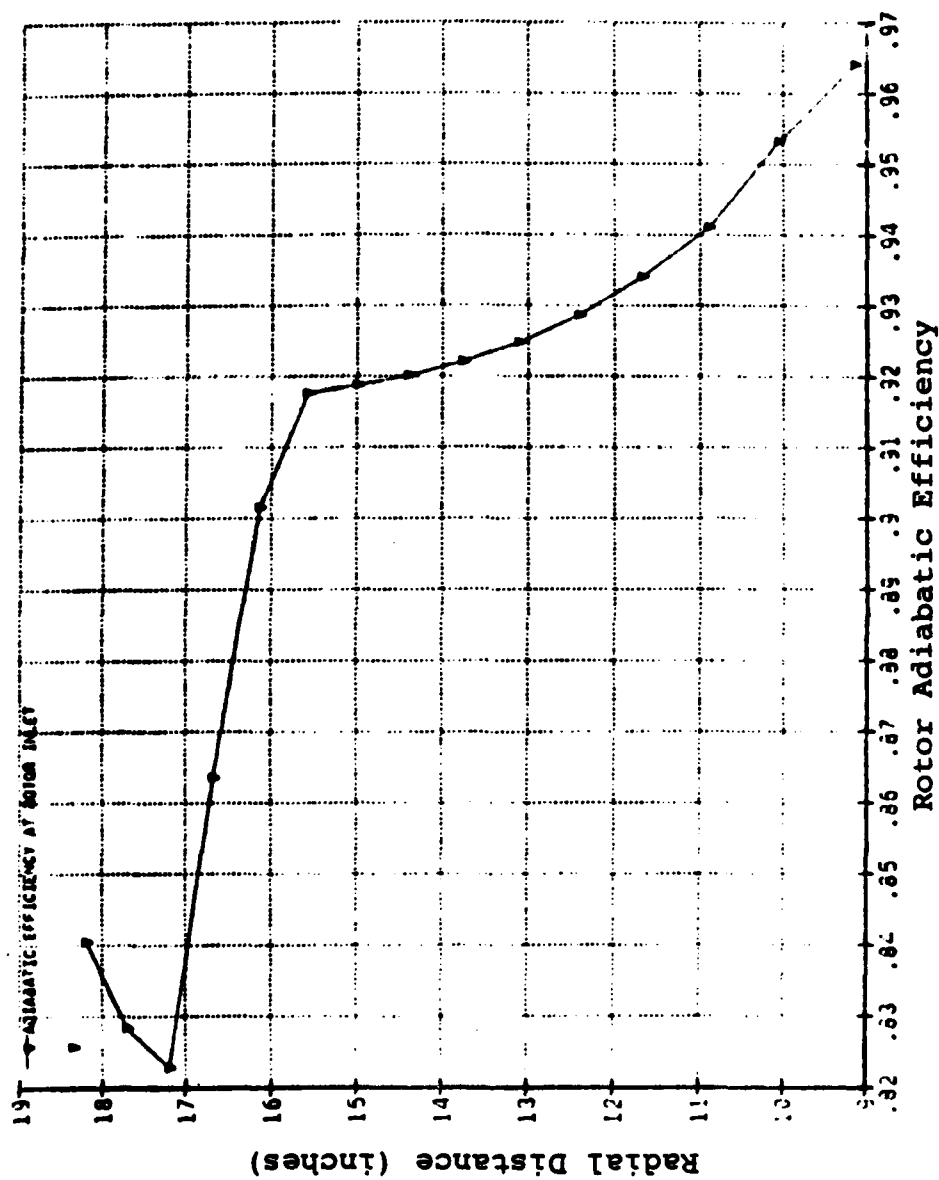


Figure 10. A TURBO Generated Tektonix 618 Plot of the Adiabatic Efficiency of the Rotor vs Inlet Radius

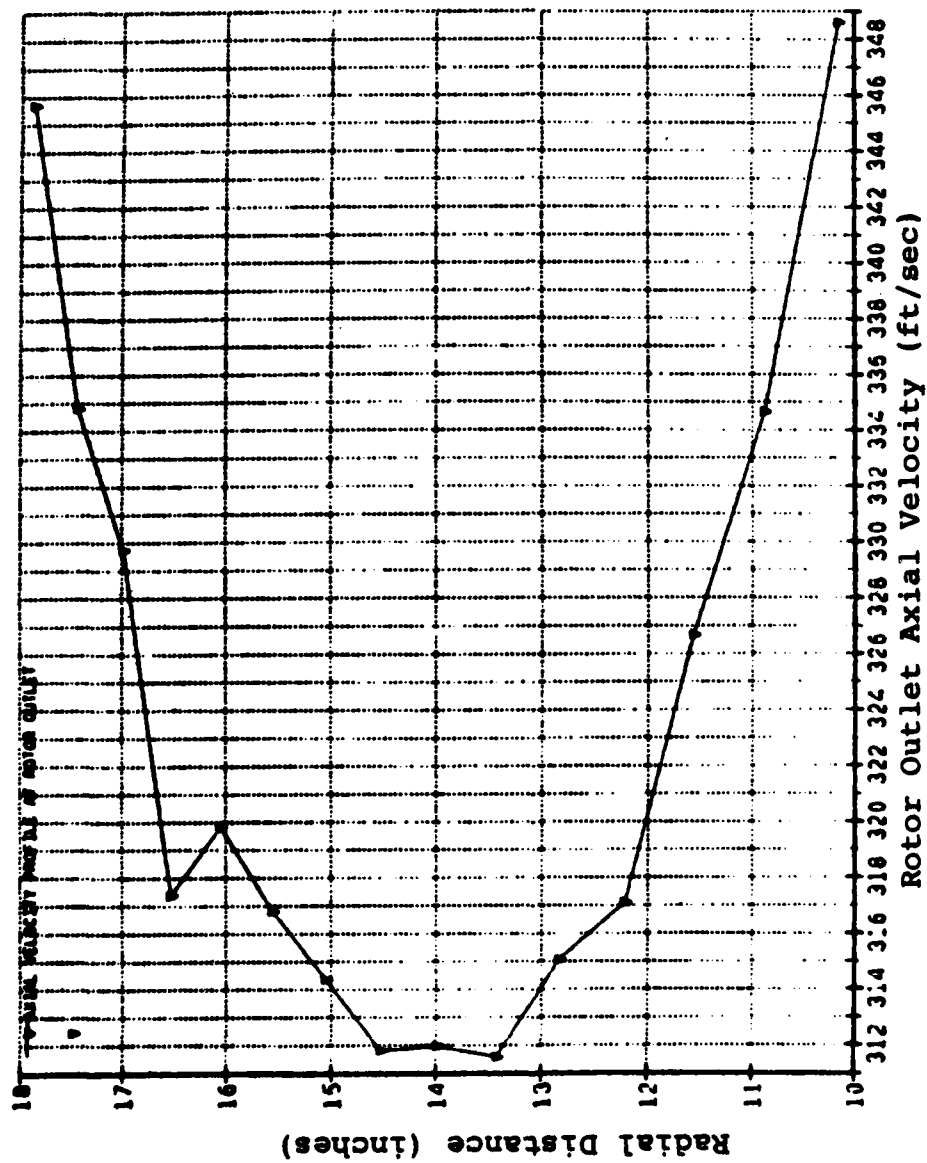


Figure 11. A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Axial Velocity

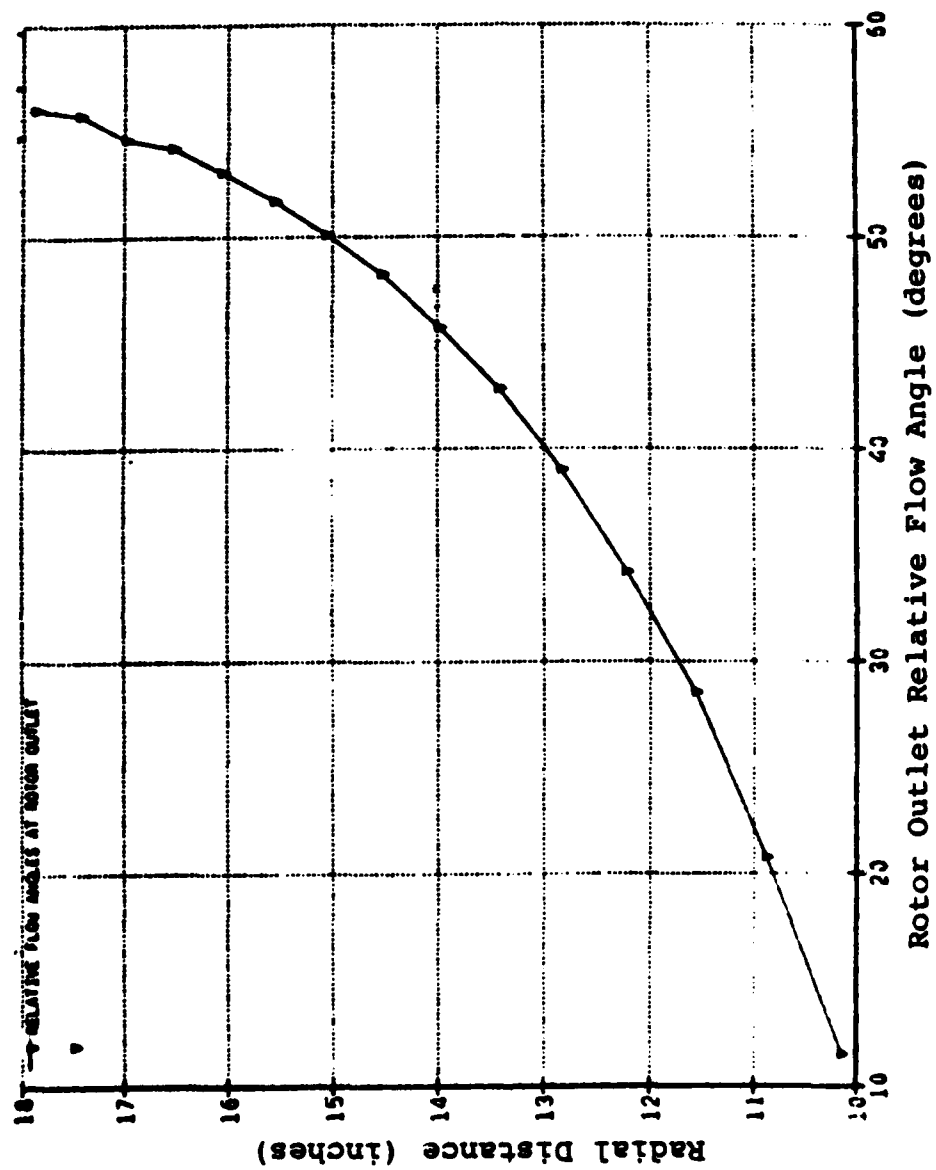


Figure 12. A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Relative Flow Angles

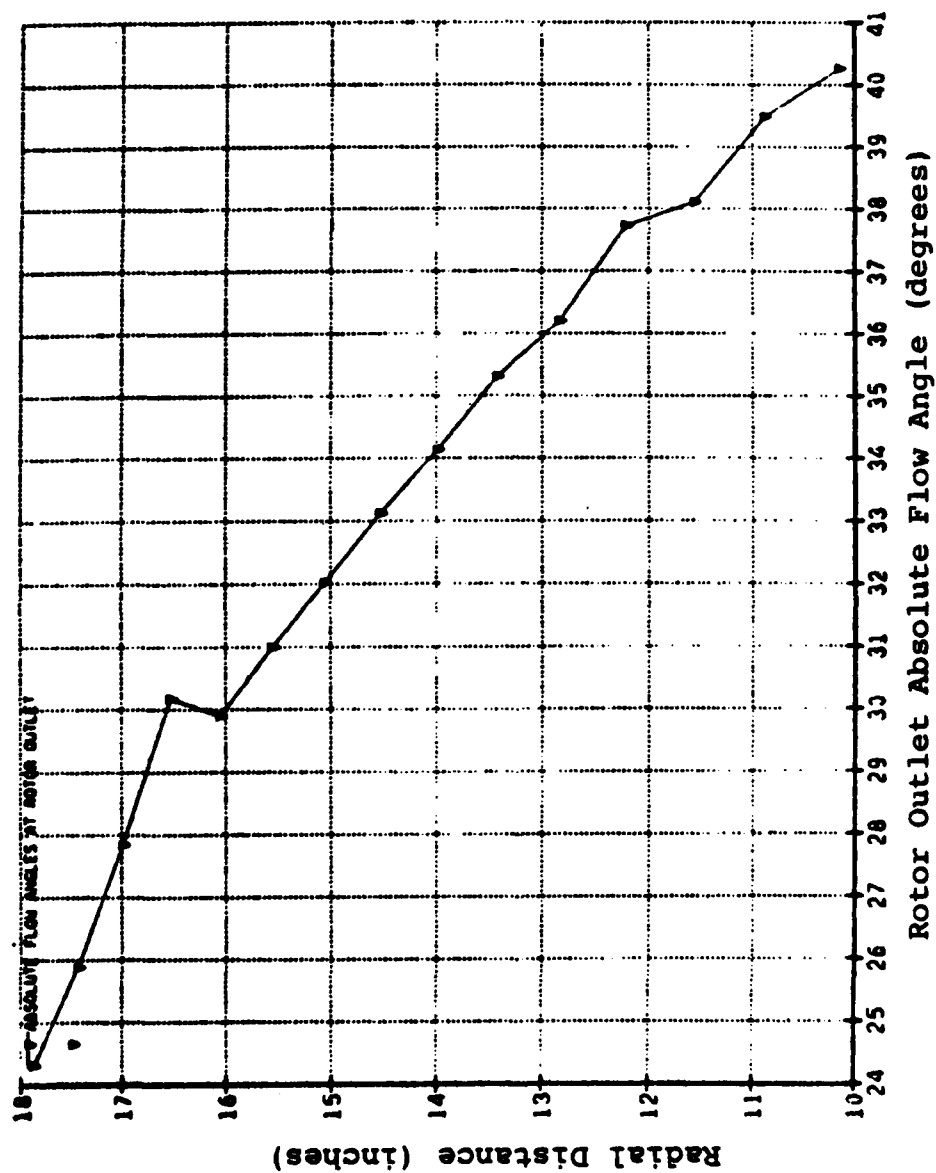


Figure 13. A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Absolute Flow Angles

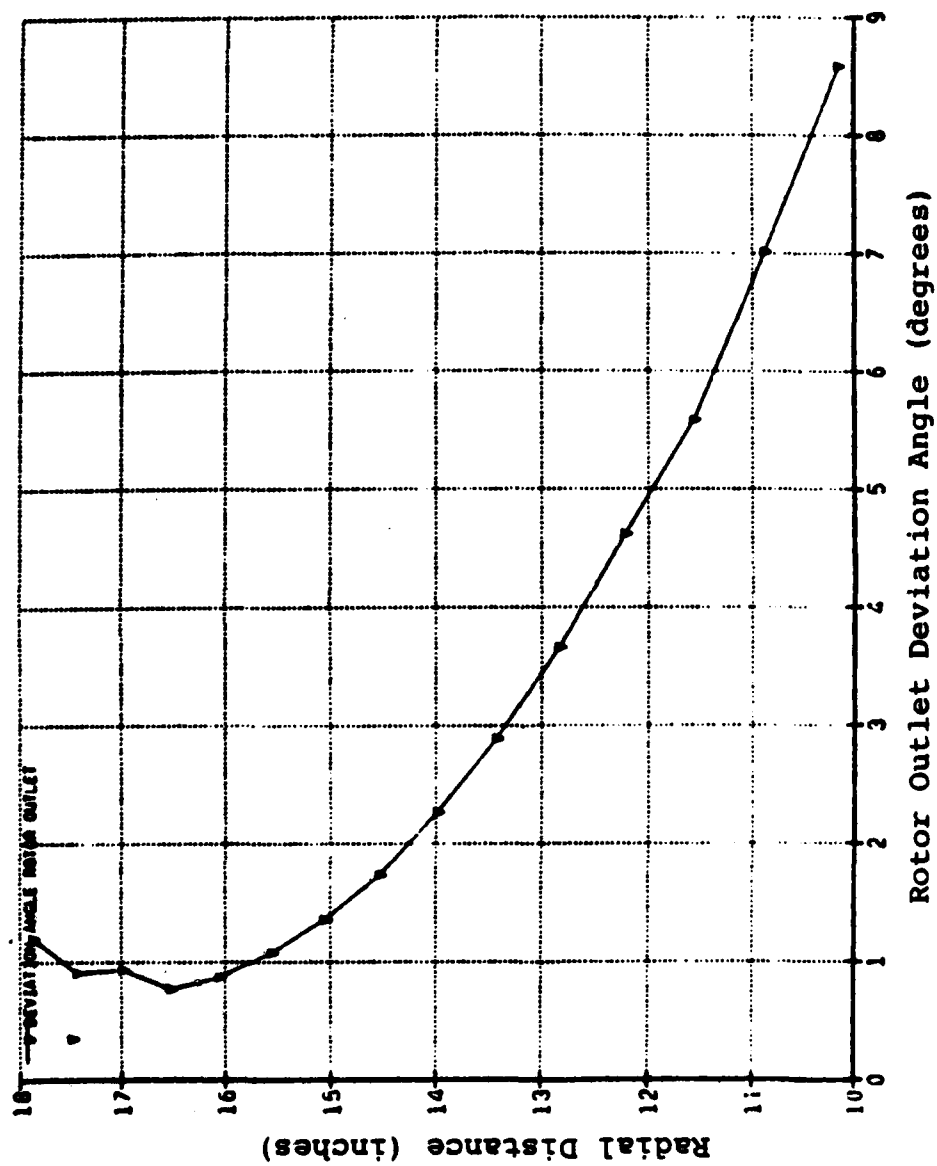


Figure 14. A TURBO Generated Tektonix 618 Plot of the Rotor Outlet Deviation Flow Angles

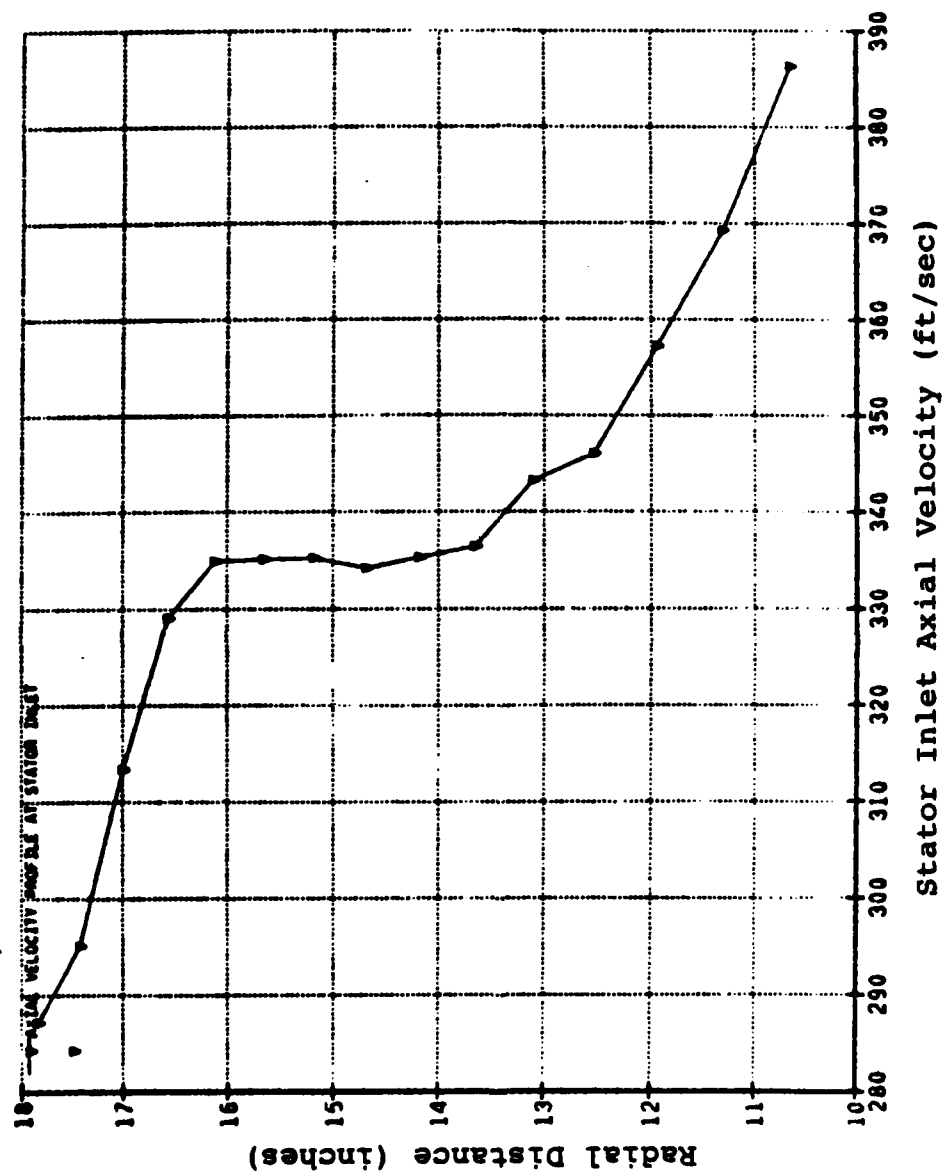


Figure 15. A TURBO Generated Tektonix 618 Plot of the Stator Inlet Axial Velocity

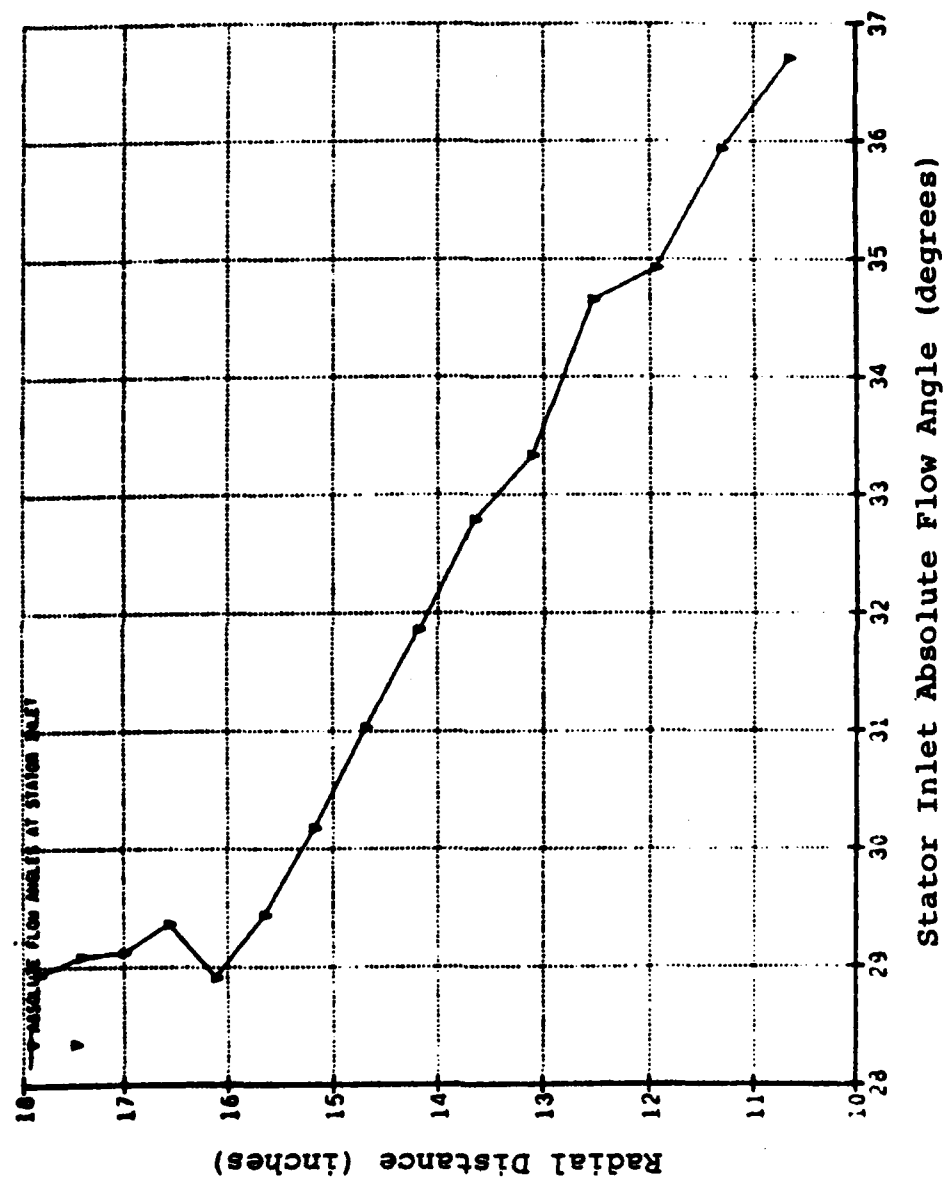


Figure 16. A TURBO Generated Tektonix 618 Plot of the Stator Inlet Absolute Flow Angles

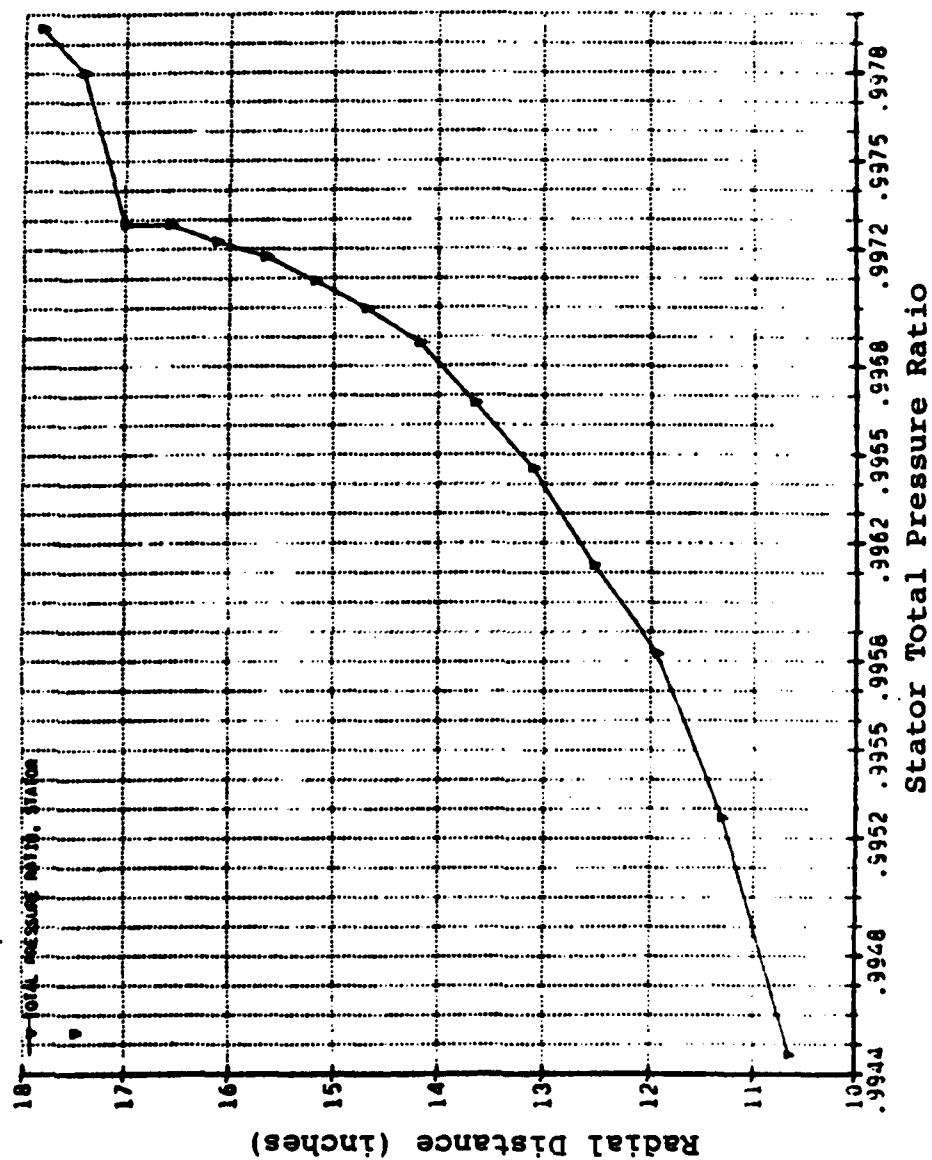


Figure 17. A TURBO Generated Tektonix 618 Plot of the Stator Total Pressure Ratio vs Inlet Radius

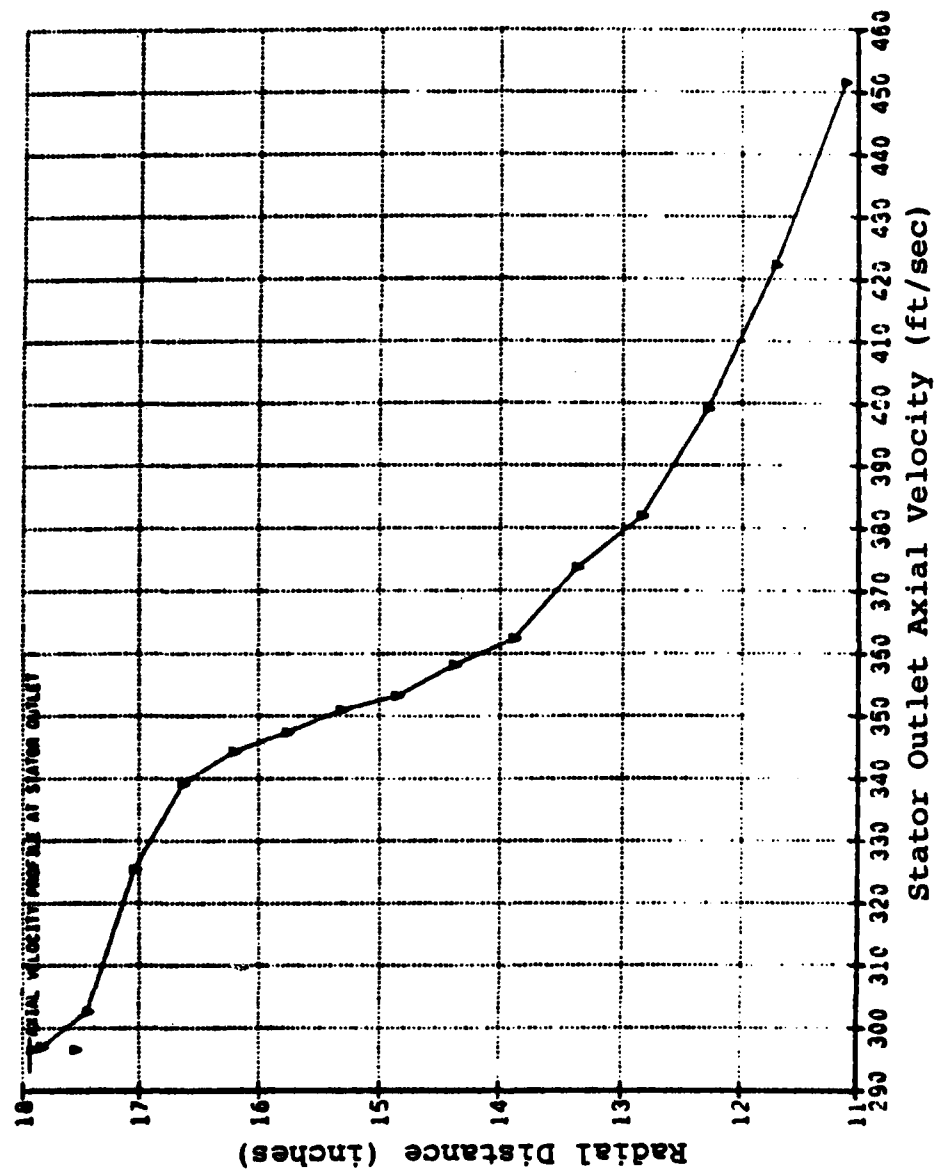


Figure 18. A TURBO Generated Tektonix 618 Plot of the Stator Outlet Axial Velocity

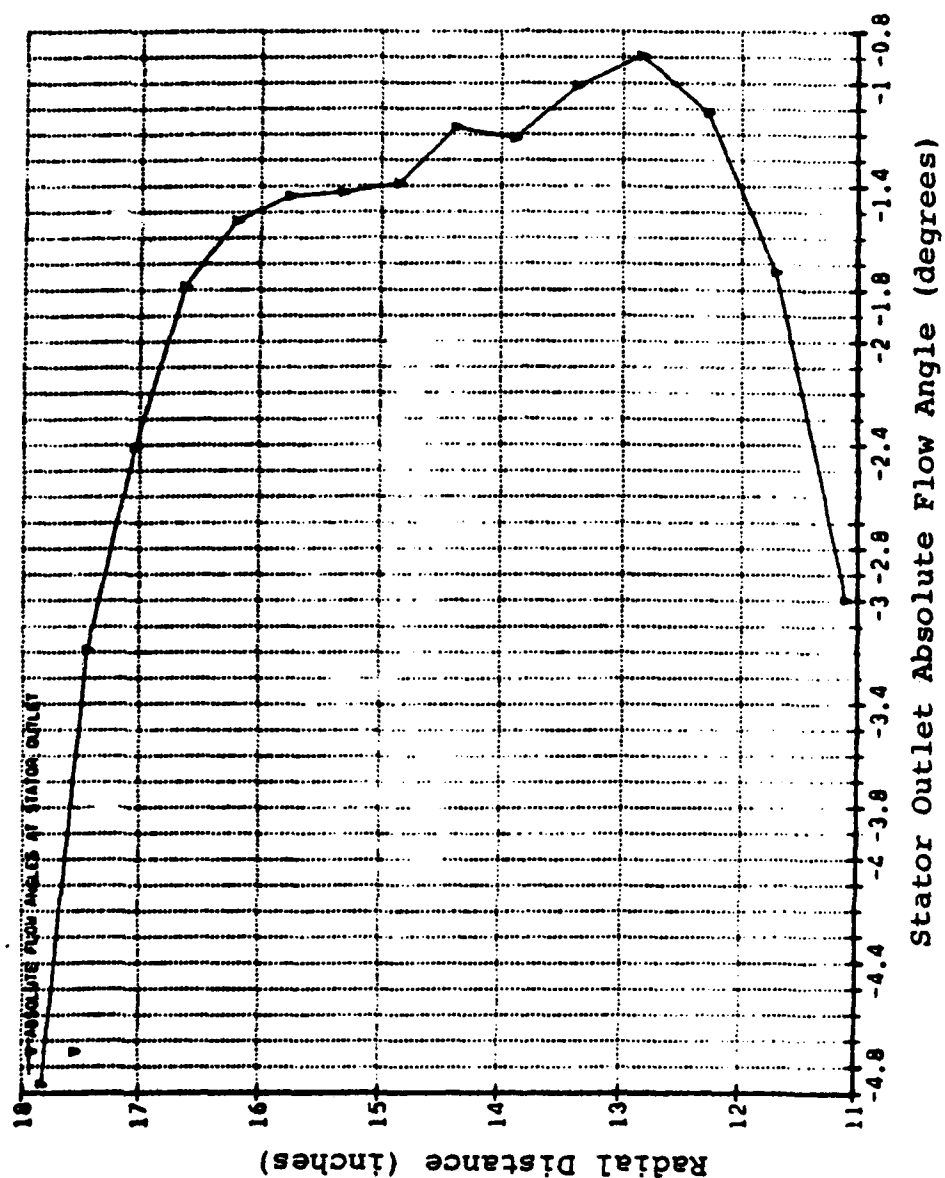


Figure 19. A TURBO Generated Tektonix 618 Plot of the Stator Outlet Absolute Flow Angles

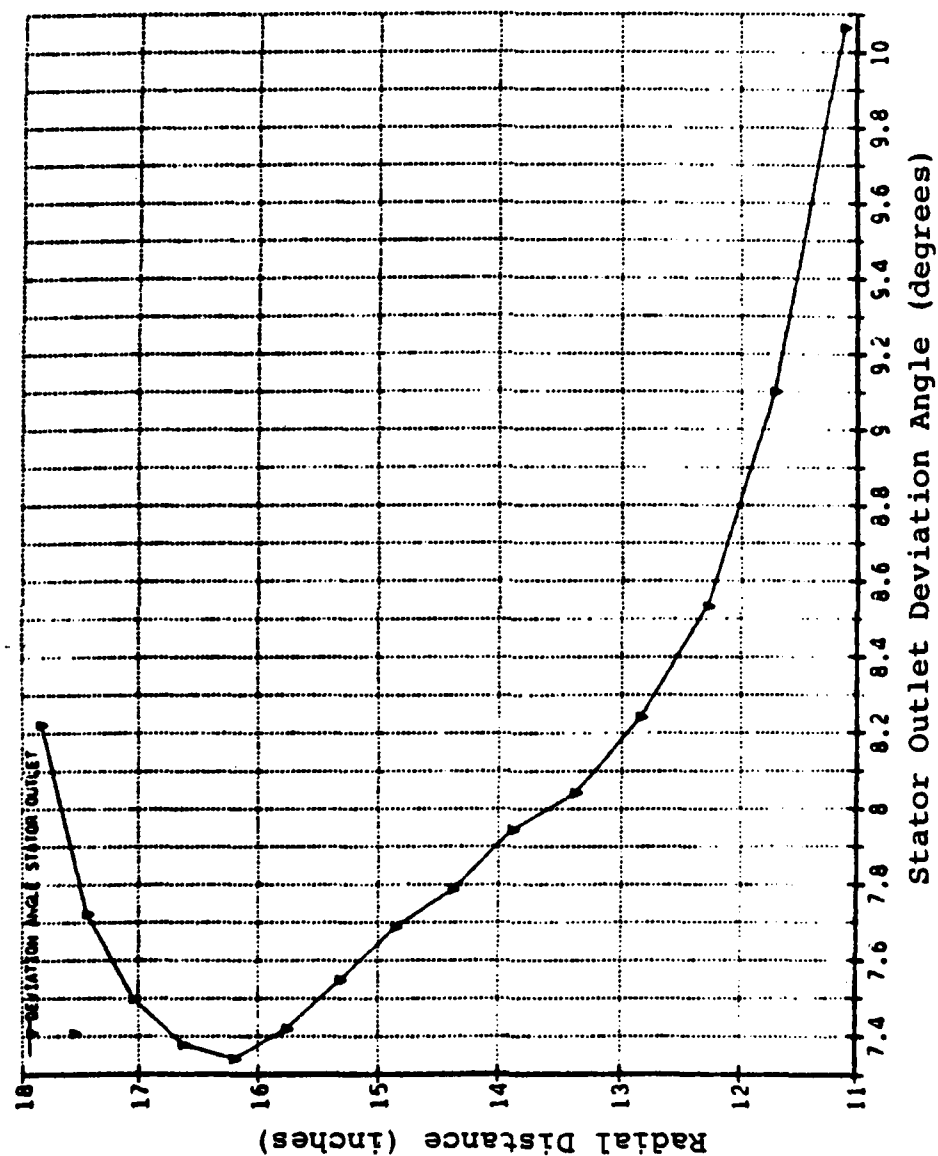


Figure 20. A TURBO Generated Tektonix 618 Plot of the Stator Outlet Deviation Flow Angles

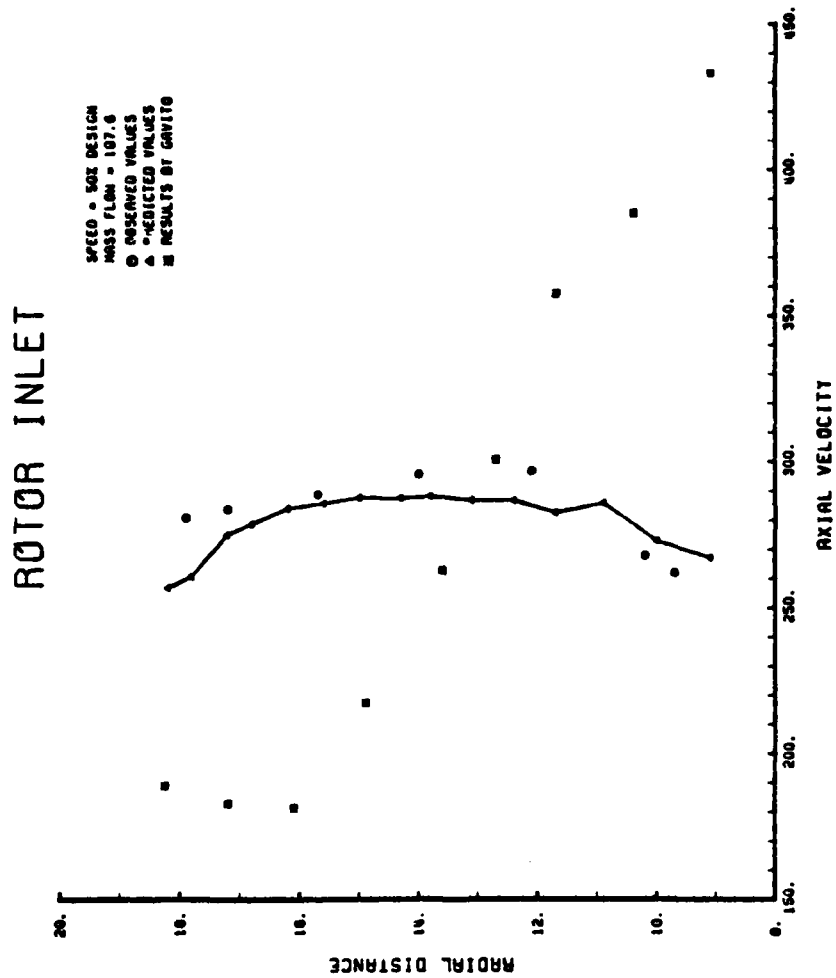


Figure 21. Comparison of Predictions to Gavito and Observations for Rotor Inlet Axial Velocity, 50% Design

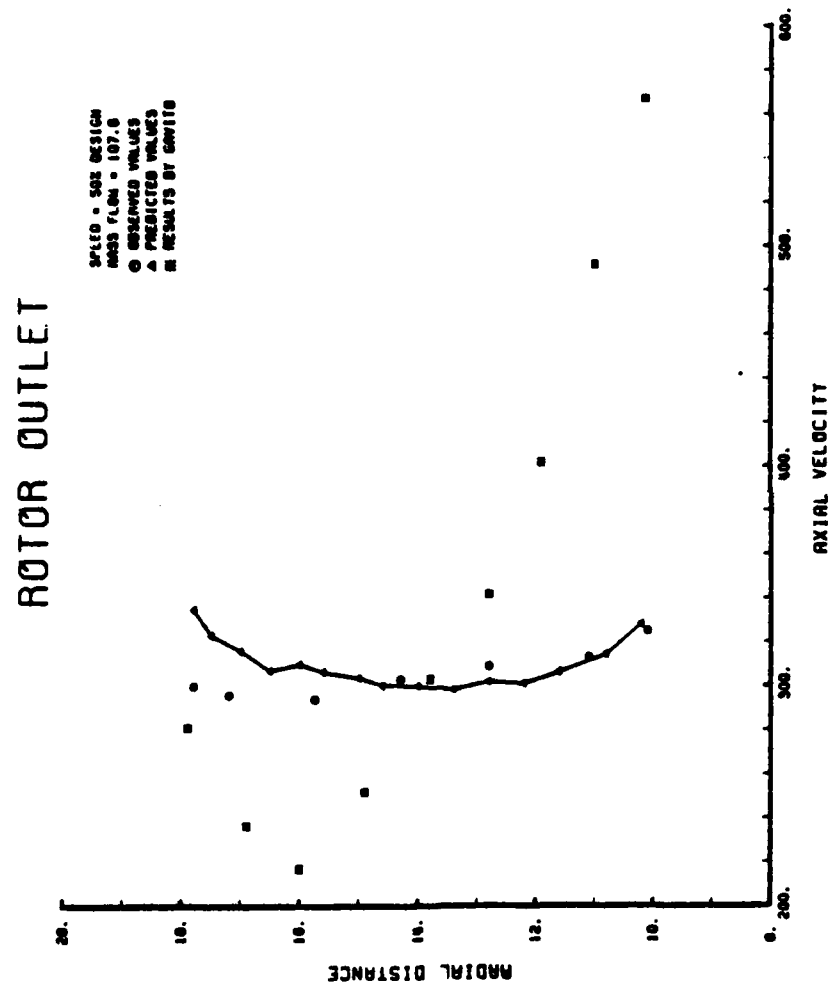


Figure 22. Comparison of Predictions to Gavito and Observations for Rotor Outlet Axial Velocity, 50% Design

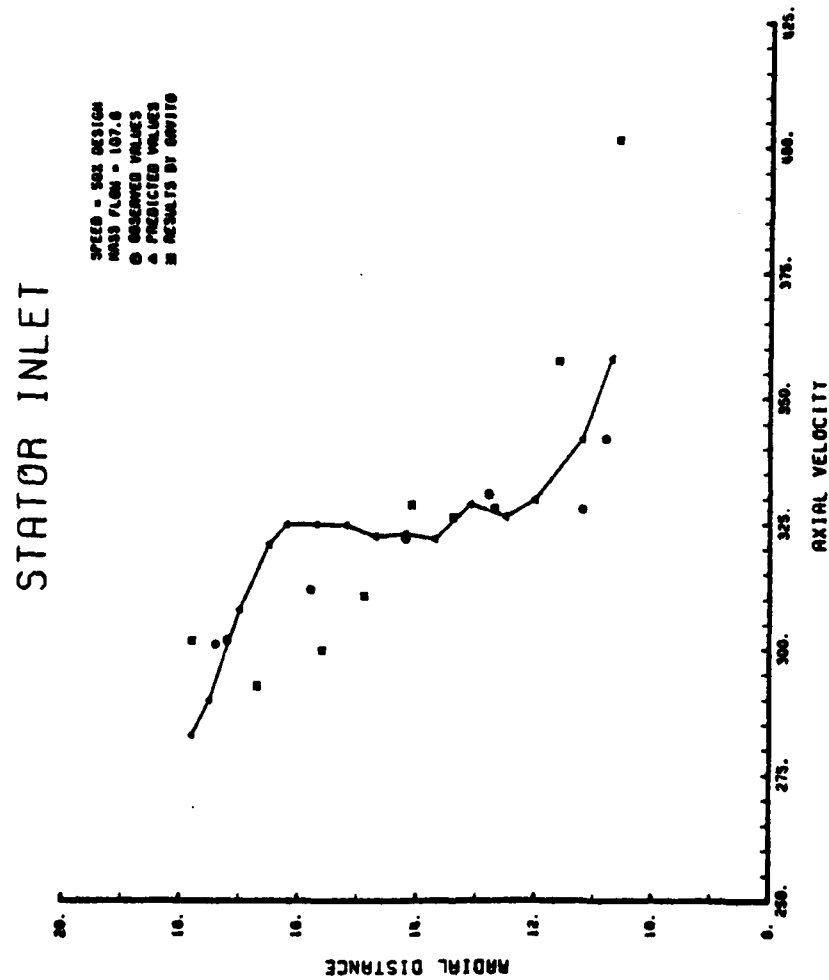


Figure 23. Comparison of Predictions to Gavito and Observations for Stator Inlet Axial Velocity, 50% Design

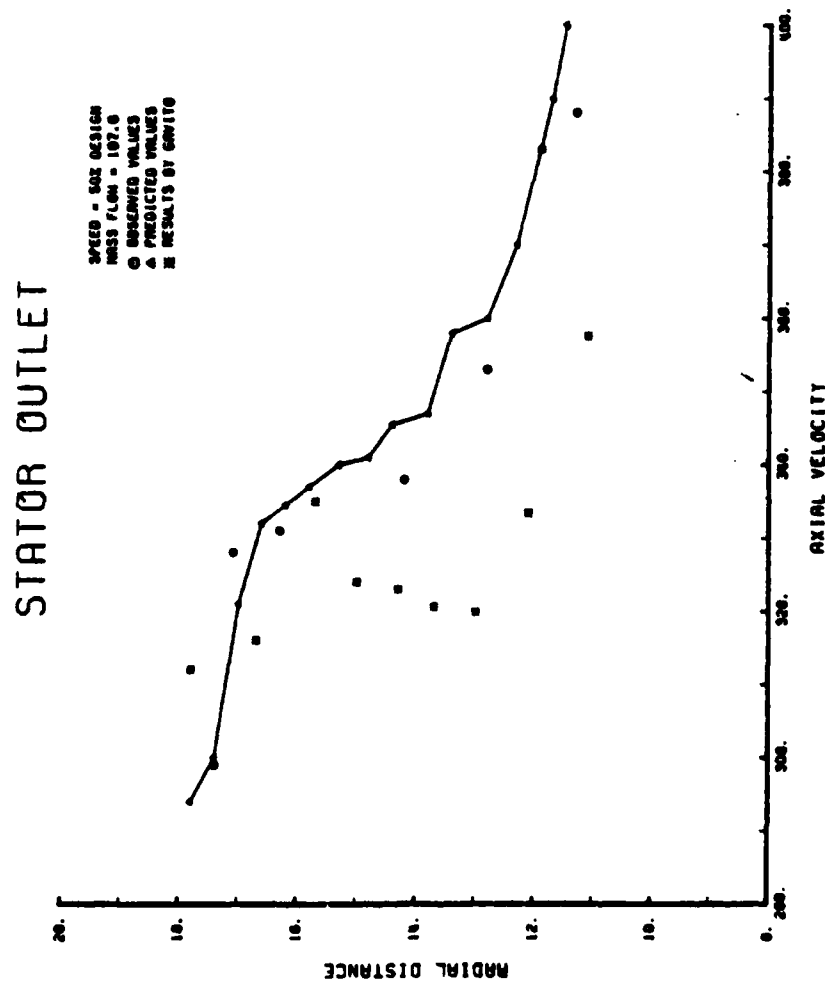


Figure 24. Comparison of Predictions to Gavito and Observations for Stator Outlet Axial Velocity, 50% Design

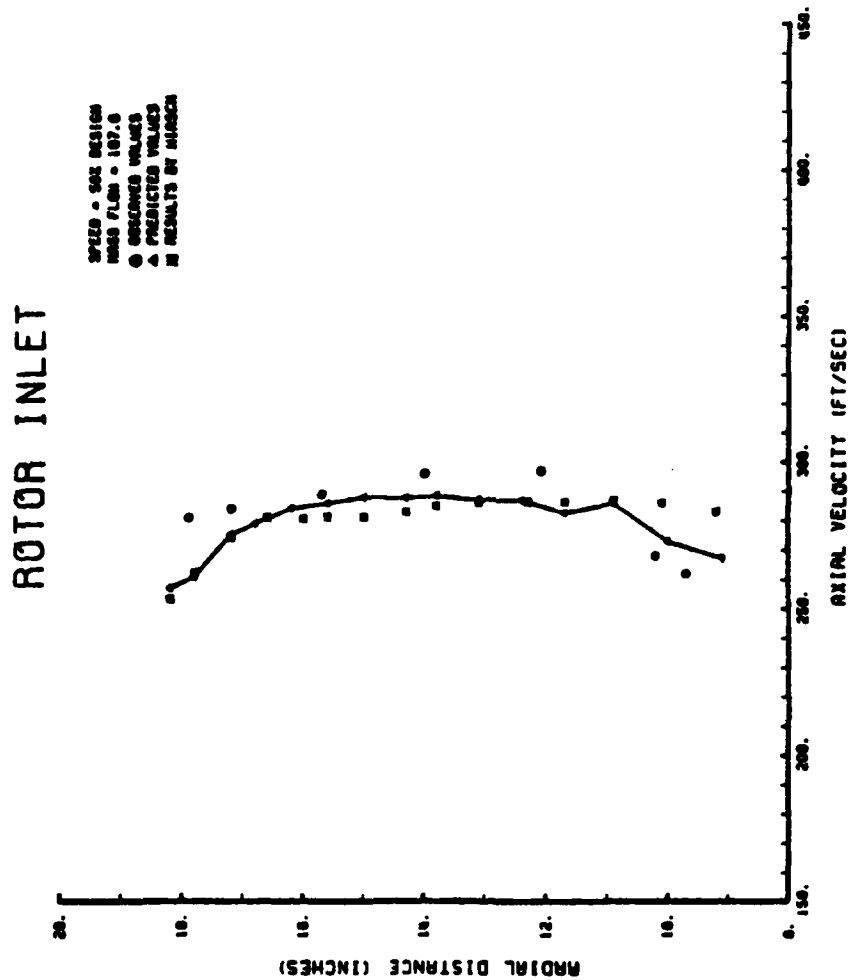


Figure 25. Comparison of Predictions to Hirsch and Observations for Rotor Inlet Axial Velocity, 50% Design

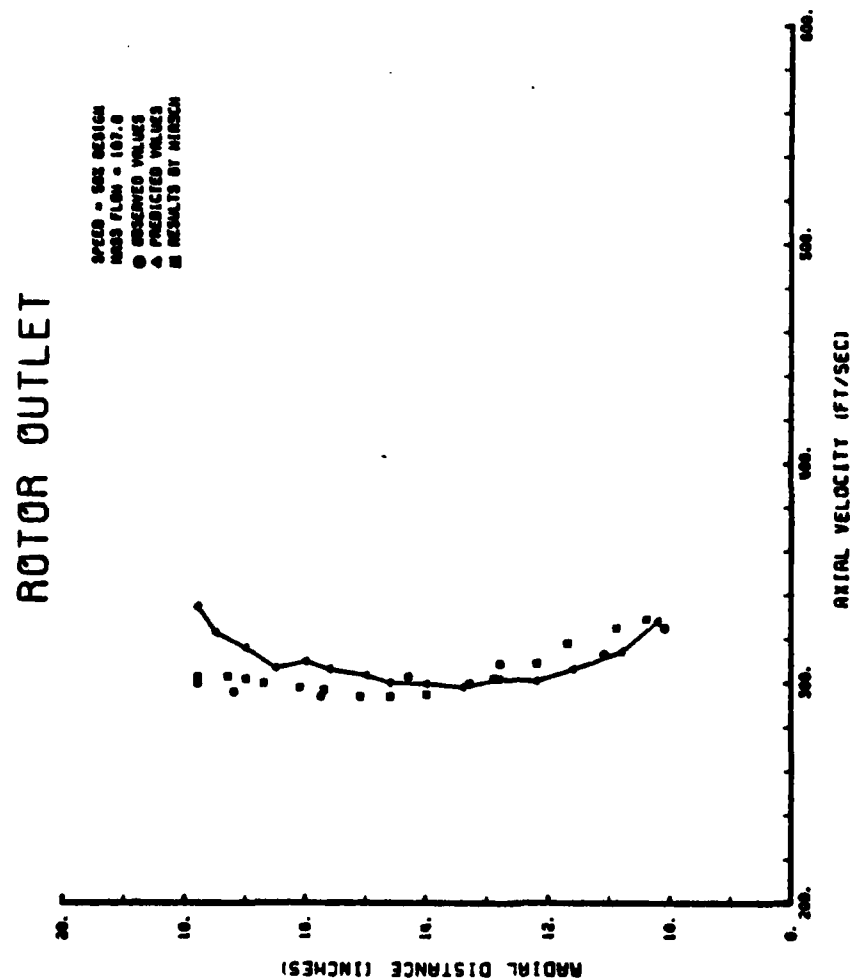


Figure 26. Comparison of Predictions to Hirsch and Observations for Rotor Outlet Axial Velocity, 50% Design

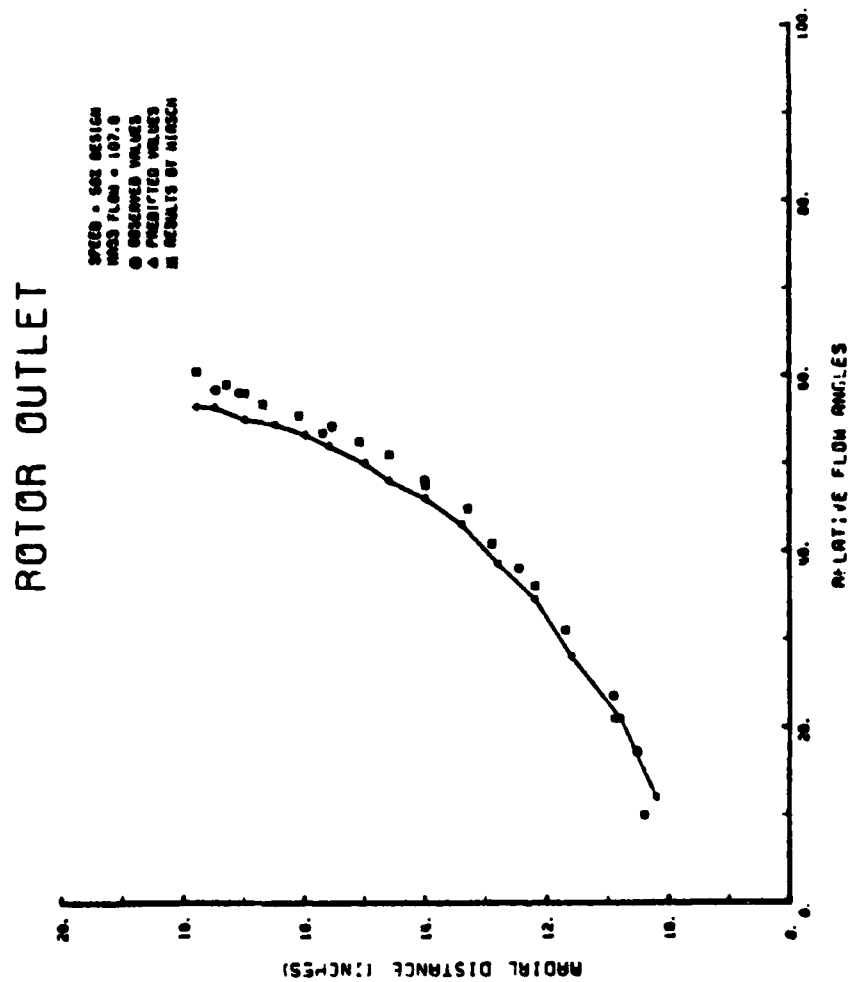


Figure 27. Comparison of Predictions to Hirsch and Observations for Rotor Outlet Relative Flow Angles, 50% Design

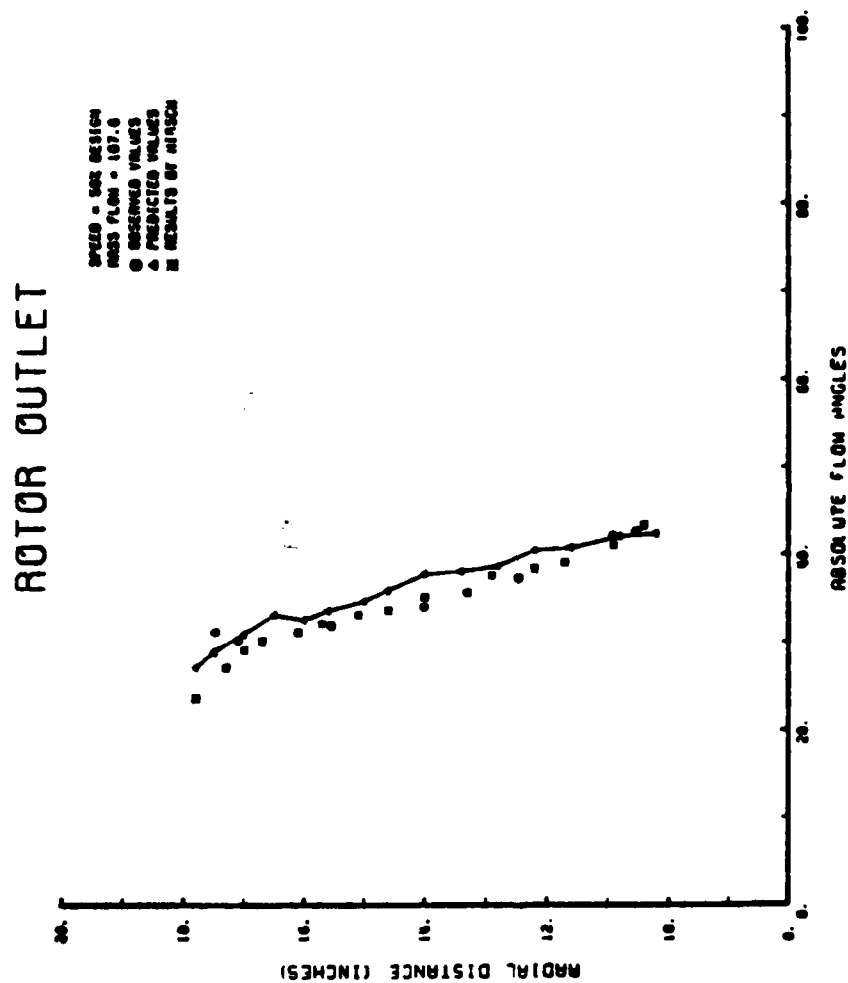


Figure 28. Comparison of Predictions to Hirsch and Observations for Rotor Outlet Absolute Flow Angles, 50% Design

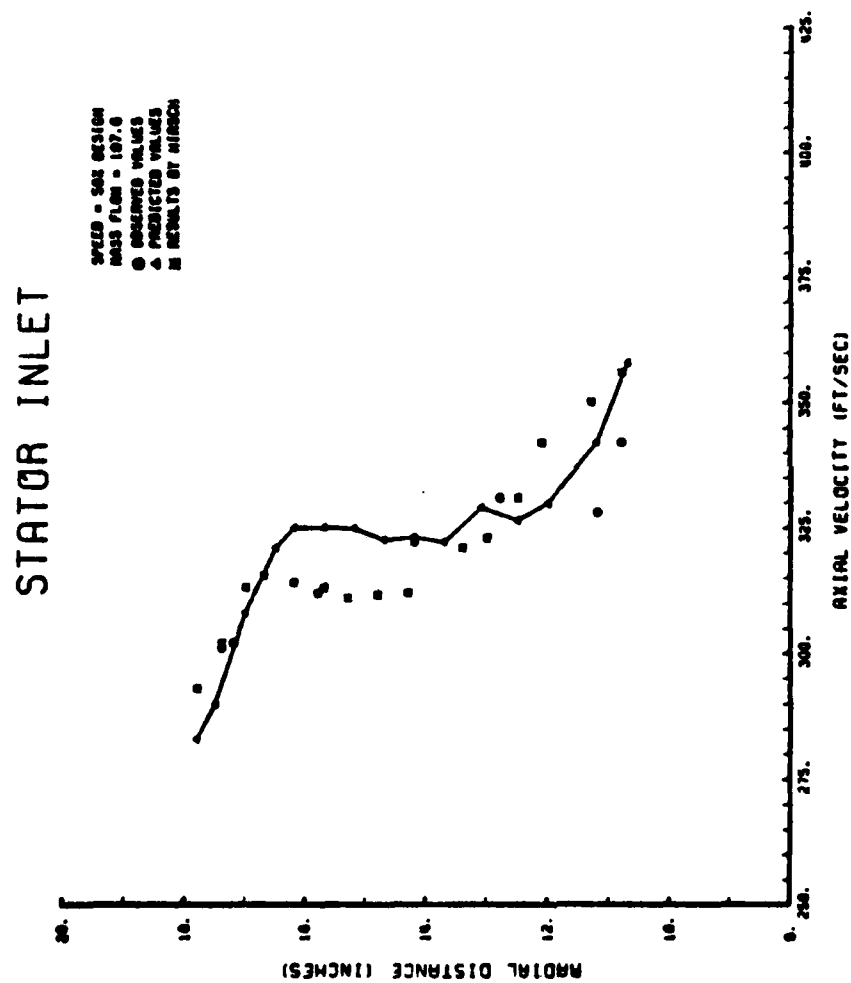


Figure 29. Comparison of Predictions to Hirsch and Observations for Stator Inlet Axial Velocity, 50% Design

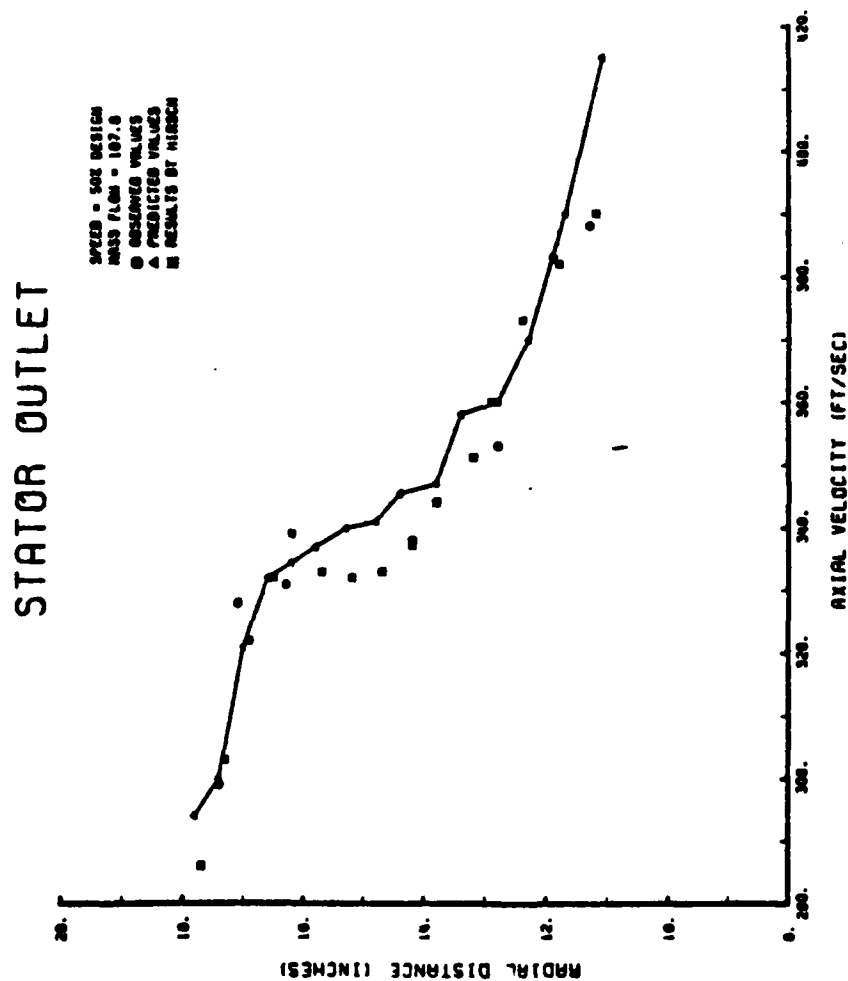


Figure 30. Comparison of Predictions to Hirsch and Observations for Stator Outlet Axial Velocity, 50% Design

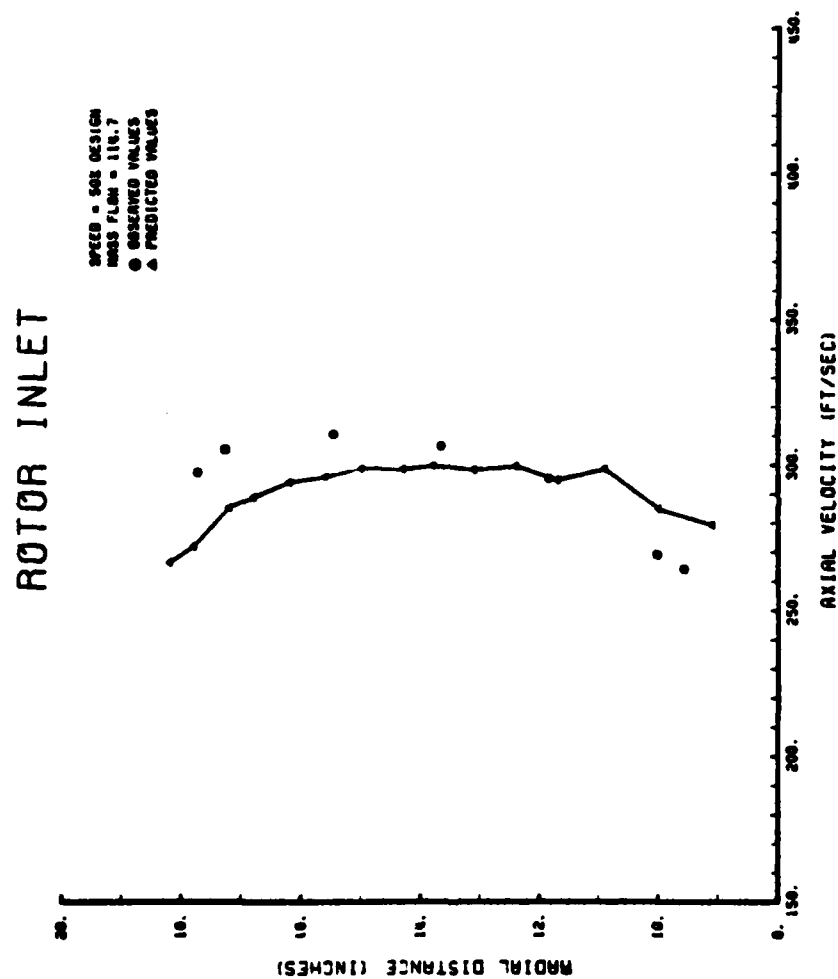


Figure 31. Axial Velocity at the Rotor Inlet, 50% Design

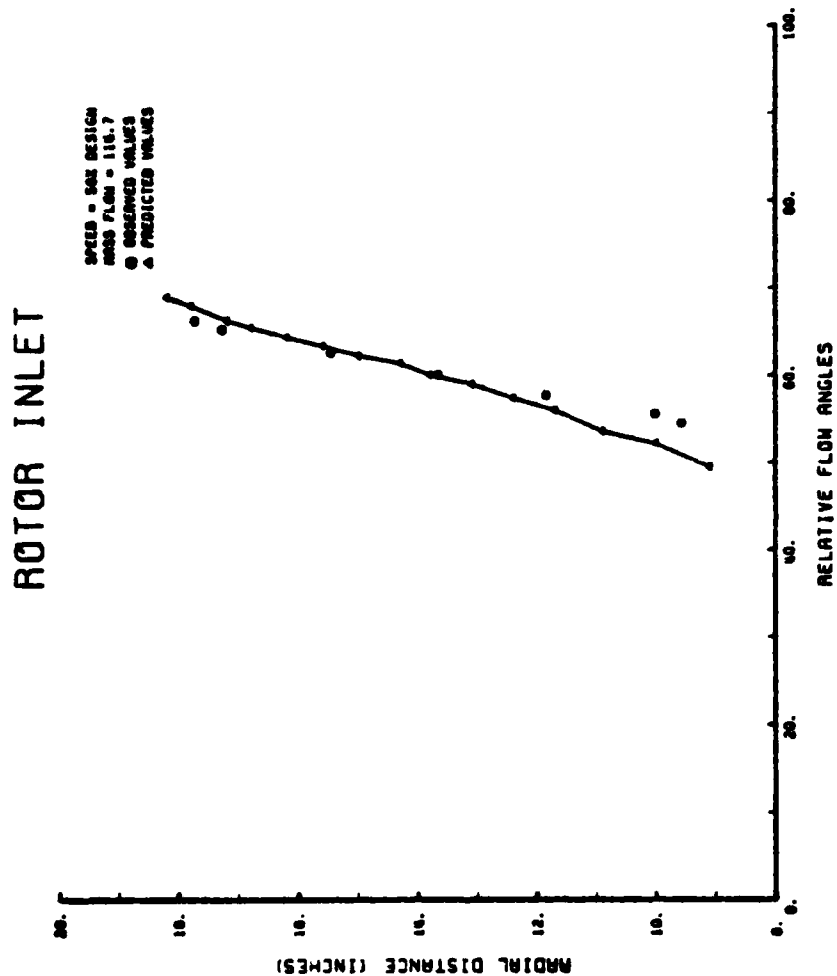


Figure 32. Relative Flow Angles at Rotor Inlet, 50% Design

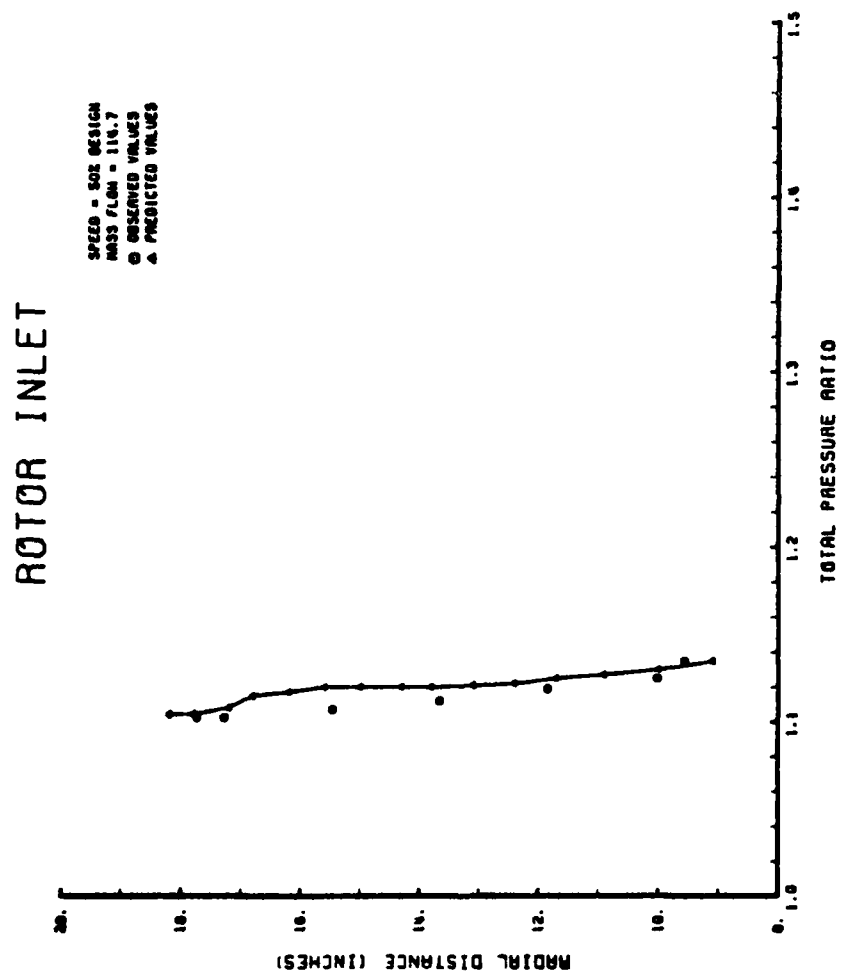


Figure 33. Total Pressure Ratio of the Rotor, 50% Design

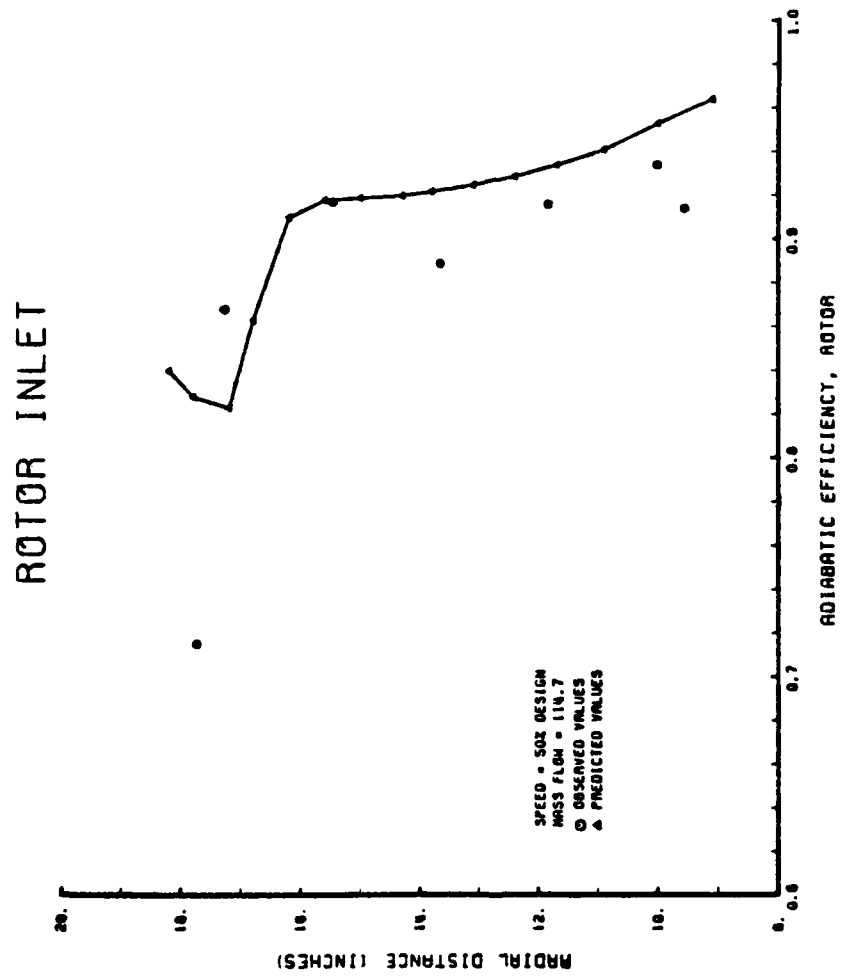


Figure 34. Adiabatic Efficiency of the Rotor, 50% Design

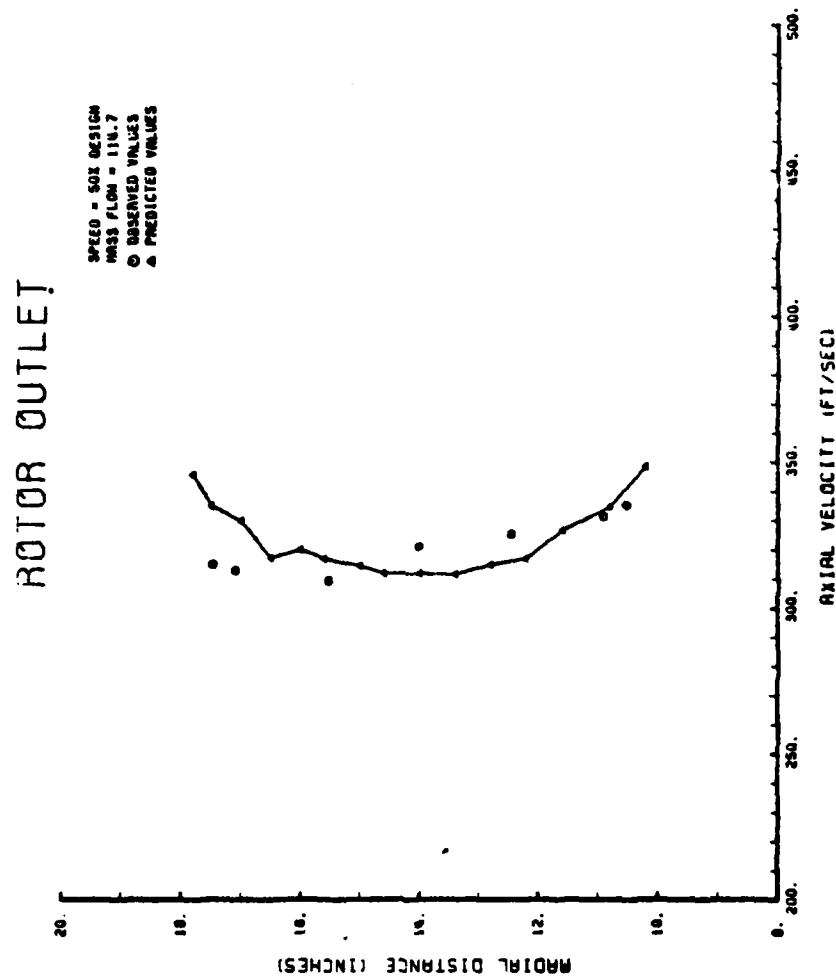


Figure 35. Axial Velocity at the Rotor Outlet, 50% Design

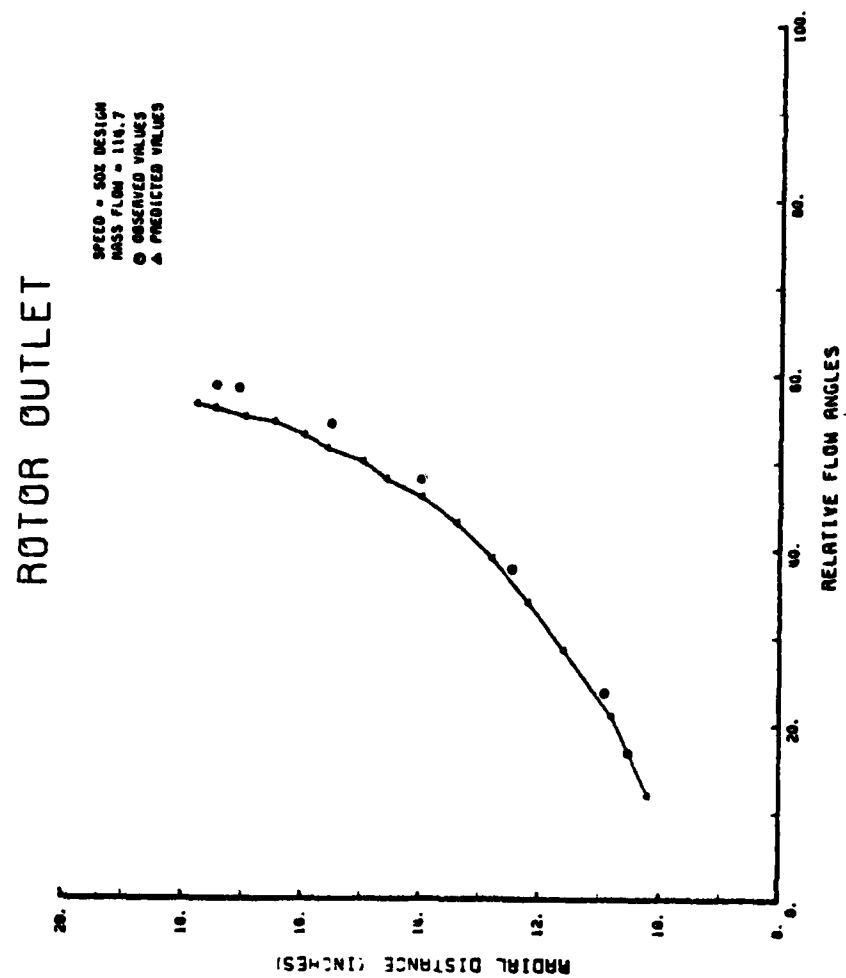


Figure 36. Relative Flow Angles at Rotor Outlet, 50% Design

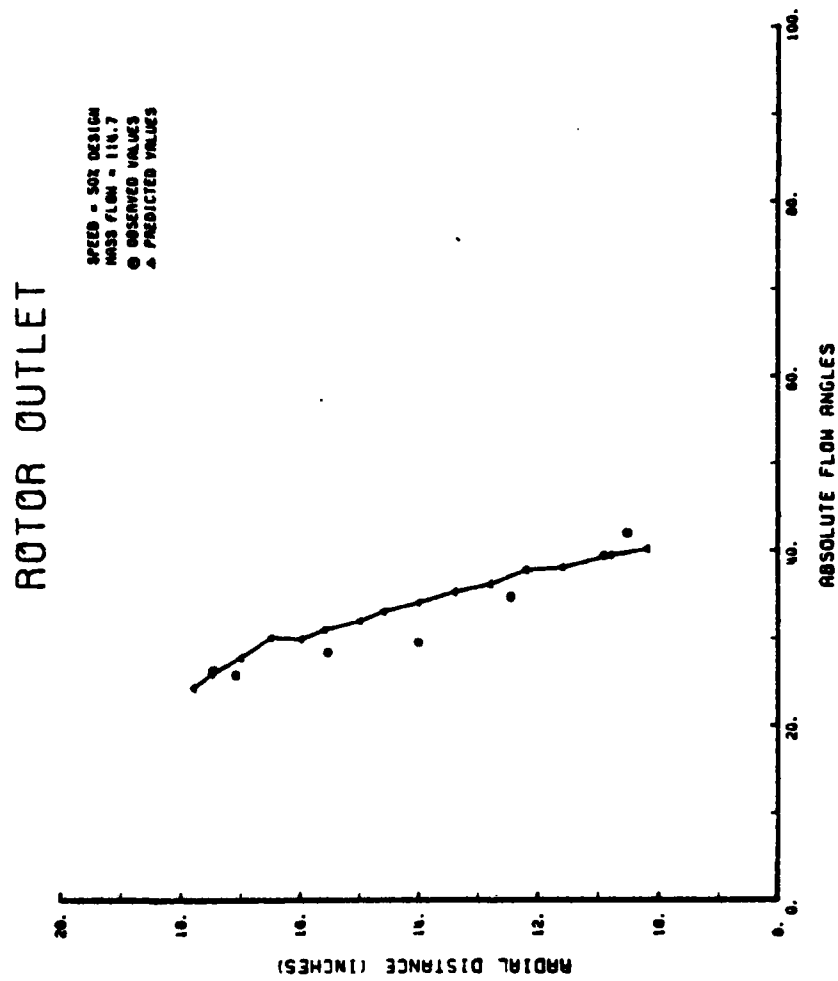


Figure 37. Absolute Flow Angles at Rotor Outlet, 50% Design

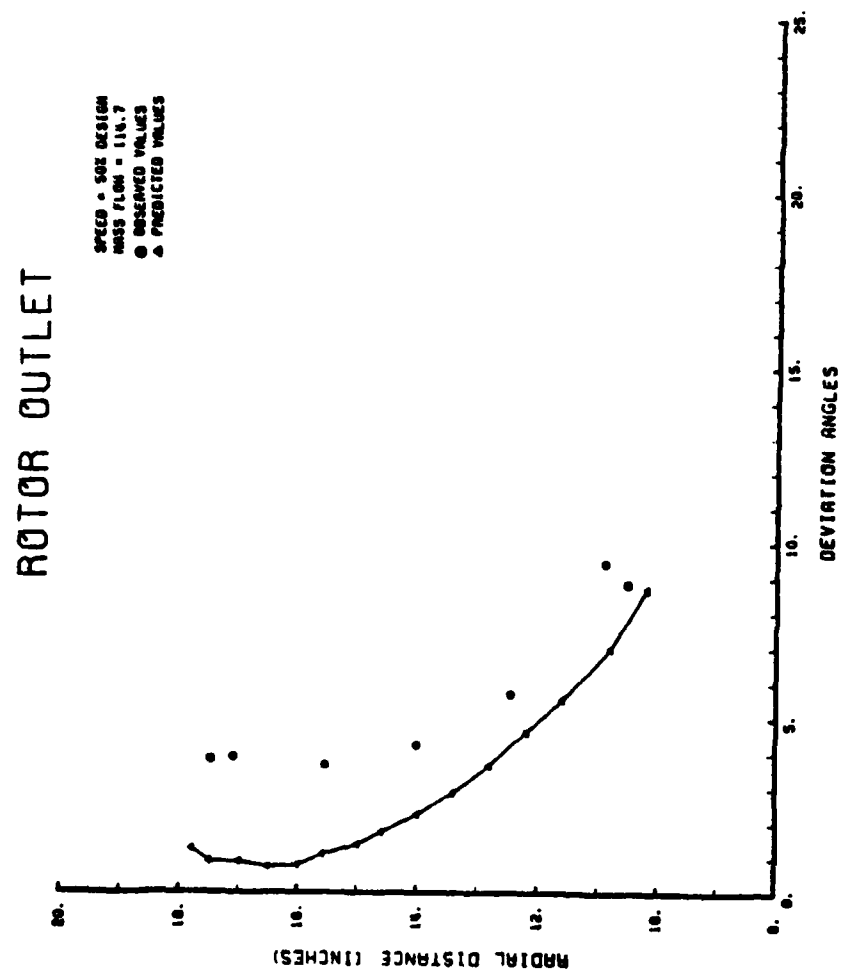


Figure 38. Deviation Flow Angles at Rotor Outlet, 50% Design

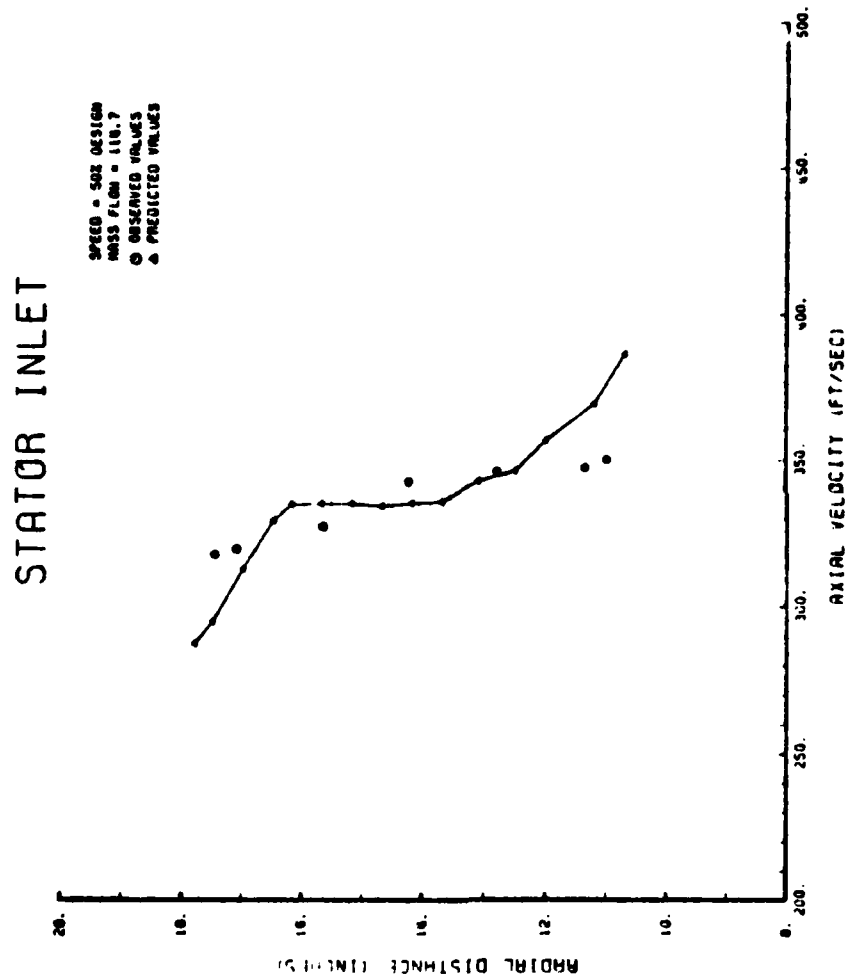


Figure 39. Axial Velocity at the Stator Inlet, 50% Design

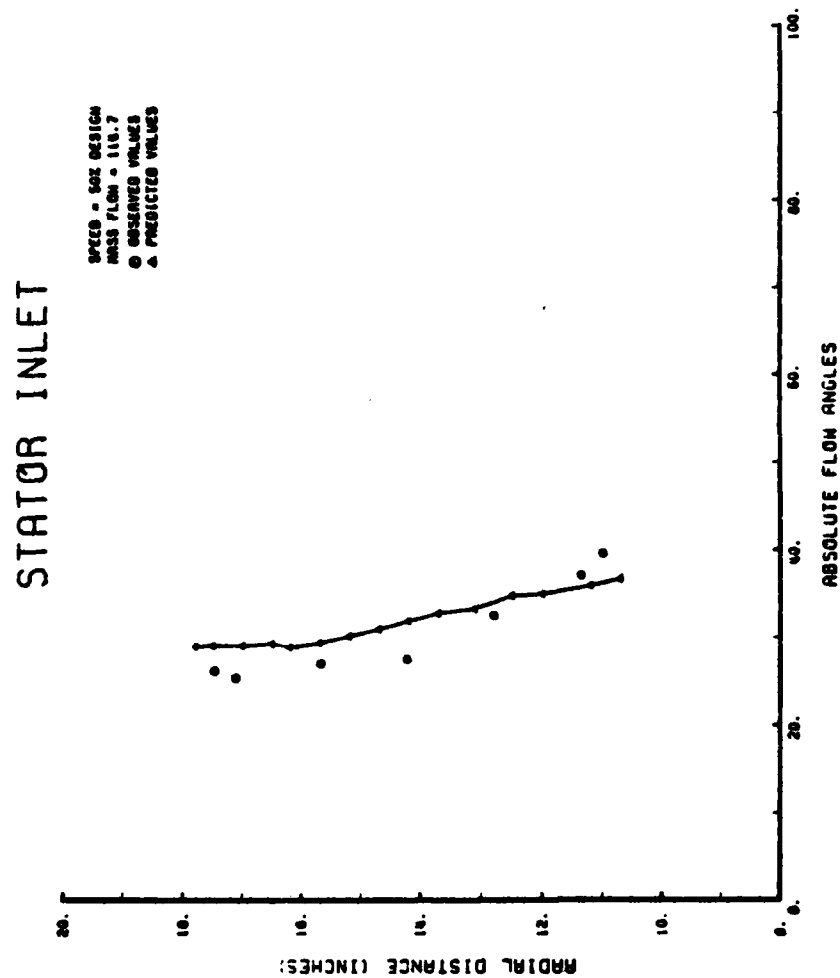


Figure 40. Absolute Flow Angles at Stator Inlet, 50% Design

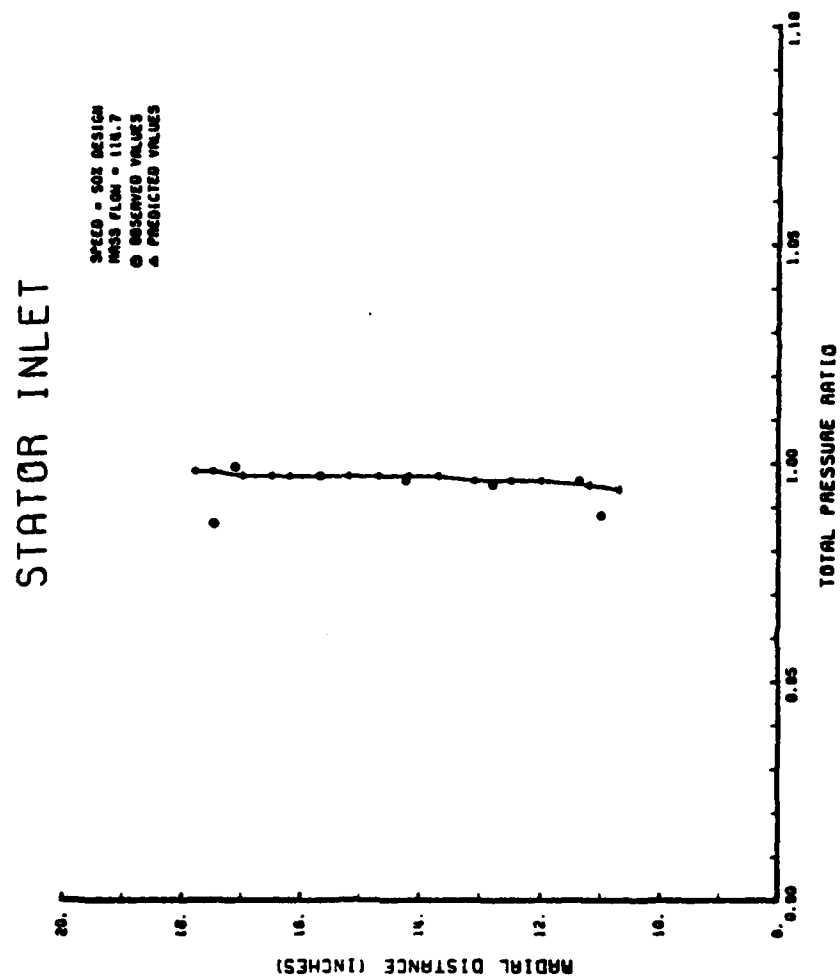


Figure 41. Total Pressure Ratio at Stator Inlet, 50% Design

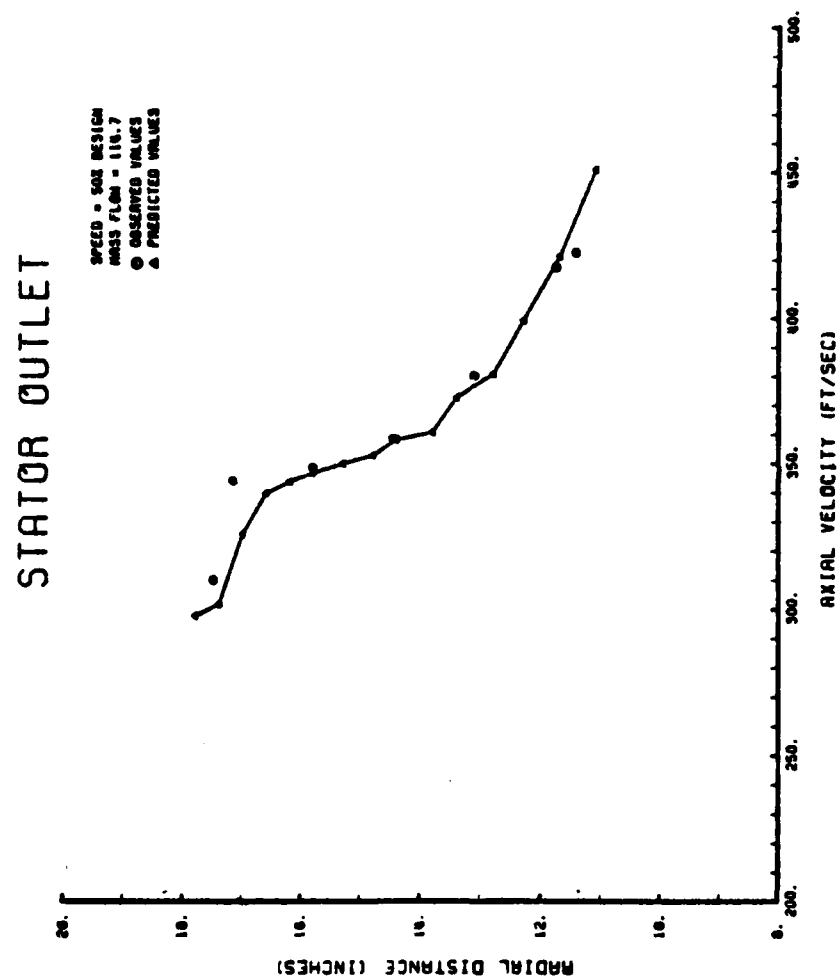


Figure 42. Axial Velocity at the Stator Outlet, 50% Design

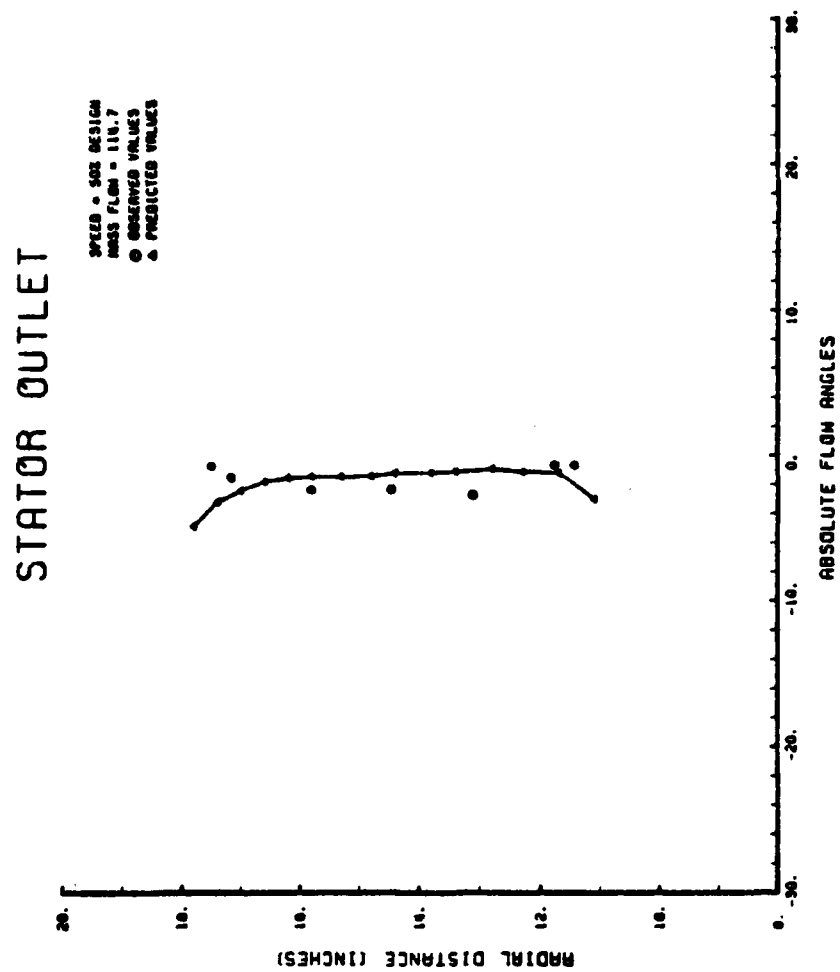


Figure 43. Absolute Flow Angles at Stator Outlet, 50% Design

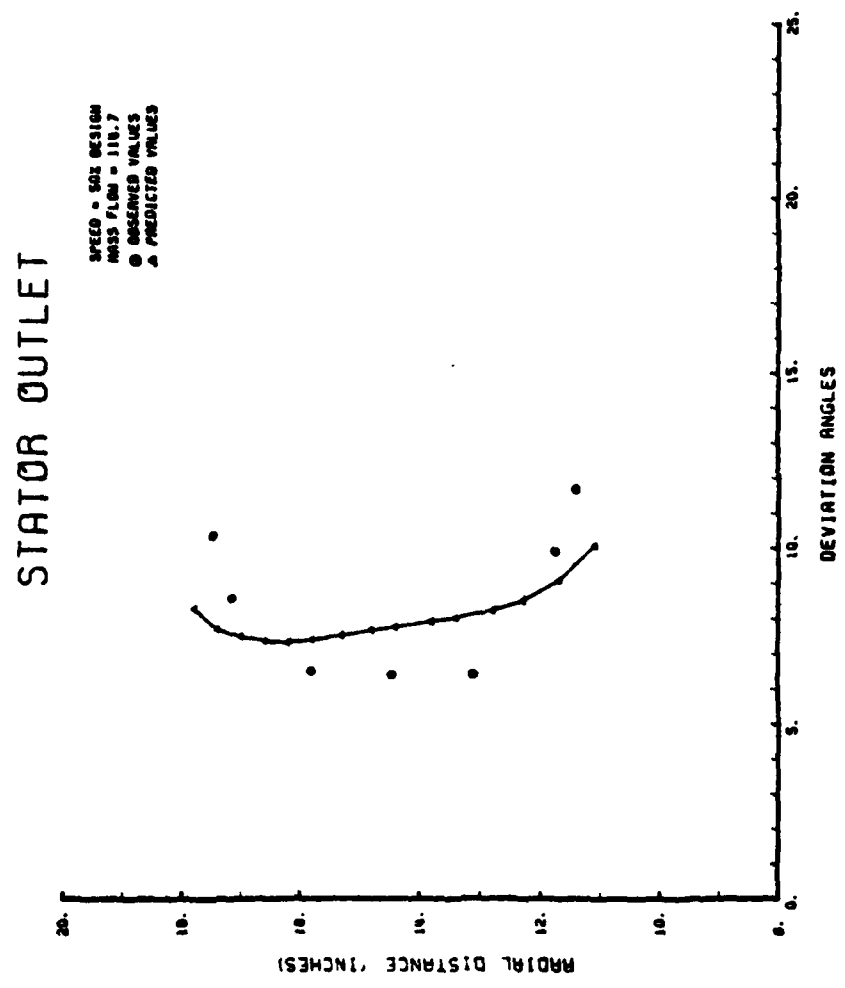


Figure 44. Deviation Flow Angles at Stator Outlet, 50% Design

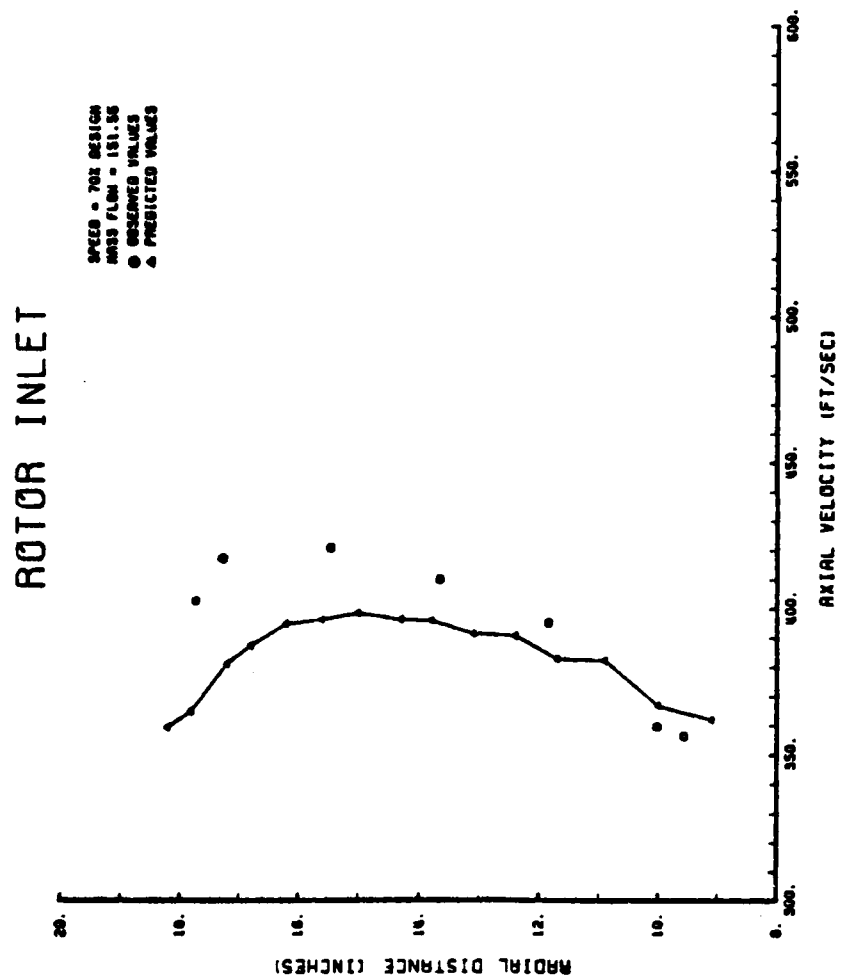


Figure 45. Axial Velocity at the Rotor Inlet, 70% Design

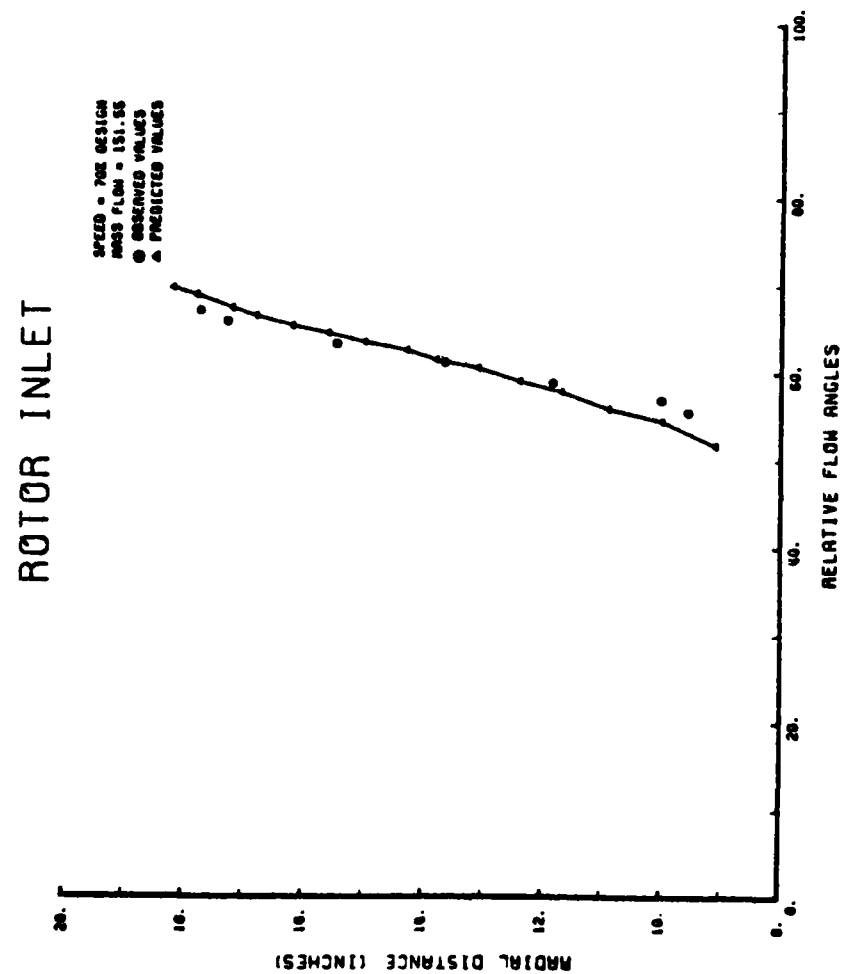


Figure 46. Relative Flow Angles at Rotor Inlet, 70% Design

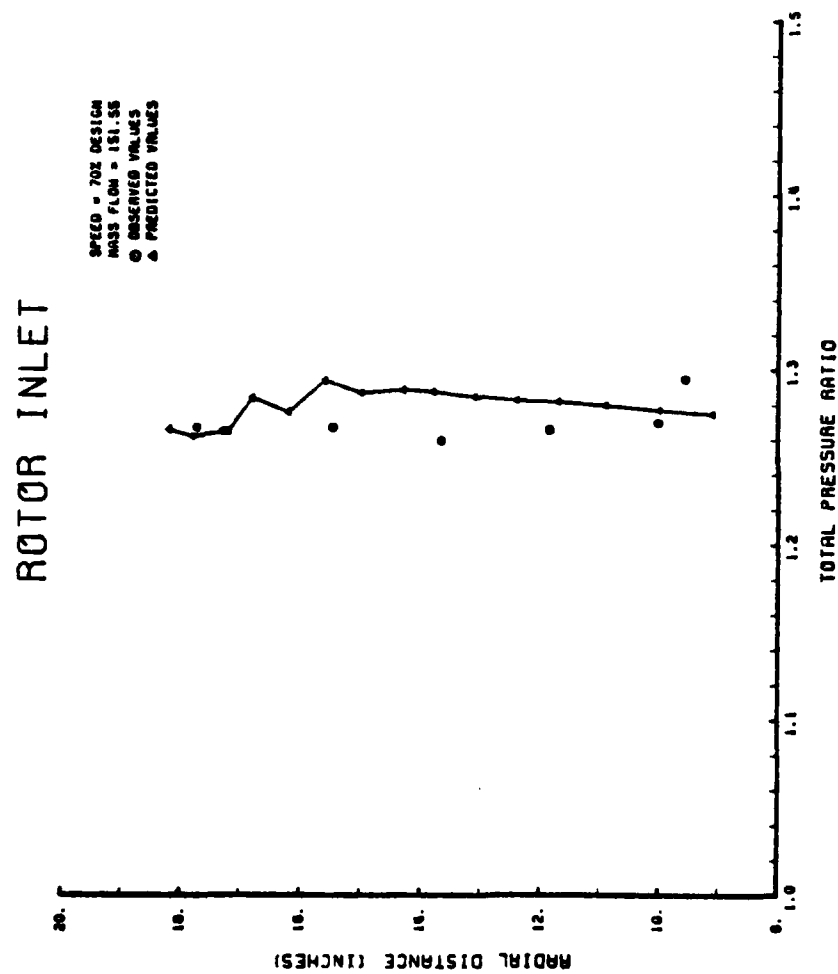


Figure 47. Total Pressure Ratio of the Rotor, 70% Design

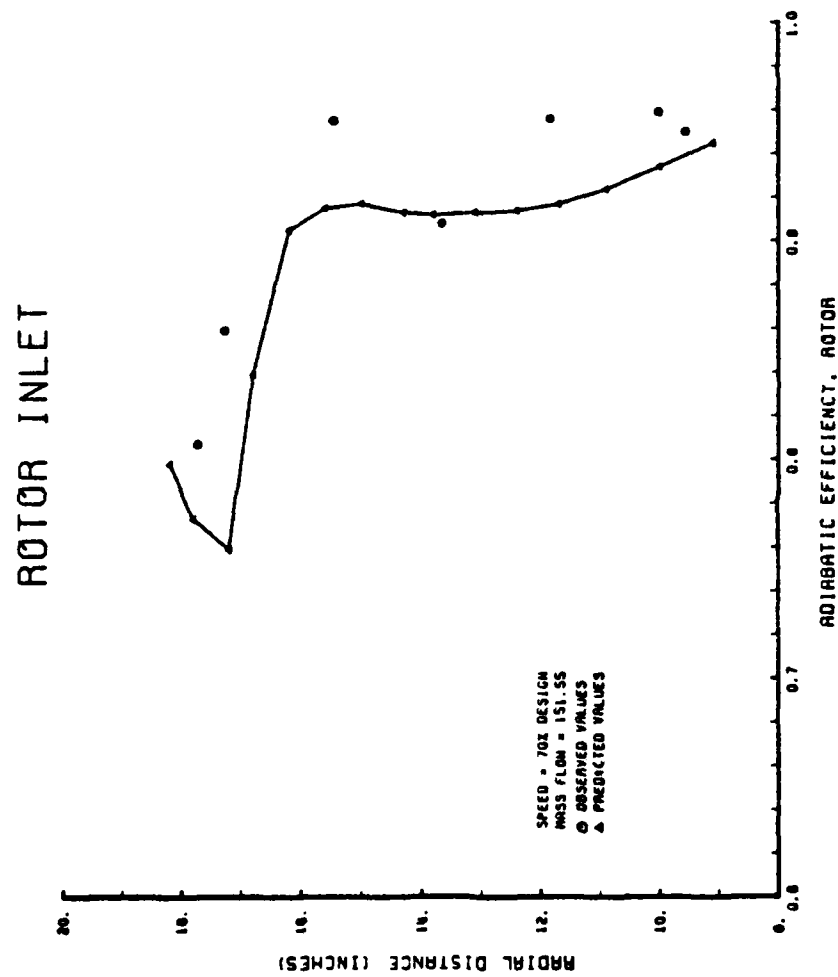


Figure 48. Adiabatic Efficiency of the Rotor, 70% Design

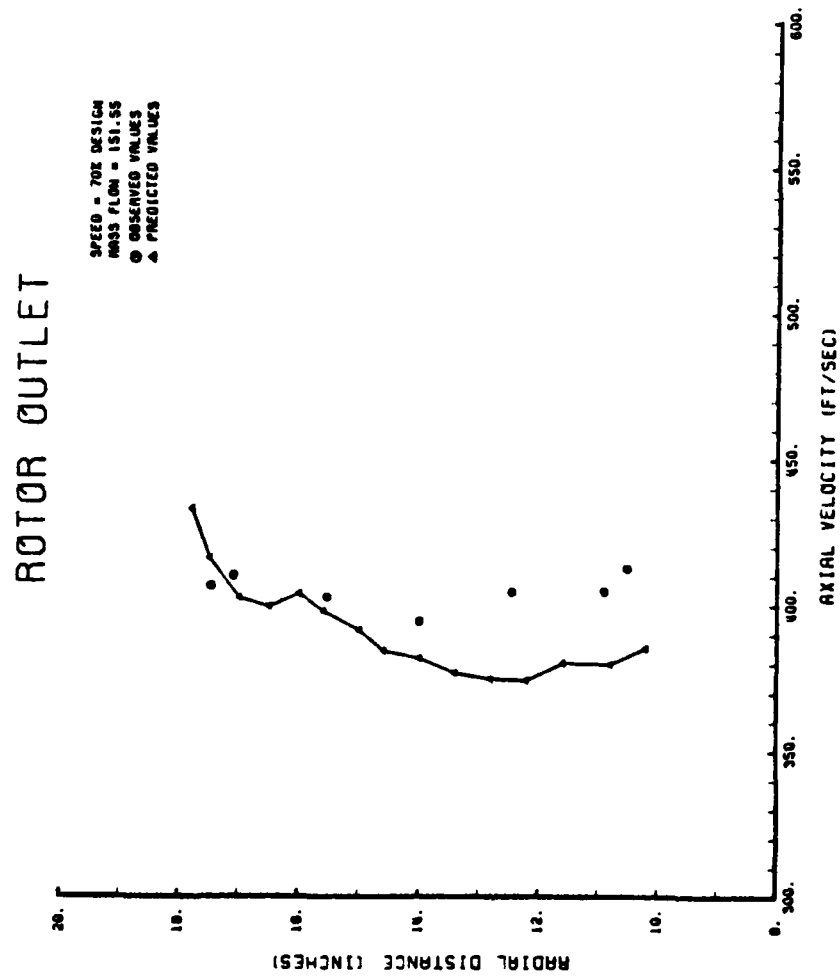


Figure 49. Axial Velocity at the Rotor Outlet, 70% Design

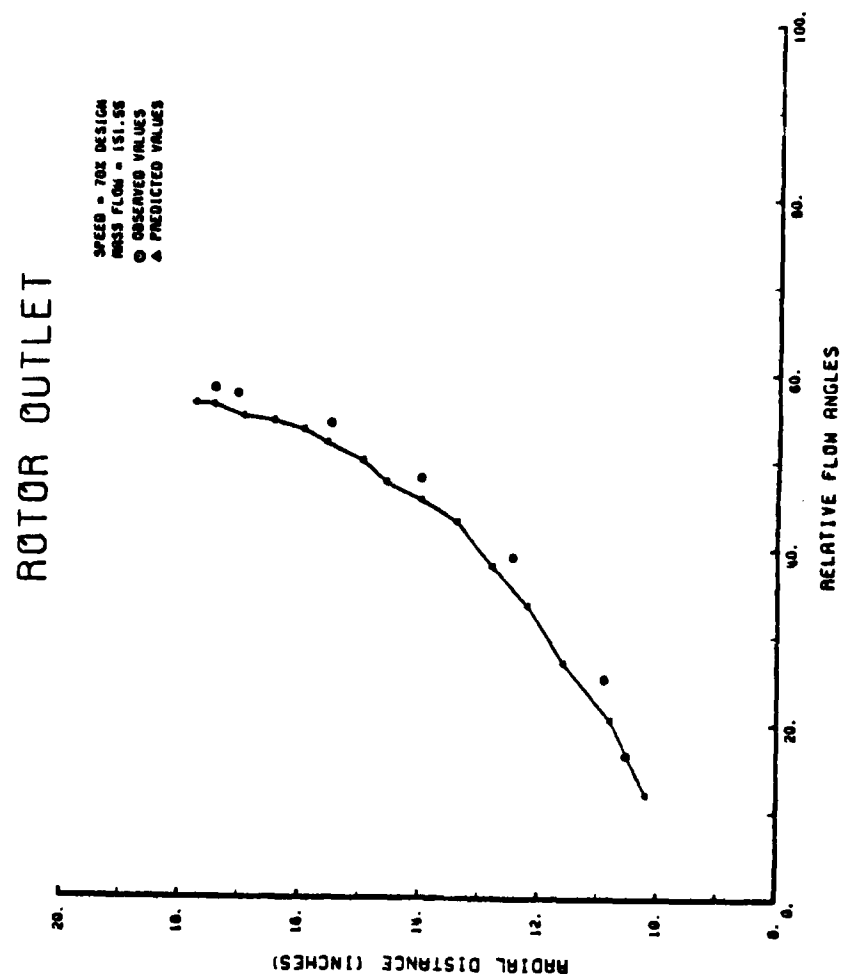


Figure 50. Relative Flow Angles at Rotor Outlet, 70% Design

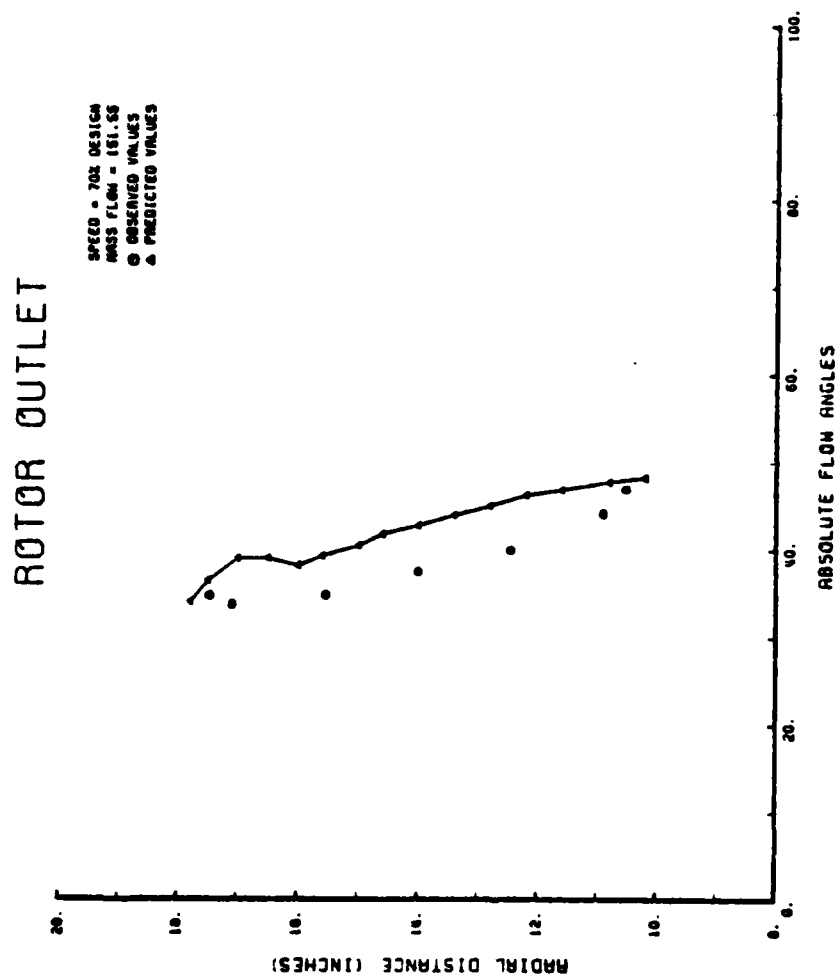


Figure 51. Absolute Flow Angles at Rotor Outlet, 70% Design

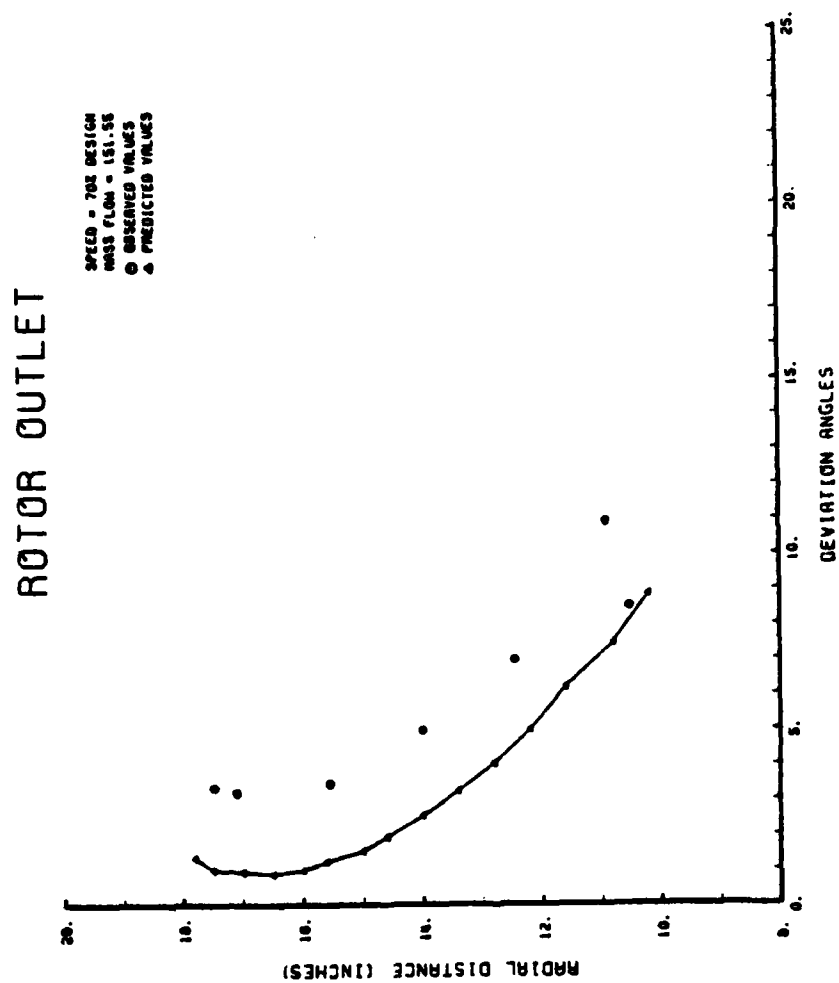


Figure 52. Deviation Flow Angles at Rotor Outlet, 70% Design

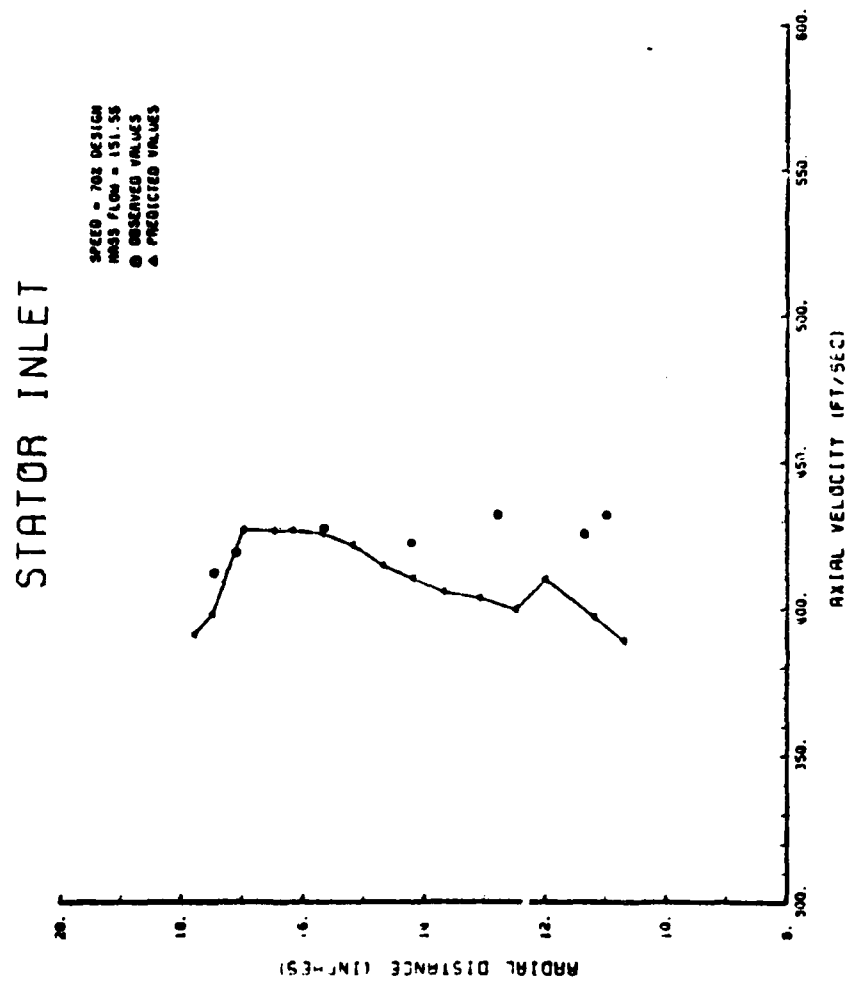


Figure 53. Axial Velocity at the Stator Inlet, 70% Design

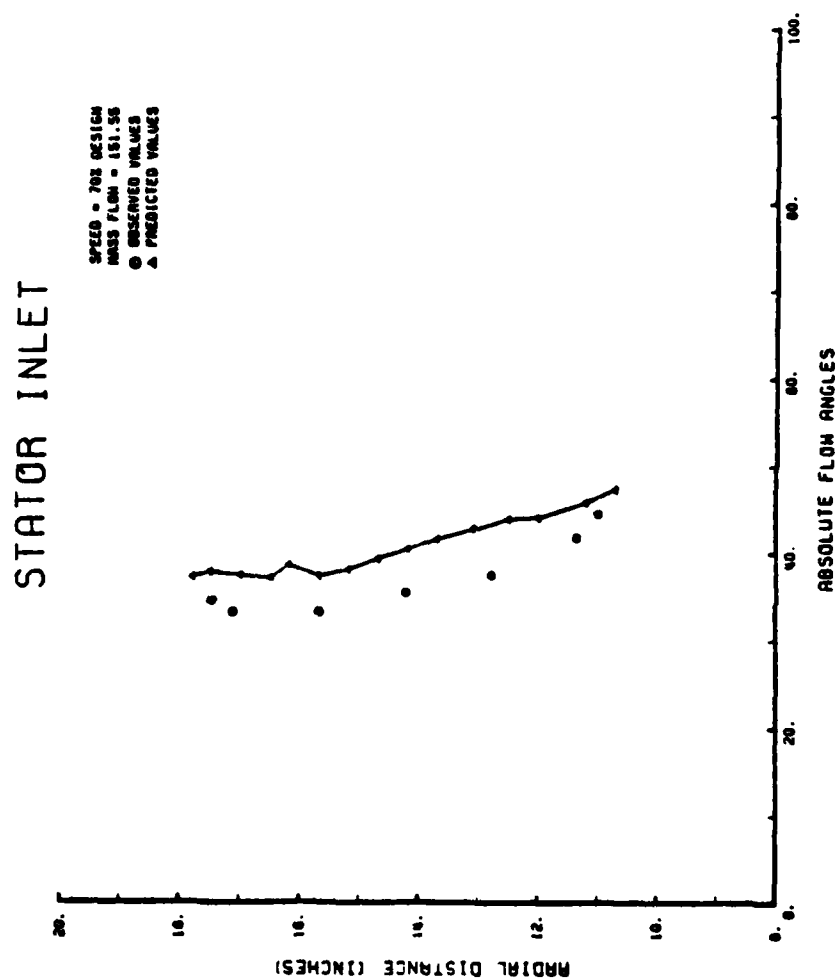


Figure 54. Absolute Flow Angles at Stator Inlet, 70% Design

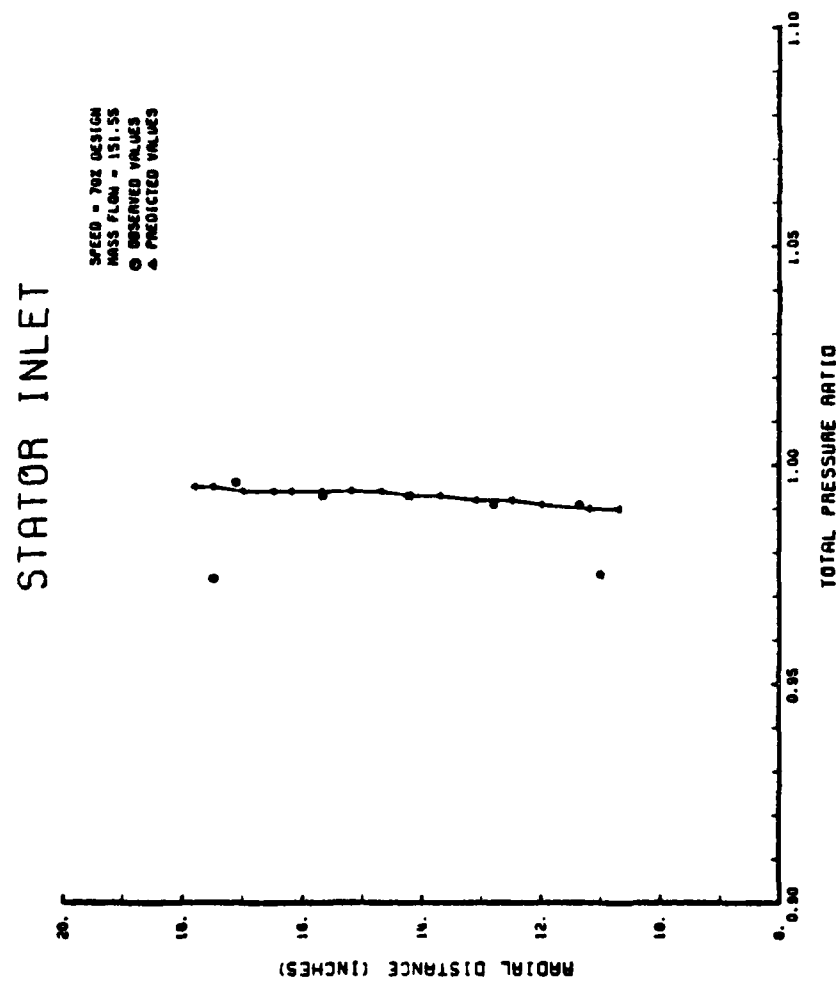


Figure 55. Total Pressure Ratio at Stator Inlet, 70% Design

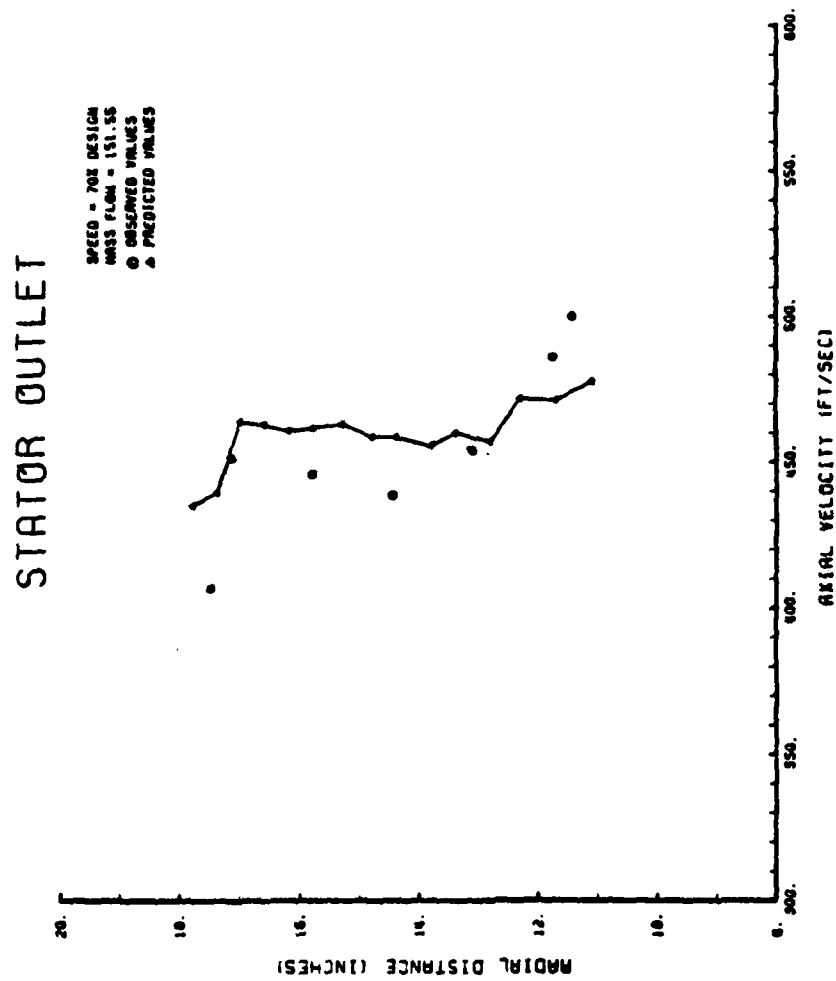


Figure 56. Axial Velocity at the Stator Outlet, 70% Design

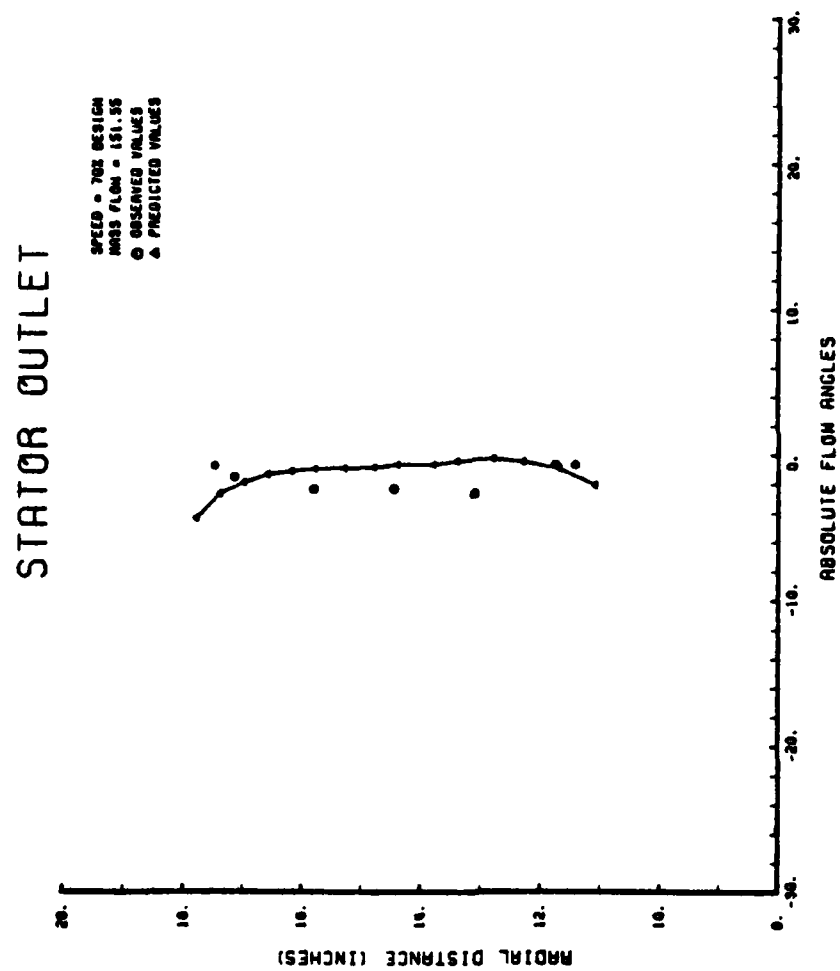


Figure 57. Absolute Flow Angles at Stator Outlet, 70% Design

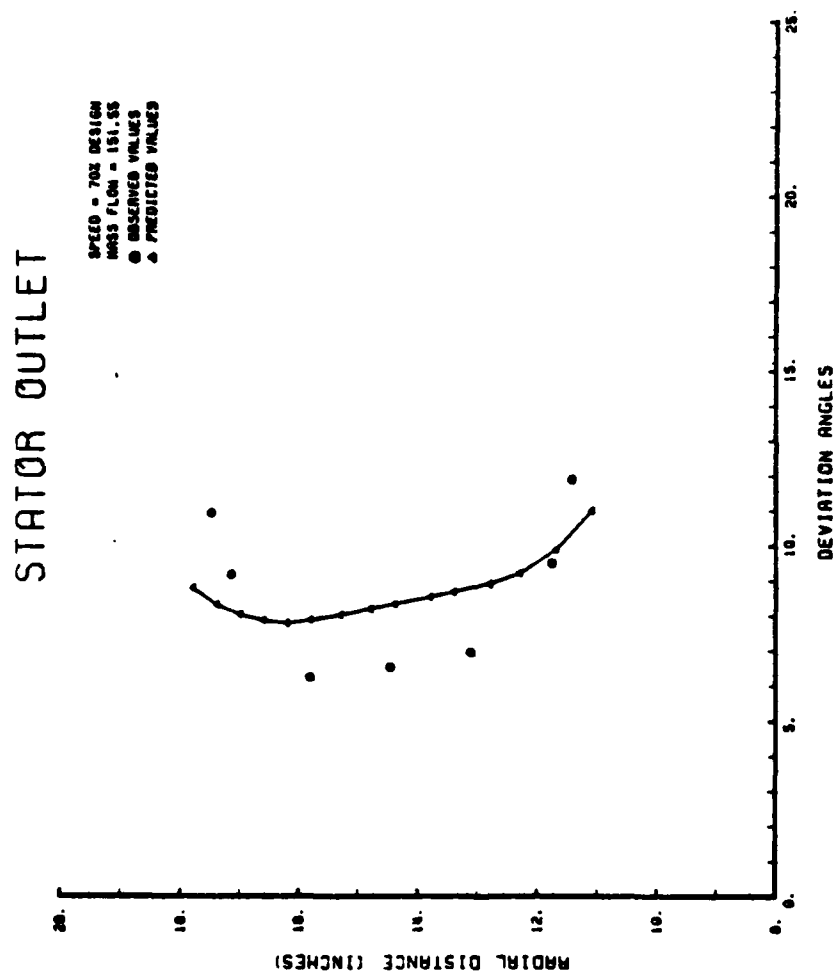


Figure 58. Deviation Flow Angles at Stator Outlet, 70% Design

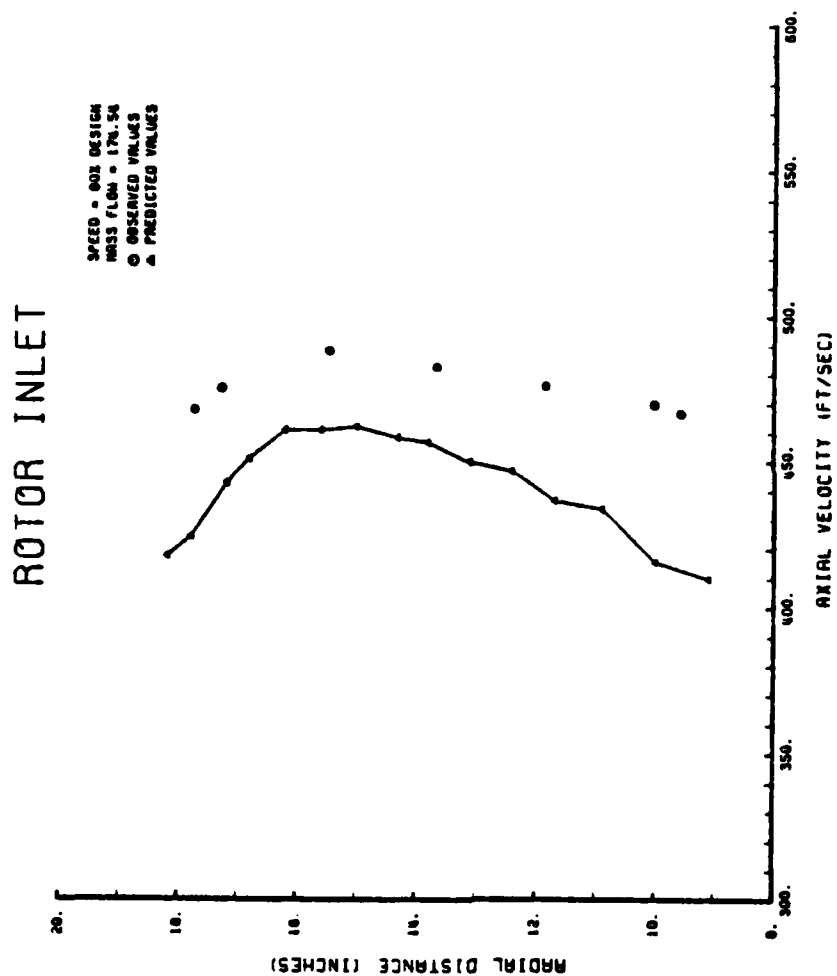


Figure 59. Axial Velocity at the Rotor Inlet, 80% Design

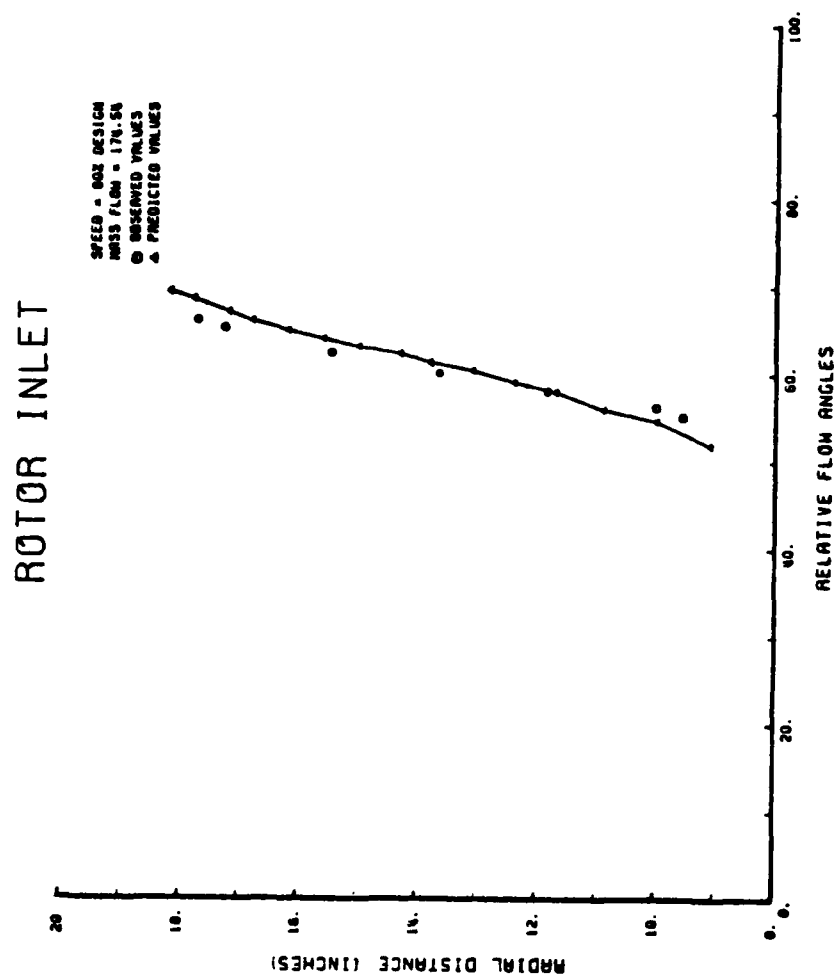


Figure 60. Relative Flow Angles at Rotor Inlet, 80% Design

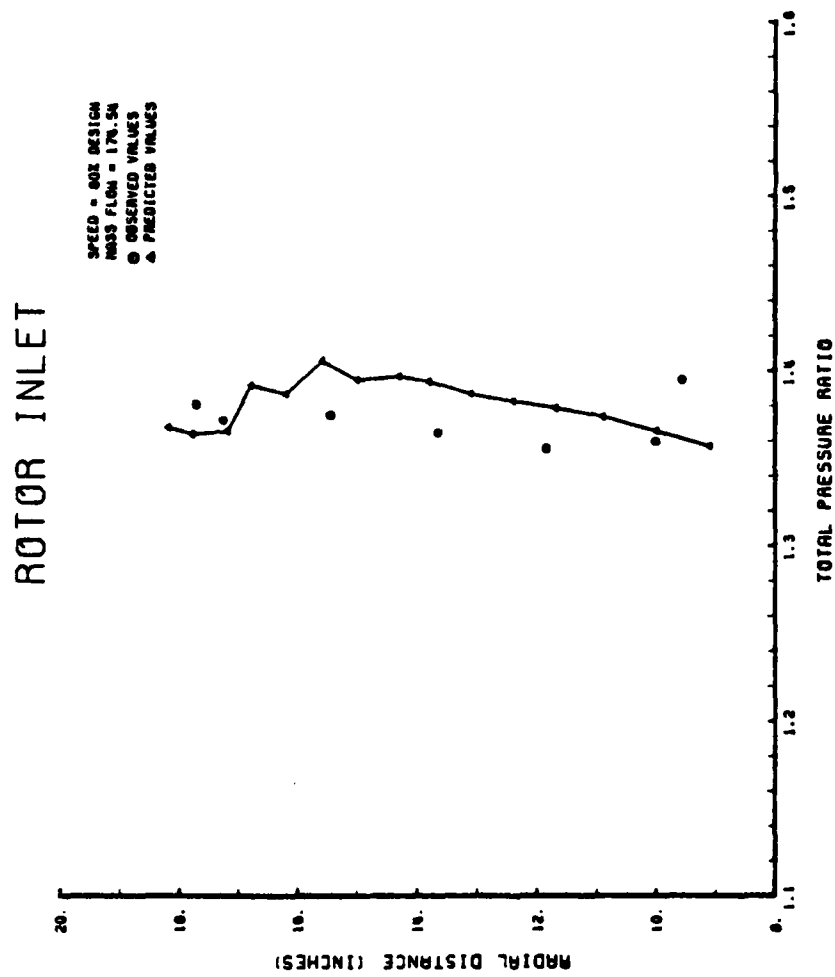


Figure 61. Total Pressure Ratio of the Rotor, 80% Design

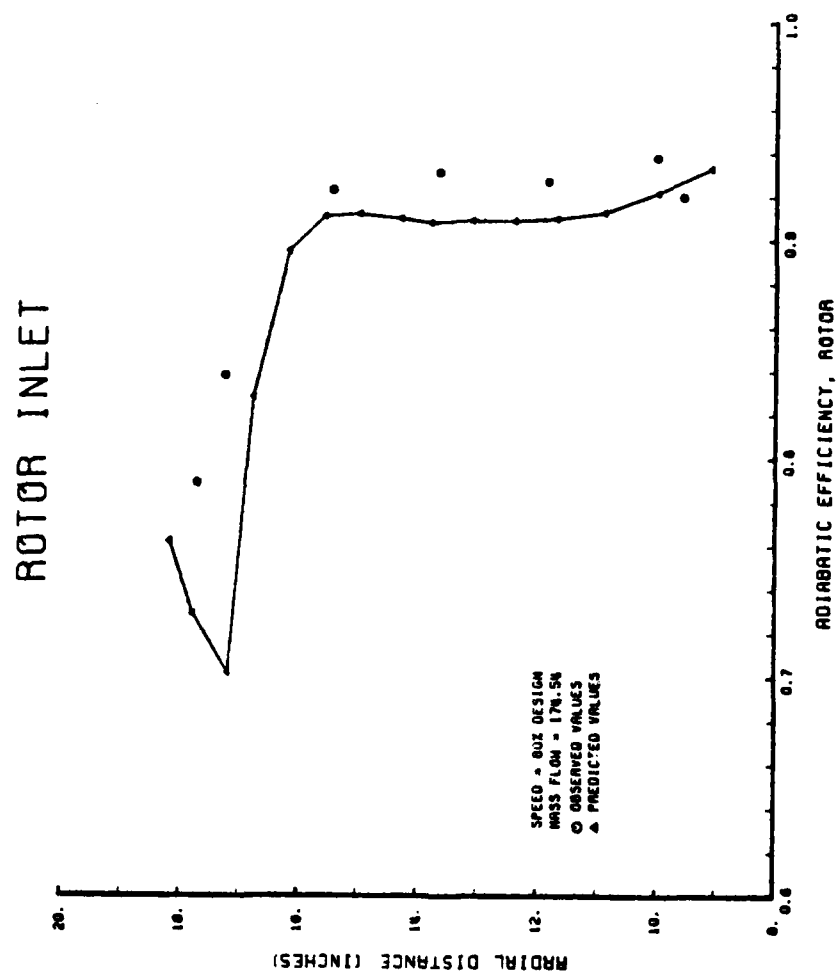


Figure 62. Adiabatic Efficiency of the Rotor, 80% Design

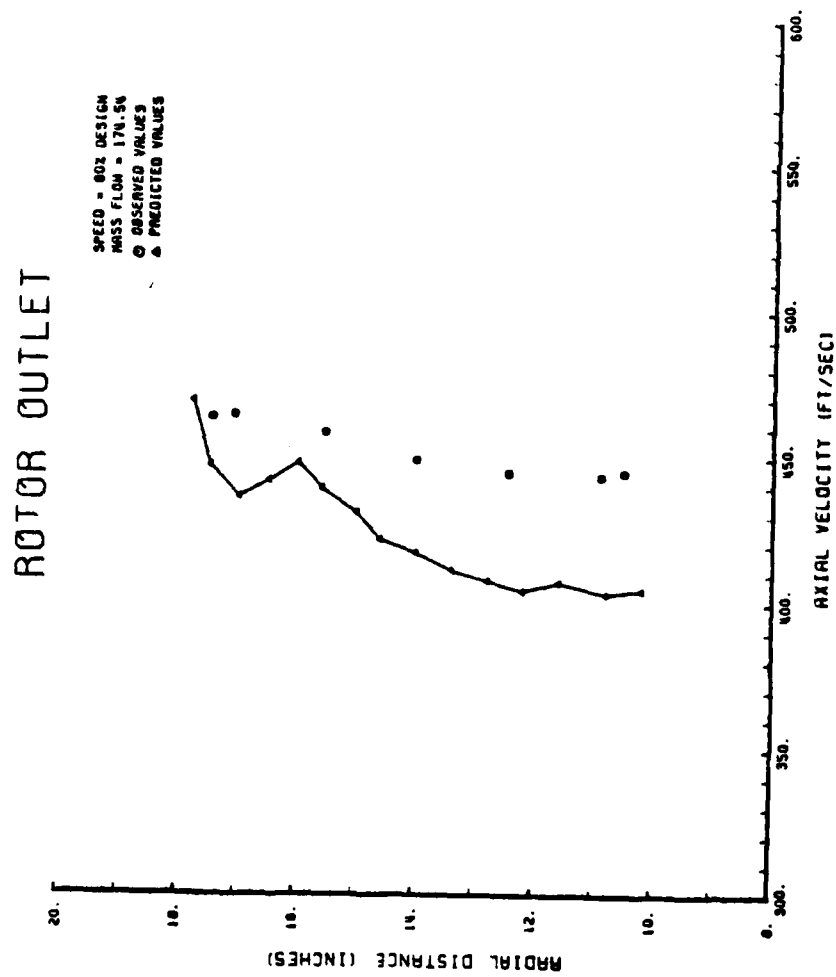


Figure 63. Axial Velocity at the Rotor Outlet, 80% Design

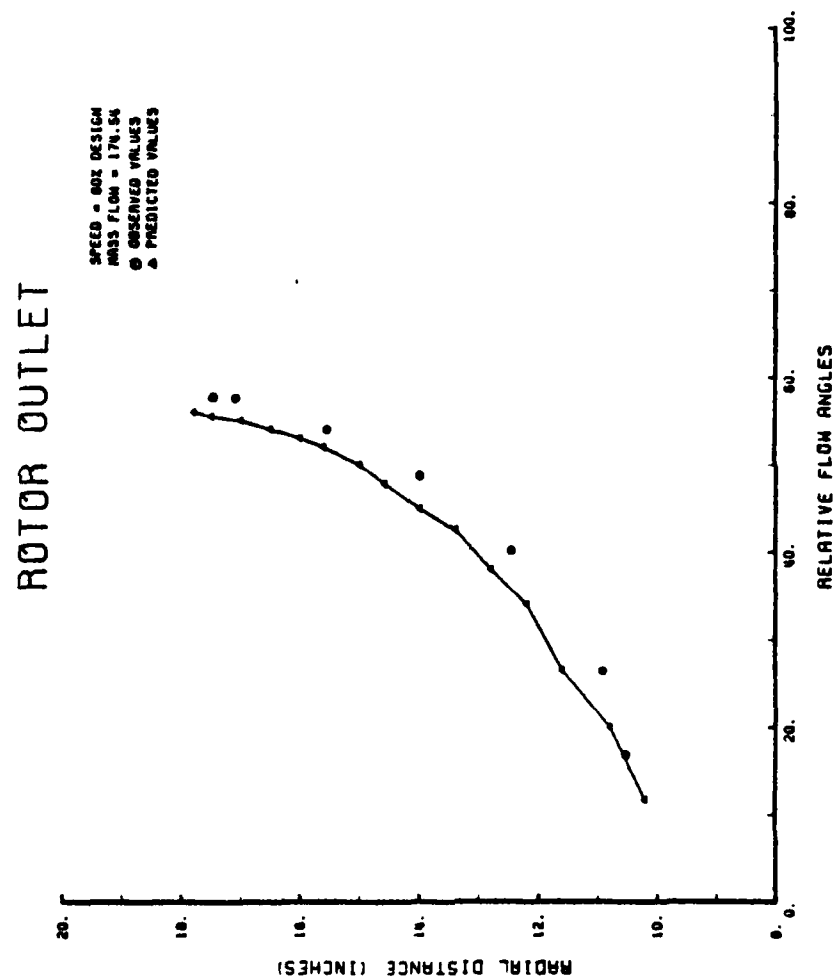


Figure 64. Relative Flow Angles at Rotor Outlet, 80% Design

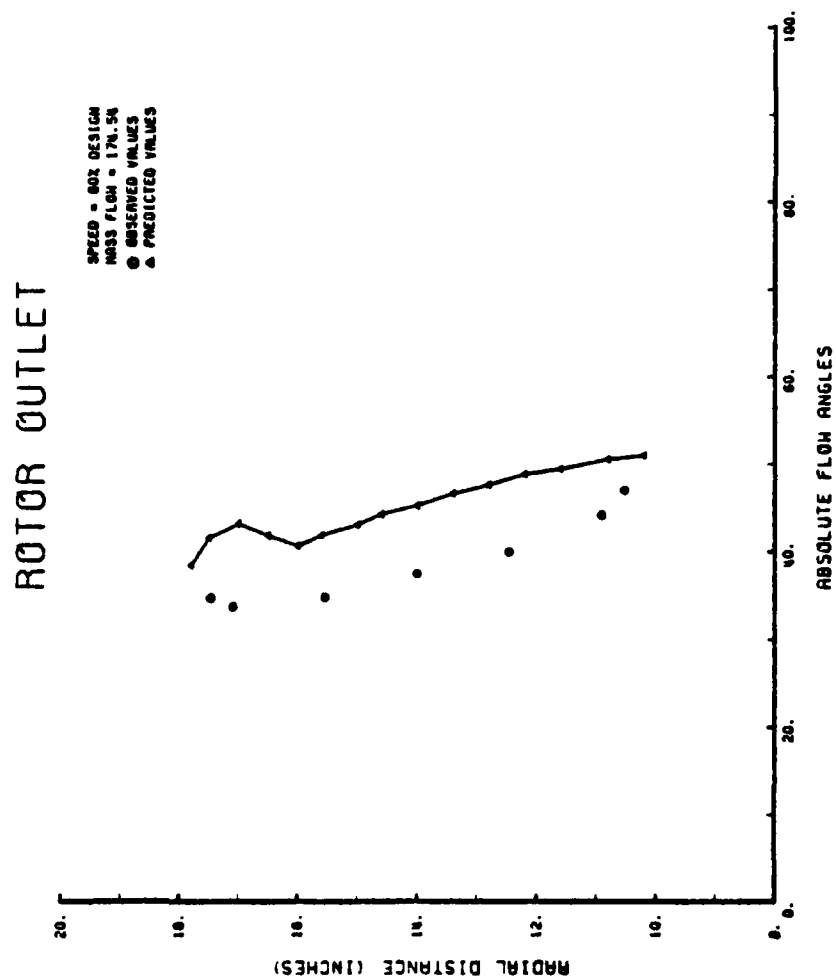


Figure 65. Absolute Flow Angles at Rotor Outlet, 80% Design

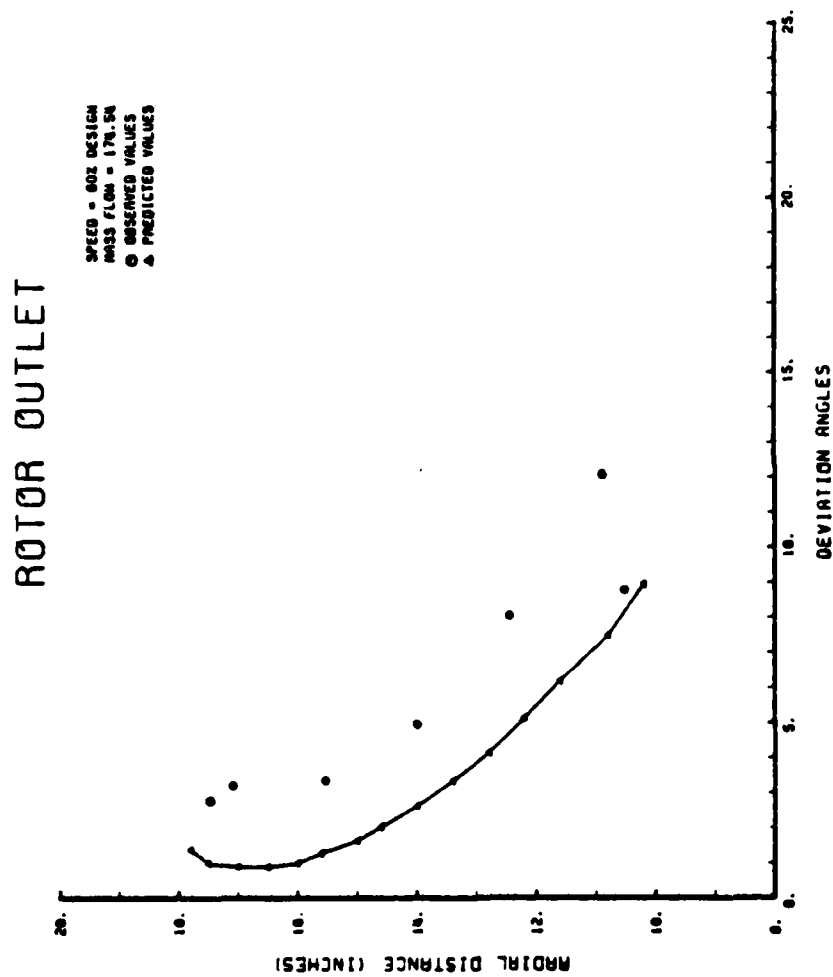


Figure 66. Deviation Flow Angles at Rotor Outlet, 80% Design

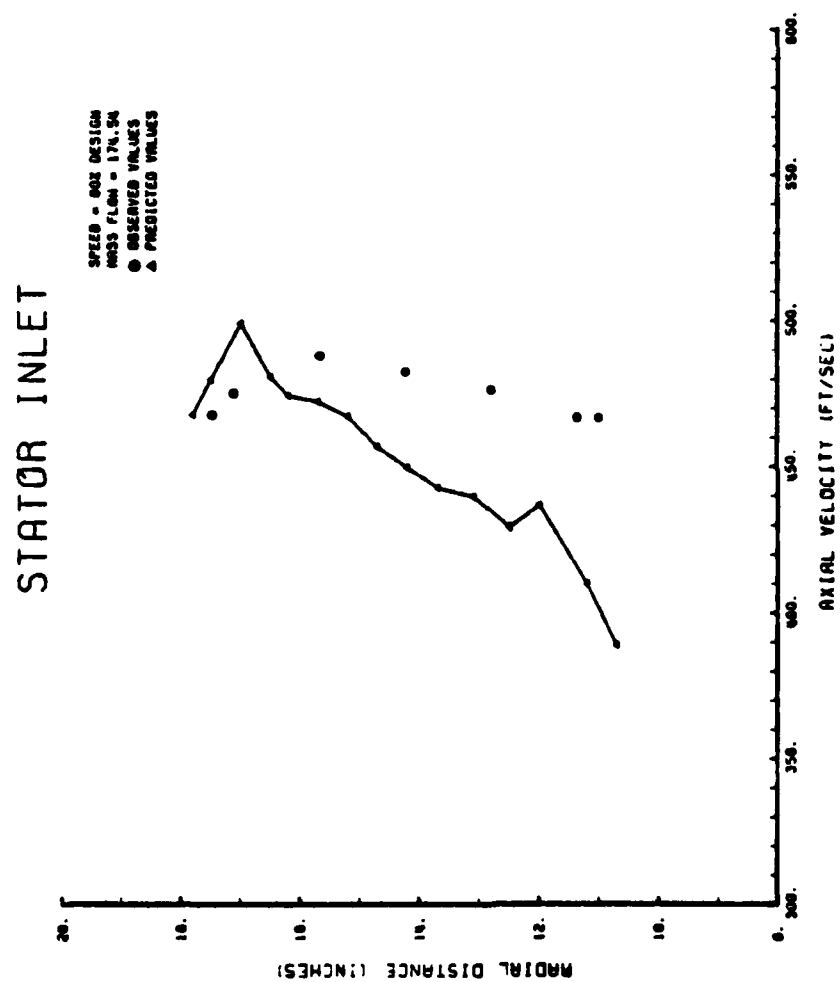


Figure 67. Axial Velocity at the Stator Inlet, 80% Design

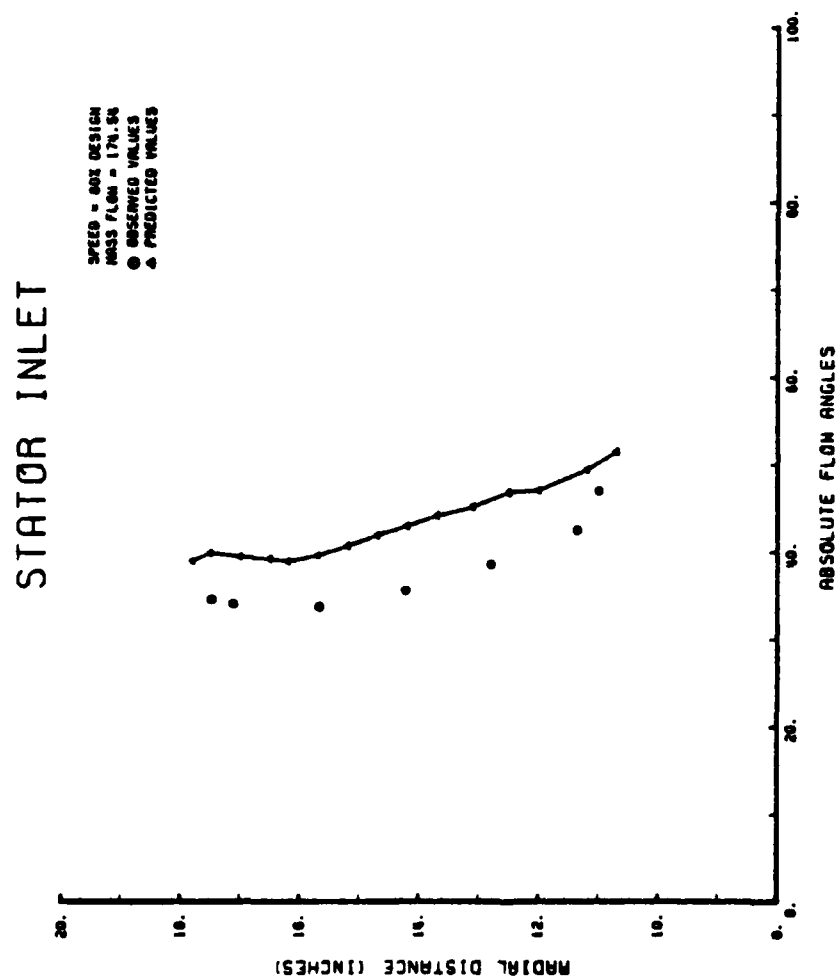


Figure 68. Absolute Flow Angles at Stator Inlet, 80% Design

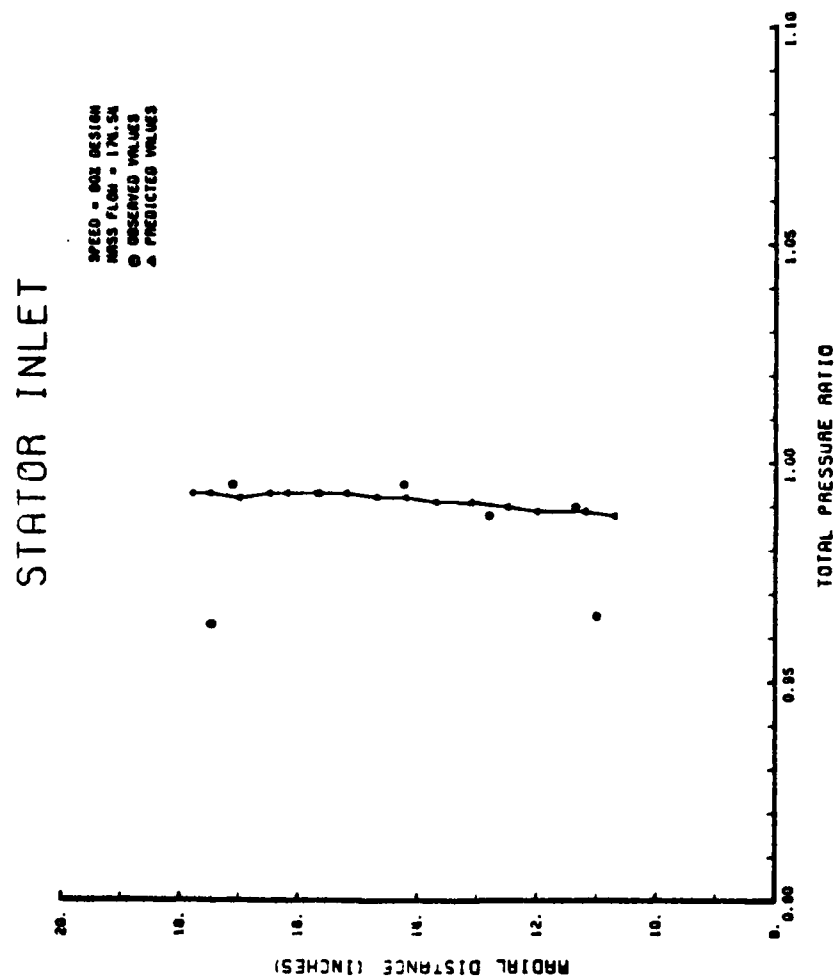


Figure 69. Total Pressure Ratio at Stator Inlet, 80% Design

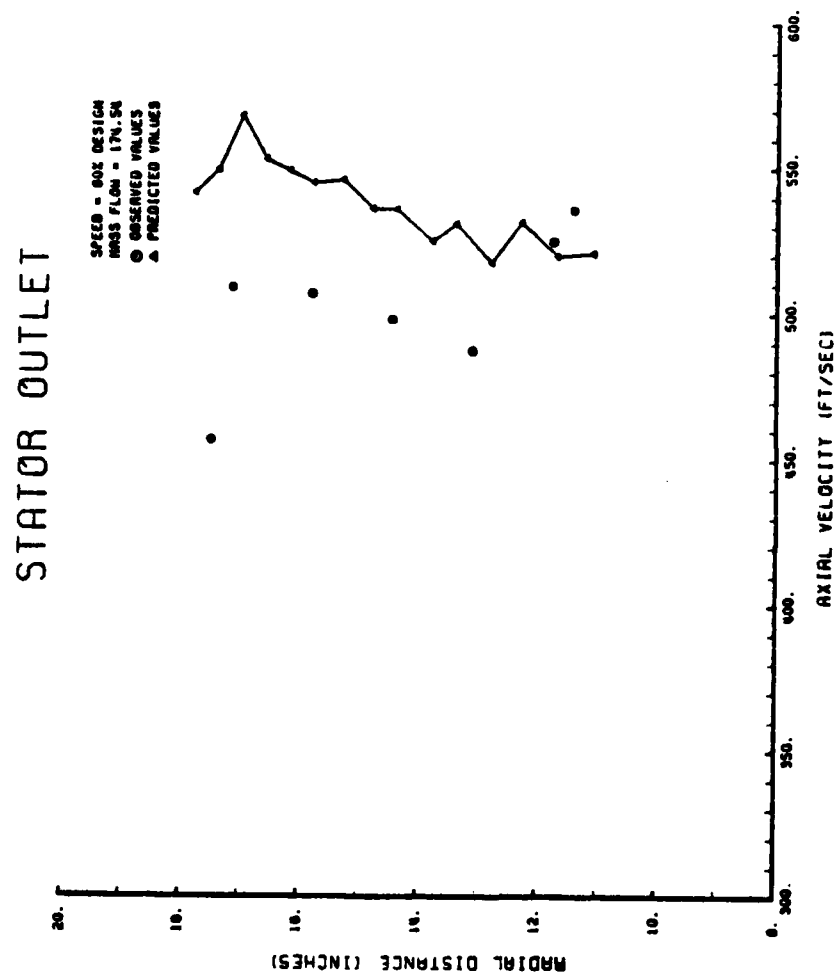


Figure 70. Axial Velocity at the Stator Outlet, 80% Design

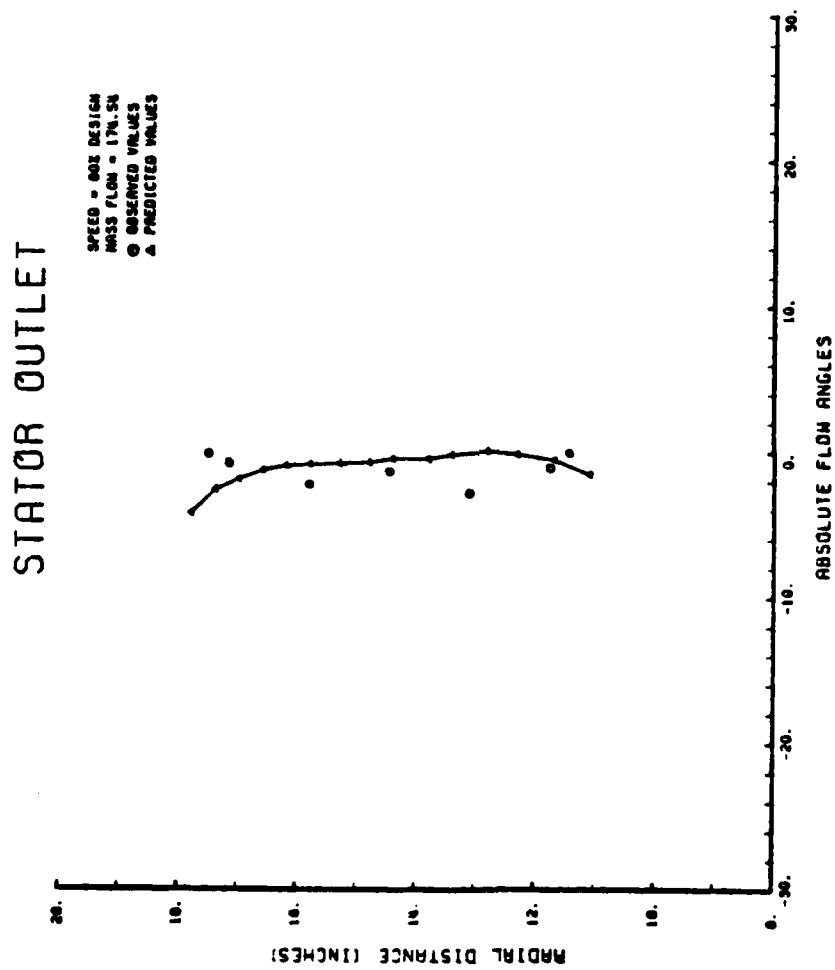


Figure 71. Absolute Flow Angles at Stator Outlet, 80% Design

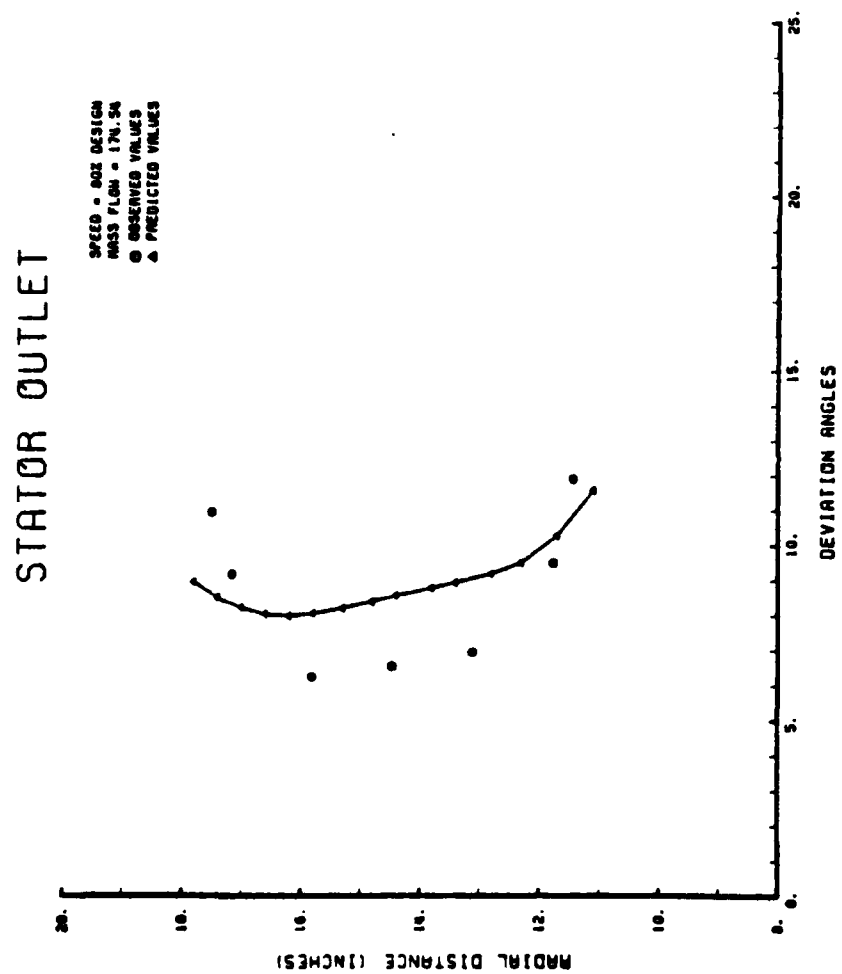


Figure 72. Deviation Flow Angles at Stator Outlet, 80% Design

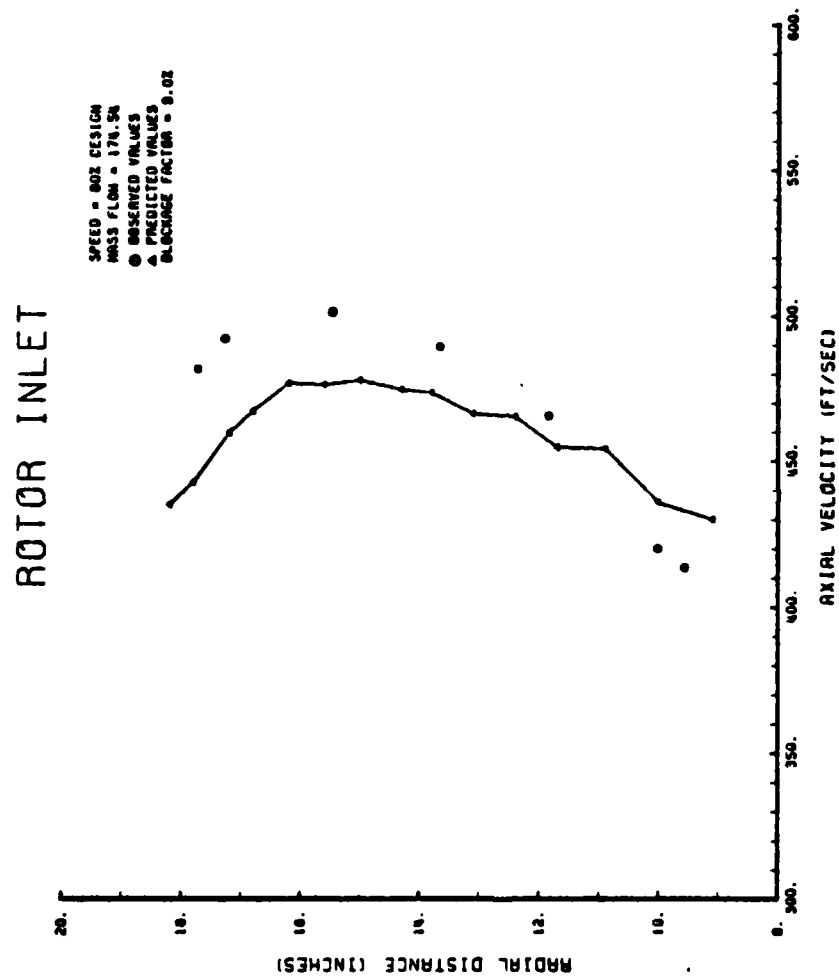


Figure 73. Axial Velocity at the Rotor Inlet, 80% Design, 9% Blockage

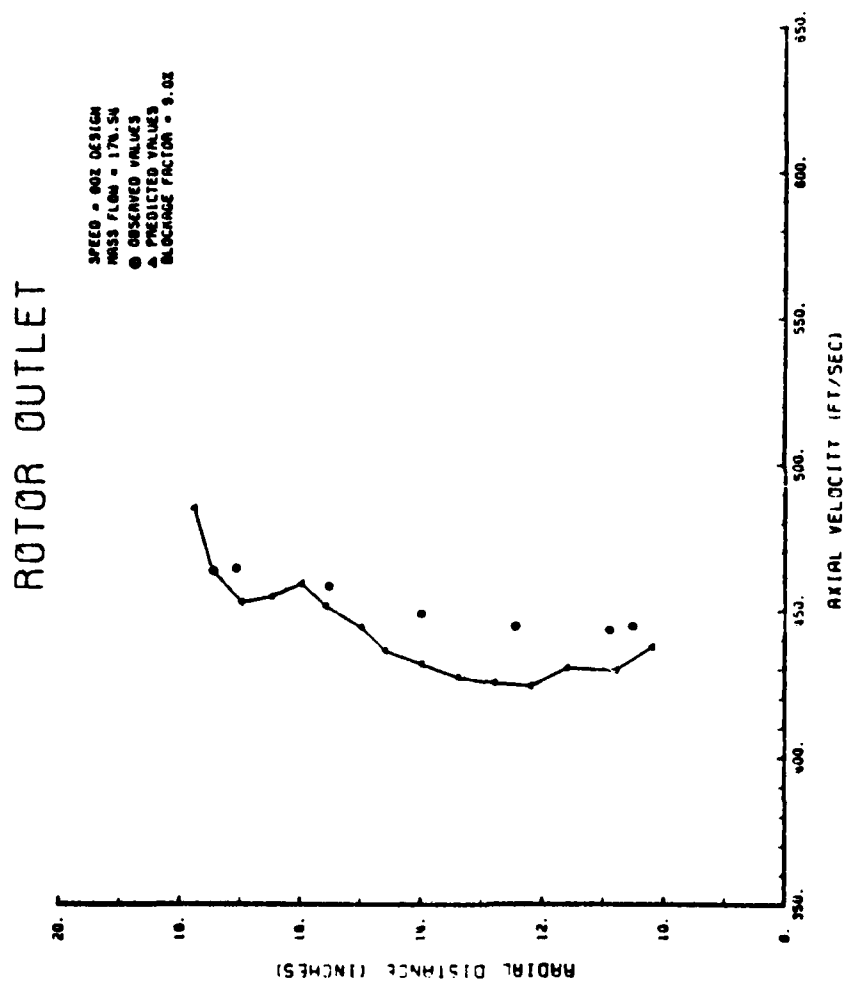


Figure 74. Axial Velocity at the Rotor Outlet, 80% Design, 9% Blockage

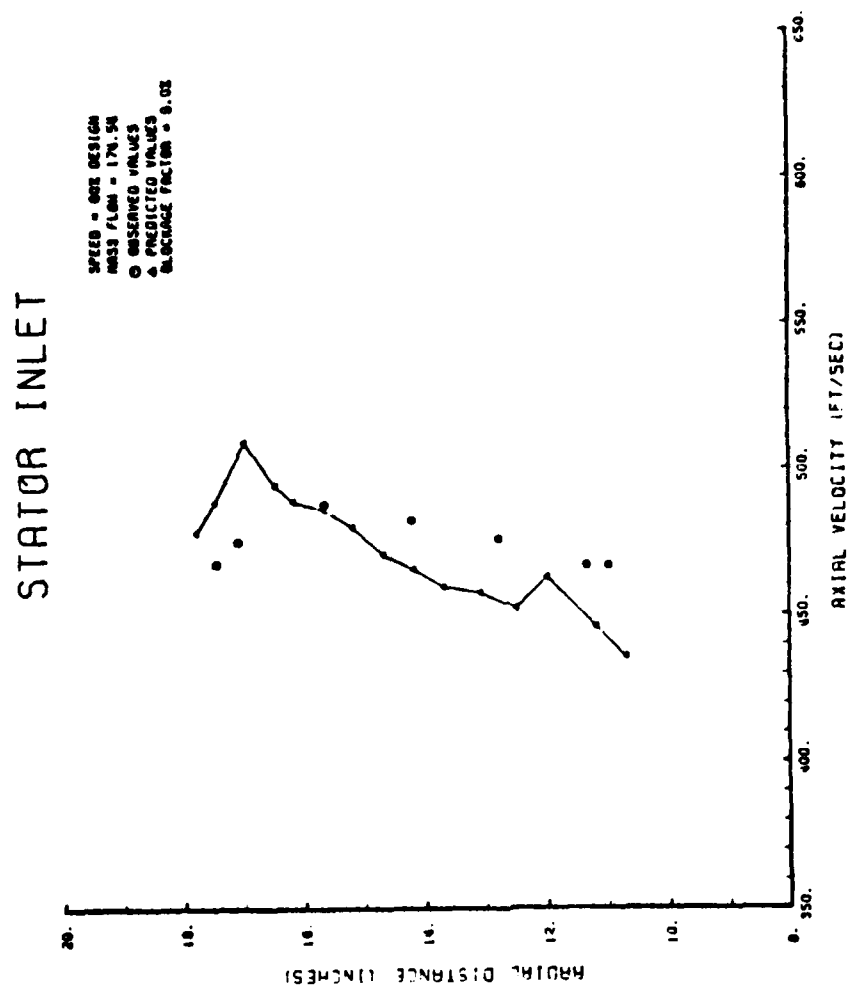


Figure 75. Axial Velocity at the Stator Inlet, 80% Design, 9% Blockage

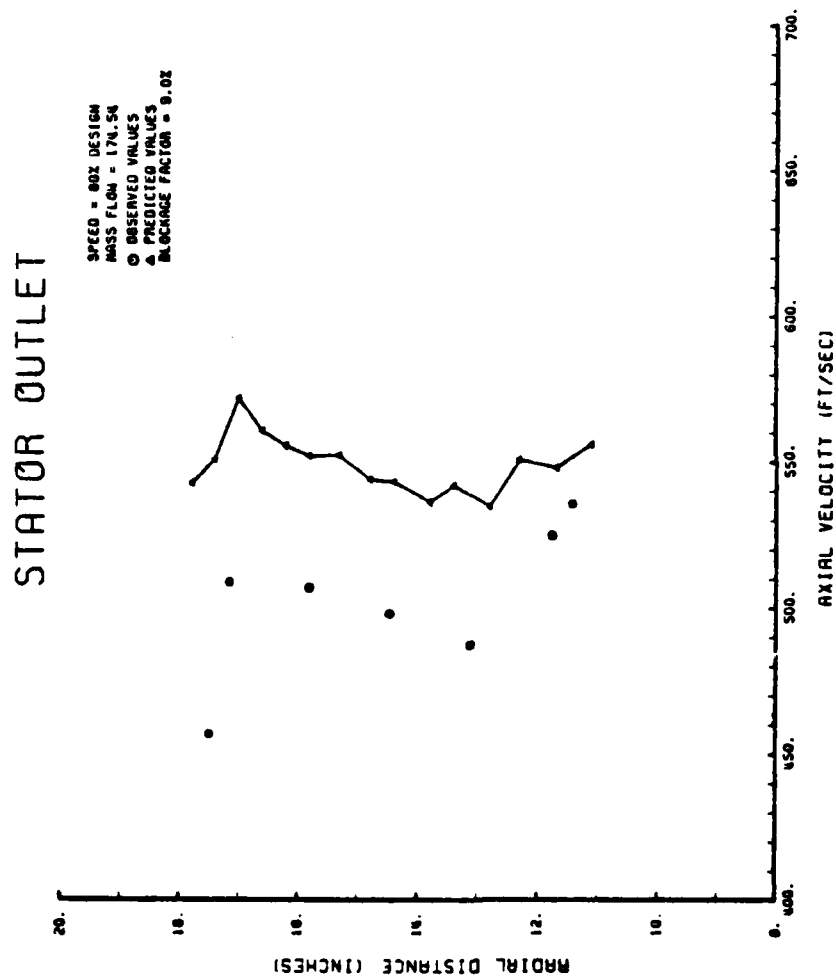


Figure 76. Axial Velocity at the Stator Outlet, 80% Design, 9% Blockage

APPENDIX A

STORAGE ALLOCATION FOR THE PROGRAM MESHGEN

The following list of pointers and variable names indicates the storage location of the first value of the corresponding array:

REAL 8 Variables (Array R8)

<u>Pointer Name</u>	<u>Array Name</u>	<u>Array Contents</u>
J1	XSE	Z Coordinates of the Super Elements
J2	YSE	R Coordinates of the Super Elements
J3	XXSE	Temporary Storage of the Z Coordinates of the Super Element Subdivision
J4	YYSE	Temporary Storage of the R Coordinates of the Super Element Subdivision
J5	ZC	Z Coordinates
J6	RC	R Coordinates
J7	PSI	Stream Function
J8	B	Blockage Factor
J9		End of Stored Values

REAL 4 Variables (Array 04)

<u>Pointer Name</u>	<u>Array Name</u>	<u>Array Contents</u>
M1	OHZC	Z Coordinates of Streamwise Element Boundaries
M2	OHRC	R Coordinates of Streamwise Element Boundaries
M3	OVZC	Z Coordinates of Transverse Element Boundaries
M4	OVRC	R Coordinates of Transverse Element Boundaries
M5	OHZC1	Z Coordinates of Streamwise Element Boundaries
M6	OHRC1	R Coordinates of Streamwise Element Boundaries
M7	OZC	Real 4 Z Coordinates of Mesh
M8	ORC	Real 4 R Coordinates of Mesh
M9		End of Stored Values

INTEGER 4 Variables (Array I2)

<u>Pointer Name</u>	<u>Array Name</u>	<u>Array Contents</u>
L1	NSEC	Number of Columns per Super Element
L2	NTSE	Super Element Type
L3	NTE	Element Type Identifier
L4	NODE	Connectivity Matrix
L5	NBC	Boundary Condition Nodes
L6		End of Stored Values

APPENDIX B

LISTING OF THE OUTPUT VARIABLES FOR THE PROGRAM MESHGEN

<u>FORTRAN Name</u>	<u>Variable</u>	<u>File</u>
ZC	Z Coordinate	TMESH DATA
RC	R Coordinate	TMESH DATA
B	Blockage Factor	TMESH DATA
PRESS	Inlet Static Pressure	FLUID DATA
PTOT	Inlet Total Pressure	FLUID DATA
TEMP	Inlet Static Temperature	FLUID DATA
TTOT	Inlet Total Temperature	FLUID DATA
RHOSTA	Inlet Static Density	FLUID DATA
RHOTT	Inlet Total Density	FLUID DATA
WDOT	Inlet Mass Flow	FLUID DATA
CP	Specific Heat	FLUID DATA
R	Gas Constant	FLUID DATA
G		FLUID DATA
RPM	RPM	FLUID DATA
VZI	Inlet Axial Velocity	FLUID DATA
NODE	Connectivity Matrix	NODE DATA
NTE	Element Type ID	NODE DATA

NBC

Boundary Nodes
Where ψ Specified

BC DATA

PSI

Stream Function

STREAM DATA

APPENDIX C

STORAGE ALLOCATION FOR THE PROGRAM TURBO

The following list of pointers and variable names indicates the storage location of the first value of the corresponding array:

REAL 8 Variables (Array R8)

<u>Pointer Name</u>	<u>Array Name</u>	<u>Array Contents</u>
NP1	ZC	Z Coordinates
NP2	RC	R Coordinates
NP3	B	Blockage Factors
NP4	ALP	Absolute Flow Angles
NP5	BE	Relative Flow Angles
NP6	H	Total Enthalpy
NP7	HS	Static Enthalpy
NP8	VZ	Axial Velocity
NP9	VR	Radial Velocity
NP10	VU	Absolute Tangential Velocity
NP11	WU	Relative Tangential Velocity
NP12	PSI	Current Streamfunction Values
NP13	PSIO	Previous Streamfunction Value
NP14	F	Right-hand Side Vector
NP15	RHS	Temporary Storage for F
NP16	RHO	Static Density

NP17	RHON	Not Used
NP18	WRL	Angular Momentum Vector
NP19	ETA	Adiabatic Efficiencies
NP20	EM	Stiffness Matrix Elements
NP21	DEV1	Rotor/Stator Deviation Angles
NP22	PRAT	Total-to-Total Pressure Ratio
NP23	TEMP	Static Temperature
NP24	TTOT	Total Temperature
NP25	PRESS	Static Pressure
NP26	PTOT	Total Pressure
NP27	RHOTT	Total Density
NP28	HR	Rhothalpy
NP29	ENTROP	Entropy
NP30		End of Stored Values

REAL 4 Variables (Array 04)

<u>Pointer Name</u>	<u>Array Name</u>	<u>Array Contents</u>
NPO1	OVEL	Velocity at a Given Station
NPO2	ORC	R Coordinates
NPO3	OBE	Relative Flow Angles
NPO4	OALP	Absolute Flow Angles
NPO5		End of Stored Values

INTEGER 4 Variables (Array I2)

<u>Pointer Name</u>	<u>Array Name</u>	<u>Array Contents</u>
NPI1	NFS	Nodes for F Calculation
NPI2	NBC	Boundary Nodes
NPI3	NTE	Element Type Identifier
NPI4	NODE	Connectivity Matrix
NPI5		End of Stored Values

APPENDIX D

LISTING OF THE OUTPUT VARIABLES FOR THE PROGRAM TURBO

<u>Listing Name</u>	<u>Variable</u>	<u>Units</u>
PSI	Stream Function	lbm/sec
VZ	Axial Velocity	ft/sec
VR	Radial Velocity	ft/sec
R	Radius	inches
DENSITY	Static Density	lbm/ft ³
WRL	Angular Momentum	ft ² /sec
HT	Total Enthalpy	BTU/lbm
VT	Absolute Tangential Velocity	ft/sec
WT	Relative Tangential Velocity	ft/sec
HS	Static Enthalpy	BTU/lbm
TEMP	Static Temperature	°R
TTOT	Total Temperature	°R
PRESS	Static Pressure	psia
PTOT	Total Pressure	psia
RHOT	Total Density	lbm/ft ³
ALPHA	Absolute Flow Angle	Degrees
BETA	Relative Flow Angle	Degrees

LISTING OF THE PROGRAM MESHGEN

```

C *****
C                                     PROGRAM MESHGEN
C *****
C
C   THIS PROGRAM PRODUCES A RECTANGULAR EIGHT-NODED
C   ISOPARAMETRIC GRID OF ARBITRARY DIMENSION M X N GIVEN
C   THE COORDINATES OF THE MESH'S "SUPER ELEMENTS".
C   THE PROGRAM ALSO COMPUTES AN INITIAL STRESS FUNCTION
C   DISTRIBUTION, TANGENTIAL BLOCKAGE FACTORS AND THE
C   APPROPRIATE INLET CONDITIONS NEED FOR THE PROGRAM TURBO.
C   THE PROGRAM IS ONLY LIMITED BY THE SPACE ALLOCATED IN
C   THE ARRAYS R9, C4, IZ OR MACHINE LIMITATIONS.
C *****
C
C   IMPLICIT REAL*(A-G,P-Z), REAL*(4)
C   INTEGER*4 NR,MRR1,NC,NCL1,IC,MRR1,NE1,MN1,MN2,MN3,MN4,NSE2,N2
C   INTEGER*4 IANS
C   COMMON /INT4/ NR,MRR1,NC,NCL1,IC
C   COMMON /ACOUNT/ MRCW,MRTW1,MNSE
C   COMMON /ACCENT/ NA,NCL,NCL1,NE,NRUTC,NSTAT
C   COMMON /APPOINT/ J1,J2,J3,J4,J5,J6,J7,J8,J9,J10,J11,J12,J13,J14,
C   IL1,L2,L3,L4,L5,L6,M1,M2,M3,M4,M5,M6,M7,M8,M9
C   COMMON /FAR/ FH(5000)
C   COMMON /FEAR/ FZ(5000)
C   COMMON /CFAR/ C4(3000)
C   LMR = 5000
C   LMI = 3000
C   LIM4 = 3000
C
C   CALL INIT1(MRCW2,ACCL2,NSE,NSE2,WDOT,PTCT,TTCT,FPM,G,R,CP,ZMAX,
C   IZMIN,RMAX,PMIR,IANS)
C
C   CALL INPLT(R8(J1),R8(J2),I2(L1),I2(L2),NE1,NSE2,NSE,MRR1,N2,ILIM,
C   LIM4,IANS)
C
C   CALL TMESH(R8(J1),R8(J2),R8(J3),R8(J4),R8(J5),R8(J6),I2(L1),
C   I2(L2),I2(L3),R8(J13),NSE2,NSE,I2,MRR1,IANS)
C
C   CALL CONEC(I2(L2),I2(L3),I2(L4),NSE,N2,NE1,NRUTC,NRUTC,
C   INSTAT,NSTATE,IANS)
C
C   CALL INIT2(R8(J6),I2(L4),I2(L5),R8(J7),RHSTA,PHOT,TTCT,WDOT,
C   IPTCT,PRESS,TEMP,VZI,K,CP,G,NSE2,MRR1,N2,NSE,IG1,IANS)
C
C   IF(IANS.EC.2) GOTO 100
C
C   CALL TASK1(R8(J6),R8(J13),NSE2,MRR1,N2,NRUTC,NRUTC,NSTATE,
C   INSTATE)
C
C   CALL FILGEN(R8(J5),R8(J6),R8(J13),R8(J7),I2(L4),I2(L3),I2(L5),
C   IPRESS,PTCT,TEMP,TTCT,RHSTA,PHUT,VZI,WDOT,KP,CP,P,G,NRUTC,
C   NSTATB,NSE2,NSE,N2,MRR1,NE1,IANS)
C
C   WRITE(15,200)
C   FORMAT('X',' DO YOU WANT A PLOT OF THE MESH ?',/, ' 1 = YES
C   1 2 = NO')

```


FILE: MESNGEN FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

C
  IMPLICIT REAL*8(A-H,P-Z)
  INTEGER*4 NR,MF1,NC,NC1,IC,MRR1,NE1,MN1,MN2,MN3,MN4,ASE2,N2
  INTEGER*4 IANS
  COMMON /INT4/ NR,MF1,NC,NC1,IC
  COMMON /NCCOUNT/ MROW,MROW1,MNSEC
  COMMON /ACCUAT/ AN,ACCL,ACCL1,AE,MROTC,NSTAT
  COMMON /NPOINT/ J1,J2,J3,J4,J5,J6,J7,J8,J9,J10,J11,J12,J13,J14,
111,L2,L3,L4,L5,L6,M1,M2,M3,M4,M5,M6,M7,M8,M9
  DIMENSION XSE(1),YSE(1)
  DIMENSION NSEC(1),NTSE(1)

C
  IF(IANS.EQ.2) GOTO 202
  ACCL = 0
  J = 1
  NSE22 = ASE2/2
  DO 100 I = 1, NSE22
    WRITE(15,101) I
    FOR MAT(5X,' ENTER 2,8 COORDINATE PAIRS FOR STATION ',I2//)
101   READ(15,*) XSE(J),YSE(J),XSE(J+1),YSE(J+1)
    J = J + 2
100  CONTINUE
  DO 200 I = 1, ASE
    WRITE(15,201) I
    FOR MAT(5X,' ENTER TYPE OF SUPER ELEMENT AND THE NO OF COLUMNS F
201 1OR SUPER ELEMENT ',I2//)
    READ(15,*) NTSE(I),NSEC(I)
    ACCL = ACCL + NSEC(I)
200  CONTINUE
202  NE = MROW + ACCL
  NE1 = NE
  ACCL1 = 2 * ACCL + 1
  MN = ACCL1 + MROW1 - MROW * ACCL
  MN1 = MN
  MRR1 = MROW1
  J1 = 1
  J2 = J1 + ASE2
  J3 = J2 + ASE2
  J4 = J3 + MRR1
  J5 = J4 + MRR1
  J6 = J5 + N2
  J7 = J6 + N2
  J8 = J7 + N2
  J9 = J8 + N2
  J10 = J9 + N2
  J11 = J10 + N2
  J12 = J11 + N2
  J13 = J12 + N2
  J14 = J13 + N2
  L1 = 1
  L2 = L1 + ASE
  L3 = L2 + ASE
  L4 = L3 + NE1
  L5 = L4 + 3*NE1
  L6 = L5 + NE1
  M1 = 1
  M2 = M1 + ACCL1
  M3 = M2 + ACCL1
  M4 = M3 + MRR1
  M5 = M4 + MRR1
  M6 = M5 + ACCL1
  M7 = M6 + ACCL1
  M8 = M7 + N2
  M9 = M8 + N2
  NEXCR = LIVR - J14
  NEXCI = LIVI - L6
  NEXC4 = LIV4 - M9
  IF(NEXCR.LT.0) CALL SPRI(NEXCR)
  IF(NEXCI.LT.0) CALL SPRI(NEXCI)
  IF(NEXC4.LT.0) CALL SPRI(NEXC4)
  WRITE(6,200) NEXC4,NEXCI

```

FILE: MESHGEN FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

300 FORMAT(10X,' MEMORY SPACE AVAILABLE : REAL =',15.2X,' INTEGER = '
1,15.2X)
RETURN
END

```

C
C
C      SUBROUTINE TRESH(XSE,YSE,XXSE,YYSE,ZC,RC,NSEC,NTSE,NTE,
1B,NSEC2,NSEC,N2,MRCW,IANS)
C      *****
C      *****
C      ** THIS SUBROUTINE PRODUCES A RECTANGULAR EIGHT-NODED **
C      ** ISOPARAMETRIC GRID OF ARBITRARY DIMENSION M X N GIVEN **
C      ** THE CORNER NODES OF THE MESH'S "SUPER ELEMENTS" **
C      *****
C      *****
C      IMPLICIT REAL*8(A-H,F-Z)
C      INTEGER*4 MRCW,NSEC,N2,MRCW,IANS
C      COMMON /INT4/ MR,MRI,NC,NC1,IC
C      COMMON /NCCOUNT/ N,NCOL,MROW
C      COMMON /NCCOUNT/ N,NCOL,NCOL1,NE,IPJTC,NSTAT
C      DIMENSION XSE(1),YSE(1),XXSE(1),YYSE(1),ZC(1),RC(1),B(1)
C      DIMENSION NSEC(1),NTSE(1),NTE(1)
C
C      IF(IANS.EQ.1) GOTO 20
C      DO 11 I=1,M
C      READ(20,12) ZC(I),RC(I),B(I)
12  FORMAT(3F15.11)
11  CONTINUE
C      GOTO 100
20  YCIV1 = FLCAT(MR1)
20  YCIV2 = FLCAT(MR2)
I = 1
IJ = 1
IL = 1
IM = 1
NSTAT = C
NFCIO = C
DO 10 N = 1,NA
ZC(N) = 0.0C
RC(N) = 0.0C
10  CONTINUE
CC 100 J = 1,NSEC
IF(J.EQ.1) GOTO 120
IF(NTSE(J)-NTSE(J-1)) 120,120,140
140 IF(IM.LT.3) GOTO 150
NROTC = IL
IM = C - IM
GOTO 120
150 NSTAT = IL
IM = C - IM
120 IT = NSEC(J) + MRCW
DO 110 L = 1,IT
NTE(IL) = NTSE(J)
IL = IL + 1
110 CONTINUE
NRCOL1 = 2 + NSEC(J) + 1
NC1 = NRCOL1
NC = NC1 - 1
XCIV1 = FLCAT(NC1)
XCIV2 = FLCAT(NC)
X1 = XSE(1)
X2 = XSE(1 + 1)
X3 = XSE(1 + 3)
X4 = XSE(1 + 5)
Y1 = YSE(1)
Y2 = YSE(1 + 1)
Y3 = YSE(1 + 2)

```

```

Y4 = YSE(1 + 3)
XDIF2 = X3 - X1
XDIF3 = X4 - X1
YDIF2 = Y1 - Y3
YDIF3 = Y2 - Y4
XINT2 = XDIF2 / XDIV2
XINT3 = XDIF3 / XDIV2
YINT2 = YDIF2 / XDIV2
YINT3 = YDIF3 / XDIV2
ISEL = -1
DO 200 K = 1, NCL
  XCIF1 = X1 - X2
  YCIF1 = Y1 - Y2
  IF (ISEL) 210, 210, 220
  XINT1 = XCIF1 / YDIV1
  YINT1 = YCIF1 / YDIV1
  MRR = MRCW1
  GOTO 230
210 XINT1 = XDIF1 / YDIV2
  YINT1 = YDIF1 / YDIV2
  MRR = MRCW + 1
220 DC 200 IC = 1, MRR
  IC1 = IC - 1
  XXSE(IC) = X1 - FLOAT(IC - 1)*XINT1
  IF (IC.NE.1) GOTO 231
  YYSE(IC) = Y1
  GOTO 232
231 YYSE(IC) = DSQRT(YYSE(IC1)*YYSE(IC1) - YINT1)
232 ZC(IJ) = XXSE(IC)
  RC(IJ) = YYSE(IC)
  IJ = IJ + 1
200 CONTINUE
  X1 = X1 + XINT2
  X2 = X2 + XINT3
  Y1 = Y1 - YINT2
  Y2 = Y2 - YINT3
  ISEL = -ISEL
200 CONTINUE
  ISEL = -ISEL
  IJ = IJ - MRCW1
  I = 2*IJ + 1
100 CONTINUE
  WRITE(6,400) MRCW, NCOL
400 FORMAT(10X, 'NUMBER OF ROWS = ', I3, 10X, 'NUMBER OF COLUMNS = ',
1, I3/)
  WRITE(6,410) NE, AN
410 FORMAT(5X, 'TOTAL NUMBER OF ELEMENTS = ', I4, 10X,
1, 'TOTAL NUMBER OF NODES = ', I4//)
  RETURN
  END

C
C
SUBROUTINE CONEC(NTSE, NTE, NNODE, NSE, N2, NE1, NPROT, NROTC, NSTATB,
1, NSTATE, IANS)
C
C *****
C
C THIS SUBROUTINE PRODUCES THE CONNECTIVITY MATRIX
C
C *****
C
IMPLICIT REAL*8(D-H, O-Z)
INTEGER*4 N2, MRR, NC, NCL, IC, MRR1, NE1, NAL, MN2, MN3, N4, NSE2, N2
INTEGER*4 IANS
COMMON /ACOUNT/ MRCW, MRCW1, NASEC
COMMON /ACOUNT/ NA, NCOL, NCOL1, NE, NROTC, NSTAT
COMMON /INT4/ MRR, NAL, NC, NCL, IC
DIMENSION NNODE(N, 1, 1), NTSE(1), NTE(1)
IF (IANS.EC.2) GOTO 601
K = 1
MK = 1

```

```

NFACTO = C
NFACTO = C
NSTATR = C
NSTATE = C
C = 1.00
CC 40 IC = 1,NEL
K1 = K + MRCW1 + MRCW + 1
K2 = K1 - MRCW - MK
K3 = K
K4 = K + 1
K5 = K + 2
K6 = K2 + 1
K7 = K1 + 2
K8 = K1 + 1
NODE(IC,1) = K1
NODE(IC,2) = K2
NODE(IC,3) = K3
NODE(IC,4) = K4
NODE(IC,5) = K5
NODE(IC,6) = K6
NODE(IC,7) = K7
NODE(IC,8) = K8
K = K + 2
MK = MK + 1
CO = FLCAT(IC)/FLCAT(MR)
IF(CO.NE.0) GOTO 40
K = K + MR + 2
MK = 1
C = C + 1.00
40 CONTINUE
64 WRITE(6,64)
FORMAT(' ')
WRITE(6,64)
IF(NPOTC.EQ.0.AND.NSTAT.EQ.0) GOTO 600
IJK1 = NFACTO
IJK2 = NFACTO + MRCW - 1
IJK3 = NSTAT
IJK4 = NSTAT + MRCW - 1
NFACTO3 = NODE(IJK1,3)
NFACTO = NODE(IJK2,7)
NSTATR = NODE(IJK3,3)
NSTATE = NODE(IJK4,7)
CONTINUE
GOTO 600
601 CC 700 I = 1, NEL
RFAC(IG,7IG) = ACCE(I,1),ACCE(I,2),ACCE(I,3),ACCE(I,4),
1 ACCE(I,5),ACCE(I,6),ACCE(I,7),ACCE(I,8),ITE(I)
710 FORMAT(9F5)
700 CONTINUE
600 RETURN
END
CCC
SUBROUTINE INIT2(IC,NODE,NBC,PSI,KHSTA,SHOTT,ITOT,WDET,PTOT,
1 PRESS,TENF,VZI,Z,CP,C,NS,E2,MRR1,MR,NSE,NEL,IENS)
+++++
++ THIS SUBROUTINE COMPUTES THE INLET CONDITIONS ++
++ THROUGH THE USE OF SUBROUTINE PLPACT. ++
++ SUBROUTINE ALSO DETERMINES THE NODE NUMBERS ++
++ WHERE THE VALUES OF PSI ARE TO BE SPECIFIED. ++
++ THE SUBROUTINE COMPUTES THE INITIAL DISTRIBUTION ++
++ OF THE STREAM FUNCTION FROM THE INLET CONDITIONS ++
++ AND THE BOUNDARY CONDITIONS OF THE PROGRAM TURBO. ++
+++++
IMPLICIT REAL*8(A-H,P-Z)
INTEGER*4 MR,MRI,MC,NC1,IC,MRR1,NEL,MN1,MN2,MN3,MN4,NSE2,NZ

```

FILE: MESFGEN FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

INTEGER*4 IANS
COMMON /ACCOUNT/ MRCH,MROW1,MNSEC
COMMON /ACCOUNT/ AN,NCOL,NCOL1,NE,NROTC,NSTAT
COMMON /INT4/ MR,MPI,NC,NC1,IC
DIMENSION FC(1),PSI(1)
DIMENSION NODE(NEL,1),NBC(1)
G = 1.4CC
R = 53.3CC
CC = 32.17400
IF(IANS.EC.1) GOTO 10
NAPC = 2 * NCOL1 + MROW1 - 2
CC 700 I = 1,NMHC
      READ(25,710) NEC(I)
      FORMAT(15)
71C
70C  CCNTINUE
      GOTO 300
10  CC 100 I = 1,MRR1
      NBC(I) = I
10C  CCNTINUE
      MN = 1
      MK = 1
      K = MRR1 + 1
      JK = K
      L = MRP1 + 1
      M = 2 * (NCOL1 - 1) + K
      CC 200 J = K,M
            IF(MM.GT.0) GOTO 210
            NBC(K) = L
            K = K + 1
            L = L + 1
            MM = C - MM
21C  GOTO 200
            IF(MK.GT.0) GOTO 220
            NBC(K) = L
            L = L + MRCH1 - 1
            MK = C - MK
            K = K + 1
            MM = C - MM
22C  GOTO 200
            NBC(K) = L
            L = L + 4RCH
            MK = C - MK
            K = K + 1
            MM = C - MM
20C  CCNTINUE
      K = MRR1 + 2 * (NCOL1 - 2) + 1
      JK = N2 - MRR1 + 1
      NBC(K) = J
      NBC(K+1) = N2
300  CCNTINUE
      GM1 = G - 1.00
      GM11 = 1.00 / GM1
      RFLB2 = FC(MRCH1) * FC(MROW1)
      ARF1 = 3.141592654 * ( RC(1) * RC(1) - RHUR2) / 144.000
      CALL FLCFCT(WOCT,PTCT,TTCT,RHCTT,VTD1,XVEL,CP,R,GM1,GM11,
1  AFEA)
      CLANT = 1.000 - XVEL * XVEL
      TEMP = TTCT * CLANT
      PRESS = PTCT * CLANT** (G*G+1)
      RHCSTA = RHCTT * CLANT**GM11
      VZ1 = VTCT * XVEL
      NC = MRCH1 - 1
      PSI1 = WOCT / (2.000 * 3.141592654)
      PSID = PSI1 / FLCAT(NC)
      J = 0
      CC 500 I = 1,NC
            PSI(I) = PSI1 - FLCAT(J) * PSID
            J = J + 1
500  CCNTINUE
      PSI(MRCH1) = 0.0000000000000000
      CC 600 I = 1,NEL
            NII = NODE(I,1)

```

FILE: MESHGEN FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

      NI2 = NCDE(1,2)
      NI3 = NCDE(1,3)
      NI4 = NCDE(1,4)
      NI5 = NCDE(1,5)
      NI6 = NCDE(1,6)
      NI7 = NCDE(1,7)
      NI8 = NCDE(1,8)
      PSI(NI1) = PSI(NI3)
      PSI(NI2) = PSI(NI3)
      PSI(NI8) = PSI(NI4)
      PSI(NI6) = PSI(NI5)
      PSI(NI7) = PSI(NI5)
400  CONTINUE
      RETURN
      END

C
C
C
      SUBROUTINE ERR1(NEXCR)
      NEXCR = -NEXCR
      WRITE(6,100) NEXCR
100  FORMAT(' EXCEEDED MAXIMUM ALLOWABLE SPACE FOR REAL*8 VARIABLES BY
      1',I5//)
      STOP
      END

C
C
C
      SUBROUTINE ERR2(NEXCI)
      NEXCI = -NEXCI
      WRITE(6,100) NEXCI
100  FORMAT(' EXCEEDED MAXIMUM ALLOWABLE SPACE FOR INT*2 VARIABLES BY
      1',I5//)
      STOP
      END

C
C
C
      SUBROUTINE ERR3(NEXC4)
      NEXC4 = -NEXC4
      WRITE(6,100) NEXC4
100  FORMAT(' EXCEEDED MAXIMUM ALLOWABLE SPACE FOR INT*4 VARIABLES BY
      1',I5//)
      STOP
      END

C
C
C
      SUBROUTINE TASK1(RC,R,NSE2,NR4,1,N2,NROTOR,NROTC,NSTAT,NSTATE)
      *****
      *****
      *****
      THIS SUBROUTINE COMPUTES THE TANGENTIAL FLOCKAGE
      FACTORS FOR THE ROTOR AND STATOR OF THE NASA TASK1
      COMPRESSOR.
      *****
      *****
      *****
      IMPLICIT REAL*8(A-H,P-Z)
      INTEGER*4 NR,MF1,AC,AC1,IC,MR1,NE1,MN1,MN2,MN3,MN4,NSF2,N2
      COMMON /ACCOUNT/ MR,MF1,AC,AC1,IC,MR1,NE1,MN1,MN2,MN3,MN4,NSF2,N2
      COMMON /ACCOUNT/ NR,AC,AC1,AC1,NE,NROTC,NSTAT
      COMMON /INT2/ NR,MF1,AC,AC1,IC
      DIMENSION FC(1),P(1)
      IF(IANS.EC.2) GOTO 700
      MN1 = NROTC + MR1 - 1
      MN2 = MN1 + 1
      MN3 = MN2 + MR1
      MN4 = MN3 + 1
      MN5 = NSTAT + MR1 - 1
      MN6 = MN5 + 1

```

FILE: MES-GEN FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

MM7 = MM6 + MRCW
MME = MM7 + 1
CC 100 I = 1,MM
E(I) = 1.000
100 CONTINUE
CC 200 I = NFCTCE,MM1
RCI = RC(I)
RTCL = 0.0131143 - 3.49672D-4 * RCI - 1.03091D-5 * RCI * RCI +
14.24339D-10 * RCI * RCI * RCI * RCI * RCI * RCI
RSIGL = 1.04973 + 1.22251D-3 * RCI * RCI + 1.46066D-8 * RCI * RCI *
1 RCI * RCI * RCI - 1.38865D-3 * DCOS(RCI) - 7.42494D-4 * DSIN(RCI)
2 + 2.395621D-5 * DTAN(RCI) - 2.12971 * DLOG(RCI)
B(I) = 1.00 - RTCL * RSIGL
200 CONTINUE
CC 300 I = MM2,MM3
RCI = RC(I)
FTCM = 0.140493 - 5.89061D-3 * RCI - 3.28044D-11 * RCI * RCI * RCI
1 * RCI * RCI * RCI * RCI + 2.15909D-15 * RCI * RCI * RCI * RCI * RCI *
2 RCI * RCI * RCI * RCI * RCI - 4.55276D-4 * DCOS(RCI) - 1.2162D-4 *
3 DSIN(RCI) - 3.84042D-6 * DTAN(RCI)
RCI = RC(I)
RSIGM = 10.9594 + .255785 * RCI - 2.6207D-6 * RCI * RCI * RCI *
1 RCI + 4.45133D-3 * DSIN(RCI) - 1.04378D-4 * DTAN(RCI)
2 - 4.83569 * DLOG(RCI)
E(I) = 1.00 - FTCM * RSIGM
300 CONTINUE
CC 400 I = MM4,NFCTCE
RCI = RC(I)
RTCT = 0.0199726 - 1.11132D-3 * RCI + 1.3479D-6 * RCI * RCI * RCI -
1 1.06833E-14 * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI
2 + 8.41826D-5 * DCOS(RCI) + 2.68422D-5 * DSIN(RCI) - 5.72139D-7 *
3 DTAN(RCI)
RSIGT = 17.9259 + .656044 * RCI - 3.56691D-6 * RCI * RCI * RCI *
1 RCI + 1.65520D-3 * DSIN(RCI) - 1.35477D-4 * DTAN(RCI)
2 - 5.50869 * DLOG(RCI)
E(I) = 1.00 - RTCT * RSIGT
400 CONTINUE
CC 500 I = NSTATE,MM5
RCI = RC(I)
STMCL = 0.013089 + 2.31157D-3 * RCI + 1.60096D-5 * RCI * RCI -
1 7.69643D-5 * DCOS(RCI) + 4.83218D-5 * DSIN(RCI) + 3.84003D-7 *
2 DTAN(RCI)
SSIGL = 4.77577 - 0.350357 * RCI + 9.45492D-3 * RCI * RCI -
1 5.52663D-13 * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI
2 - 1.31164D-3 * DSIN(RCI) - 1.72749D-5 * DTAN(RCI)
STEML = 0.272602 - 5.02566D-3 * RCI + 1.9943D-11 *
1 RCI * RCI * RCI * RCI * RCI * RCI + 3.06062D-4 * DCOS(RCI)
2 + 2.25631D-4 * DSIN(RCI) - 0.87345D-5 * DTAN(RCI)
B(I) = 1.000 - STEML * STMCL * SSIGL
500 CONTINUE
CC 600 I = MM6,MM7
RCI = RC(I)
STNCM = 0.0158762 + 2.58621D-3 * RCI + 9.85191D-6 * RCI * RCI -
1 7.56625E-5 * DCOS(RCI) + 6.89499D-5 * DSIN(RCI) - 4.16541E-6 *
2 DTAN(RCI)
SSIGM = 5.17732 - .401549 * RCI + 1.12424D-2 * RCI * RCI -
1 6.59085D-13 * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI
2 2.61139E-4 * DTAN(RCI)
B(I) = 1.000 - STNCM * SSIGM
600 CONTINUE
CC 700 I = MM8,NSTATE
RCI = RC(I)
STNCT = 0.00994821 + 3.14858D-3 * RCI - 3.60047D-11 * RCI * RCI * RCI
1 * RCI * RCI * RCI + 1.80829D-5 * DCOS(RCI) - 3.03125D-5 * DSIN(RCI) -
2 1.39128E-5 * DTAN(RCI)
SSIGT = 5.50125 - .505055 * RCI + 1.52234D-2 * RCI * RCI -
1 2.16170E-11 * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI
2 + 1.07797D-4 * DTAN(RCI)
STETMT = 0.515185 + 3.5254D-4 * RCI * RCI - 6.34D-14 * RCI *
1 RCI * RCI * RCI * RCI * RCI * RCI * RCI * RCI + 1.64098D-3 * DCOS(RCI)
2 - 0.194717 * DLOG(RCI)
B(I) = 1.000 - STETMT * STNCT * SSIGT

```

700 CONTINUE
RETURN
END

```

      SLEPCUTINE FLOCCT(WDLT,PTOT,TTOT,RHOTT,VTOT,XVEL,C,P,GH1,GYII,
      1AREA)
      +-----+
      THIS COMPUTES THE NONDIMENSIONAL VELOCITY X
      +-----+

```

180

```

100  IT = 0
      XVEL = C.100
      EPS = 1.0D-06
      RJ = 778.2 * 32.174
      RHCTT = (FJIT * 144.000)/(R * TTOT)
      VTCT = CSCRAT(2.000 * CP * RJ * TTCT)
      PH11 = WCTT / (RHCTT * VTCT * APEA)
      PHI = XVEL * (1.000 - XVEL * XVEL)**GM11
      C1FF = CABS(PH11 - PHI)
      IF(C1FF.LT.EPS) GOTO 200
      CPH1DX = PHI*(1.00/XVEL - (2.00*XVEL)/(GM1*(1.000 - XVEL*XVEL)))
      XVEL = XVEL + (PH11 - PHI) / DPH1DX
      IT = IT + 1
      IF(IT.LT.21) GOTO 100
110  WJITF(5,110)
200  FCFMAT(' CONVERGENCE NOT REACHED',/)
      CCATINUE
      VZI = XVEL * VTCT
      RETURN
      ENCL

```

מה

✓PL030:

U

THIS SUBROUTINE CREATES A TETRACNIX 418 PLOT OF EIGHT-NODE ISOPARAMETRIC ELEMENT MESH.

```

      IMPLICIT REAL*8(A-H,P-Z) , REAL*4(C)
      INTEGER*4 MP, NR1, NC, NC1, IC, NE1, N14, NR14, NC14, NC11
      LOGICAL*1 TITL1
      COMMON /FFAR/ FB(5000)
      COMMON /NPAS/ I2(2000)
      COMMON /CFAP4/ Q4(3000)
      COMMON /INT4/ MP, NR1, NC, IC
      COMMON /NCCUNT/ NROW, ARDW1, MNSEC, LIMI, LIMP, LIM4
      COMMON /NCCUNT/ NN, NCL1, NCCL1, NE, NE1, NSTAT
      DIMENSION CHZC(1), CHRC(1), CVZC(1), CVRC(1), CXYL(4)
      DIMENSION CHZC1(1), CHRC1(1), CZC(1), DRC(1), DPS(10)
      DIMENSION ZC(1), PC(1)
      DIMENSION KLK(1)
      DIMENSION TITL1(37)
      DATA TITL1/'EIGHT NODE ISOPARAMETRIC ELEMENT MESH

```

[illegible]

```

CONVERT THE NODAL COORDINATES TO REAL*4
TO MAKE THEM COMPATIBLE WITH PLOTG CALLS

```

3

```
CC 30 I = 1, MN
OZC(I) = ZC(I)
CSC(I) = SC(I)
```

30

```

CXYL(1) = ZMIN
CXYL(2) = ZMIN
CXYL(3) = ZMAX
CXYL(4) = ZMAX
CALL DSEINIT
CALL DSEFE

```

33

PLCT NOAL POINTS

```
CC 5 I = 1, NCC L1
C+ZC(1) = C.C
```

FILE: MESHGEN FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

      CHRC(1) = 0.0
5      CONTINUE
      DO 6 I = 1, MFCW1
      CVZC(1) = 0.0
      CVRC(1) = 0.0
6      CONTINUE
      MM1 = MFCW1 + 1
      ACC1 = ACCL1 + 1
      A14 = NA
      MR14 = MFCW1
      AC14 = ACCL1
      AN1 = NA + 1
      MP1 = MFCW + 1
      CALL PLOT('*****NNNNL',A14,CZC,CPC,OXYL,37,TITL1)
C *****
C **          PLOT HORIZONTAL ELEMENT BOUNDARIES          **
C *****
      MM = 1
      KKK = 1
      DO 100 J = 1, ACCL1
      CHZC(J) = CZC(MM)
      CHFC(J) = CRC(MM)
      KKK(J) = NA
      IF(KKK.LT.C) GOTO 90
      MM = MM + MFCW1
      KKK = 0 - KKK
      GOTO 100
90      MM = MM + MP1
      KKK = 0 - KKK
100     CONTINUE
      CALL PLOT('*****NNNNL',AC14,CHZC,CHRC,OXYL,37,TITL1)
      DO 200 I = 1, MFCW
      MLK = 2 * I
      KKK = 1
      DO 800 II = 1, ACCL1
      IF(KKK.LT.C) GOTO 850
      III = KKK(II) + MLK
      CHZC1(III) = CZC(III)
      CHRC1(III) = CRC(III)
      KKK = 0 - KKK
      GOTO 800
850      III = KKK(II) + 1
      CHZC1(III) = CZC(III)
      CHRC1(III) = CRC(III)
      KKK = 0 - KKK
800     CONTINUE
      MM = (2 * I) + 1
      MLK = MLK + 1
      CALL PLOT('*****NNNNL',AC14,CHZC1,CHRC1,OXYL,37,TITL1)
200     CONTINUE
C *****
C **          PLOT VERTICAL BOUNDARIES          **
C *****
      MM = 1
      AC11 = ACCL + 1
      DO 400 I = 1, AC11
      DO 300 J = 1, MFCW1
      OVZC(J) = ZC(MM)
      OVFC(J) = FC(MM)
      MM = MM + 1
300     CONTINUE
      CALL PLOT('*****NNNNL',MR14,CVZC,CVPC,OXYL,37,TITL1)
      MM = MM + MP1
400     CONTINUE
      CALL OSTERM
      RETURN
      END

```

LISTING OF THE PROGRAM TURBO

```

PROGRAM TUPED

THIS MERIDIONAL THROUGH-FLOW ANALYSIS PROGRAM
APPLIES A GALERKIN FINITE ELEMENT METHOD TO A
STREAM FUNCTION FORMULATION. THE PROGRAM USES
EIGHT NODE, ISOPARAMETRIC ELEMENTS AND THREE-
POINT GAUSS-LEGENDRE QUADRATURE NUMERICAL
INTEGRATION. SELECTED RESULTS ARE DISPLAYED
ON THE TEKTRONIX 619 GRAPHICS TERMINAL.

IMPLICIT REAL*8(A-H,I-Z), REAL*4(C)
INTEGER*4 NREAD,NWRITE,IC,NACO,NE4,NNE4,NPOWS,LIMR,LIMI
DIMENSION ALL ARRAY VARIABLES

COMMON /REAL2/ R2(5000)
COMMON /REAL4/ R4(500)
COMMON /INT2/ I2(2000)
COMMON /NCOINT/ NCOL,NCOL1,NN,NE
1,NNE,NNEC,NNNEC
COMMON /ECON/ RC,G,CP,PT,TT,AG,WDOT,RHCT,RHSTA,
1LINLET,COLLET,PS11,BTJ,E212,F111,G
COMMON /NCOINT/ NCOL,NCOL1,NN,NE
COMMON /LTC/ NREAD,NWRITE
COMMON /NPOINT/ NP1,NP2,NP3,NP4,NP5,NP6,NP7,NP8,NP9,NP10,NP11,
1NP12,NP13,NP14,NP15,NP16,NP17,NP18,NP19,NP20,NP21,NP22,NP23,
2NP24,NP25,NP26,NP27,NP28,NP29,NP30,NP31,NP32,NP33,NP34,NP35,
3NP36,NP37,NP38,NP39,NP40,NP41,NP42,NP43,NP44,NP45,NP46,
DIMENSION R2(5000),R4(500),I2(2000),E(8),SF(3)
DIMENSION FOS(8),Z04(4),F41(8,3),RNB(3)
DIMENSION N(3),L1(2),L1(2)
DIMENSION TITLE(10),RJAC(2,2),RIJAC(2,2)
DATA STOP/'STOP'/

CALL RDATA(ZA,EA,W,RELX,KK,LIMI,LIMR,LIM4)

CALL INIT1(LIMI,LIMR,TITLE,NPOTOD,NSTATS,LIM4,NACO,NE4,NNE4,
1NPOWS)

CALL ZERCI(0,P(NP14),R2(NP12),R2(NP15),R2(NP3),R2(NP9)
2,R2(NP10),R2(NP23),R2(NP2),R2(NP11),R2(NP3),
3I2(NP11),I2(NP13),I2(NP15),R2(NP16),R2(NP17),R2(NP21),R2(NP7)
4,R2(NP4),R2(NP11),R2(NP5),R2(NP21),R2(NP22),R2(NP20),
5R2(NP12),I2(NP13),I2(NP14),R2(NP25),R2(NP29),R2(NP19),NACO,NE4,
6R2(NP4),NPOWS)

CALL INFL(R4(NP11),R4(NP2),R2(NP3),R2(NP4),R2(NP5),I2(NP13),
1I2(NP14),R4(NP22),R4(NP21),R2(NP4),R2(NP1),R2(NP12),R4(NP13),
2R2(NP23),R4(NP24),R4(NP25),R4(NP25),R4(NP27),R2(NP27),
3I2(NP11),I2(NP12),TITLE,NPOTOD,NACO,NE4,NACO,NPOWS)

```

CCC

CC

88

CCC

הנה

CC

١١١

2

2

2

הנהגה

CCC

5cc

C

C

C

CC

184

[illegible]

```

SUBROUTINE INITI(LIMI,LIMP,TITLE,NPOTCB,NSTATB,LIN4,NACD,NE4,
1ANF4,NRCHS)
C+++++
C+++++
C++
C++
C++
C+++++

```

```

1  IMPLICIT REAL*8(I=4, F=2)
2  INTEGER*4  NREAD, NWRITE, IC, NACO, NE4, NNE4, NROWS, LIMP, LIMI
3  COMMON /ACCOUNT/ NCOL, NCOL1, NN, NE
4  1, NNE, NNHC, NNHNC
5  COMMON /ACCOUNT/ NROW, NROW1, KK
6  COMMON /POINT/  NP1, NP2, NP3, NP4, NP5, NP6, NP7, NP8, NP9, NP10, NP11,
7  INF12, NP13, NP14, AP15, AP16, AP17, NP18, NP19, NP20, NP21, NP22, NP23,
8  NP24, NP25, NP26, NP27, NP28, NP29, NP30, NP31, NP32, NP33, NP34, NP35,
9  NPF01, NPF02, NPF03, NPF04, NPF05, NPF06
10 COMMON /LIC/ NREAD, NWRITE
11 DIMENSION TITLE(10)

```

```

C      READ(INREAD,100)TITLE
10C      FORMAT(10A4)

```

READ IN NUMBER OF PAGES AND NUMBER OF ELEMENTS

```

200 READ(25,200) NN,ARCH,NCCL,MPCAL,NCCL1,NRCTCB,NSTATB
      FORMAT(7IS)

```

```

AE = 3
AE = 4 * ACCL
ACD = AE
AE4 = AE
AE4 = AE * APCW + 1
AP1 = 1
AP2 = 1
AP3 = 1
AP4 = 1
AP5 = 1
AP6 = 1
AP7 = 1
AP8 = 1
AP9 = 1
AP10 = 1
AP11 = 1
AP12 = 1
AP13 = 1
AP14 = 1
AP15 = 1
AP16 = 1
AP17 = 1
AP18 = 1
AP19 = 1
AP20 = 1

```

[illegible]

CC

00

```

E(3) = (-Z1 + 2.00*E1 + Z1*Z1 - 2.00*Z1*E1)/4.00
E(4) = -E1 + E1*Z1
E(5) = (Z1 + 2.00*E1 - Z1*Z1 - 2.00*Z1*E1)/4.00
E(6) = (-1.00 + Z1*Z1)/2.00
E(7) = (-Z1 + 2.00*E1 - Z1*Z1 + 2.00*Z1*E1)/4.00
E(8) = -(E1 + Z1*E1)
CC 100 I = 1,8
RJAC(1,1) = RJAC(1,1) + D(I)*ZCS(I)
RJAC(1,2) = RJAC(1,2) + D(I)*RC(I)
RJAC(2,1) = RJAC(2,1) + E(I)*ZCS(I)
RJAC(2,2) = RJAC(2,2) + E(I)*RC(I)
10C CCNTINUE
RETURN
END

```

187

```

GM1 = G - 1. CDC
ISTA(1) = 3
ISTA(2) = 2
ISTA(3) = 1
ISTA(4) = 1
ISTA(5) = 1
ISTA(6) = 2
ISTA(7) = 3
ISTA(8) = 3
CC 100 I = 1, MPOW1
H(I) = CP * (TCT(I) * BTU * GC * 144.000
HS(I) = CP * TEMP(I) * BTU * GC * 144.000
100 CC CONTINUE
CC 200 II = 1, NE
NTE1 = NTE(II)
IF(NTE1 - 2) 210, 220, 230

C
C
C 210
CC 240 M = 1, NNE
ISTA1 = ISTA(M)
CALL SLIN(FC, PSI, VZ, VE, NCDF, VM1, PC11, PC12, ALP1, NTE1, RHC1,
1RHC, ALP, EL, IT, ISTA1, V, RHOT, TEMP, TTOT, PRESS, PTOT, B, F1, T1, TT1, PT1,
2RHOT1, H1, FS1, B2, OPSIZ2, OPSIZ2, F, HS, Z1, VZ1, VR1, ZC, ENTROP, ENT1,
3NACC, NE4, NNE4, NNE4, NNE4)
IJK = AB - VRCH
IF(I1, LE, IJK) GOTO 211
IF(ISTA1, GE, 3) GOTO 211
OPSIZ2 = 0. CDC
211 CALL ELCT(FC1, PC12, T1, P1, TT1, PT1, RHOT1, H1, FS1,
2RHC1, ALP1, ALP2, OPSIZ2, OPSIZ2, ISTA1, T2, TT2, P2, PT2, RHC2, RHOT2,
3VZ1, VM2, NACC, NE4, NNE4, NNE4, NNE4)
IF(ISTA1, GE, 1) GOTO 240
NIM = NCDF(II, M)
PRESS(NIM) = P2
PTOT(NIM) = PT2
TTOT(NIM) = TT2
TEMP(NIM) = T2
H(NIM) = H1
FS(NIM) = FS1
RHC(NIM) = RHC2
RHOT(NIM) = RHOT2
ALP(NIM) = ALP2
VZ(NIM) = 144. CDC * OPSIZ2 / (RHC2 * B2 * PC12 / 12.00)
VR(NIM) = -144. CDC * OPSIZ2 / (RHC2 * B2 * PC12 / 12.00)
VM(NIM) = VM2 * DTAN(ALP2)
WFL(NIM) = RC12 * VL(NIM)
ENTROP(NIM) = ENT1
240 CONTINUE
GOTO 200

C
C
C 220
CC 250 M = 1, NNE
ISTA1 = ISTA(M)
RMAX1 = RC(NACC, M)
RMIN1 = RC(NACC, M + MPOW1 - 1)
CALL SLIN(FC, PSI, VZ, VE, NCDF, VM1, PC11, PC12, ALP1, NTE1, RHC1,
1RHC, ALP, EL, IT, ISTA1, V, RHOT, TEMP, TTOT, PRESS, PTOT, B, F1, T1, TT1, PT1,
2RHOT1, H1, FS1, B2, OPSIZ2, OPSIZ2, F, HS, Z1, VZ1, VR1, ZC, ENTROP, ENT1,
3NACC, NE4, NNE4, NNE4, NNE4)
CALL ELCT(FC1, VZ1, BET11, BET12, AMACH1, RMAX1, RMIN1,
1FCV, PC12, RHC1, RHC2, T1, T2, P1, P2, PT1, ALP1, RHOT1, RHOT2, AC, ENT, OPSIZ2,
2OPSIZ2, OFAC, ALP2, P2, PT2, T2, TT2, TONCS, VM2, RPASS, NACC, NE4, NNE4,
3NACC)
NIM = NCDF(II, M)
RFACT = (1. CDC - OPASS) * (1. CDC - RPASS)
IF(ISTA1 - 2) 251, 251, 252
PRESS(NIM) = (P2 + P1) / 2.000
PTOT(NIM) = (PT2 + PT1) / 2.000
TTOT(NIM) = (TT2 + TT1) / 2.000
TEMP(NIM) = (T2 + T1) / 2.000
251

```

```

H(NIM) = CP * TTOT(NIM) * BTU * GC * 144.000
HS(NIM) = CP * TEMP(NIM) * BTU * GC * 144.000
RHO(NIM) = (RHC2 + RHC1) / 2.000
PHOTT(NIM) = (RHCT2 + RHCT1) / 2.000
ALP(NIM) = (ALP2 + ALP1) / 2.000
BE(NIM) = (BETA2 + BETA1) / 2.000
VZ2 = 144.000 * OPSIZ2 / (RHC2 * H2 * RC12 / 12.00)
VP2 = -144.000 * OPSIZ2 / (RHC2 * H2 * RC12 / 12.00)
VZ(NIM) = (VZ1 + VZ2) / 2.000
VR(NIM) = (VR1 + VR2) / 2.000
VU(NIM) = (VM2 + VM1) * CTAN(ALP(NIM)) / 2.000
WU(NIM) = (VM2 + VM1) * CTAN(BETA2) / 2.000
WPL(NIM) = (RC11 + RC12) * VU(NIM) / 2.000
HR(NIM) = H(NIM) - WG * (RC11 + RC12) * VU(NIM) / 2.000
ENTRCP(NIM) = C.5000 * ENT
GOTO 250
252 PRESS(NIM) = P2
PTOT(NIM) = PT2
TTOT(NIM) = TT2
TEMP(NIM) = T2
DEV1(NIM) = DEV
H(NIM) = CP * TT2 * BTU * GC * 144.000
HS(NIM) = CP * T2 * BTU * GC * 144.000
RHO(NIM) = RHC2
PHOTT(NIM) = RHCT2
ALP(NIM) = ALP2
BE(NIM) = BETA2
VZ(NIM) = 144.000 * OPSIZ2 / (RHC2 * H2 * RC12 / 12.00)
VP(NIM) = -144.000 * OPSIZ2 / (RHC2 * H2 * RC12 / 12.00)
VU(NIM) = VM2 * CTAN(ALP2)
WU(NIM) = VM2 * CTAN(BETA2)
WPL(NIM) = RC12 * VU(NIM)
HR(NIM) = H(NIM) - WG * RC12 * VU(NIM)
ENTRCP(NIM) = ENT
GOTO 250
253 BE(NIM) = BETA1
WU(NIM) = VM1 * CTAN(BETA1)
HP(NIM) = H(NIM) - WG * RC11 * VU(NIM)
ETA(NIM) = (TT1 / (TT2 - TT1)) * (((PT2 / PT1) ** (GM1 / G1)) - 1.000)
PRAT(NIM) = PT2 / PT1
CONTINUE
GOTO 200
C
C
C
230 DO 260 M = 1, NRE
ISTAT1 = ISTAT(M)
RMAX1 = RC(ISTATB)
RMINI1 = RC(ISTATB) + RMAX1 - 1)
NIM = NCOE(II, M)
CALL SLIME(FC, FSI, VZ, VE, NCOE, VM1, RC11, RC12, ALP1, BETA1, PHO1,
1RHC, ALP, EI, II, ISTAT1, N, PHOTT, TEMP, TTOT, P2, SS, PTOT, P, P1, T1, T11, PT1,
2PHCT1, P1, FSI, B2, OPSIZ2, OPSIZ2, F, HS, Z1, VZ1, VR1, ZC, ENTRCP, ENT1,
3NRC, NE4, NRE4, NRC4)
IF (ISTAT1.EQ.1) GOTO 264
BETA1 = ALP1
GOTO 265
264 BETA1 = ALP(NIM)
265 CALL STAT(RC11, VM1, BETA1, BETA2, RHC2, RMAX1, RMINI1,
1DEV, RC12, PHO1, P1, P2, T1, TT1, P1, P11, ALP1, RHCT1, RHCT2, RC, ENT, OPSIZ2,
2OPSIZ2, CFAC, ALP2, P1, PT2, T2, TT2, TC, FC, VP2, RPASS, NCOE, NE4, NRE4,
3NRC4)
RFACT = (1.00 - RPASS) * (1.00 - RPASS)
IF (ISTAT1.EQ.2) GOTO 261, 262
PRESS(NIM) = (P2 + P1) / 2.000
PTOT(NIM) = (PT2 + PT1) / 2.000
TTOT(NIM) = (TT2 + TT1) / 2.000
TEMP(NIM) = (T2 + T1) / 2.000
H(NIM) = F1
HS(NIM) = H1
RHO(NIM) = (RHC2 + RHC1) / 2.000
PHOTT(NIM) = (RHCT2 + RHCT1) / 2.000

```

```

ALP(NIM) = (RETA2 + RETA1) / 2.000
VZ2 = 144.000 * OPSIR2 / (RHC2 * B2 * PCI2 / 12.00)
VP2 = -144.000 * OPSIZ2 / (RHO2 * B2 * PCI2 / 12.00)
VZ(NIM) = (VZ1 + VZ2) / 2.000
VR(NIM) = (VR1 + VR2) / 2.000
VU(NIM) = (VM2 + VM1) * ETAN(ALP(NIM)) / 2.000
ENTRCF(NIM) = C.5000 * ENT + ENT1
WRL(NIM) = PCI2 * VU(NIM)
GOTO 260
PRESS(NIM) = P2
FTOT(NIM) = PT2
TTOT(NIM) = TT2
TEMP(NIM) = T2
DFV1(NIM) = CEV
H(NIM) = F1
FS(NIM) = FS1
RHC(NIM) = RHC2
RHDT(NIM) = RHDT2
ALP(NIM) = RETA2
VZ(NIM) = 144.000 * OPSIR2 / (RHO2 * B2 * PCI2 / 12.00)
VR(NIM) = -144.000 * OPSIZ2 / (RHC2 * B2 * PCI2 / 12.00)
VU(NIM) = VM2 * ETAN(ALP2)
WRL(NIM) = PCI2 * VU(NIM)
ENTRCF(NIM) = ENT + ENT1
GOTO 260
ETA(NIM) = FT2/PT1
PRAT(NIM) = PT2/PT1
CONTINUE
C
RETURN
END

```

```

C ***** SUBROUTINE SHAPE(E,Z,SF) *****
C *****
C ***** THIS SUBROUTINE CALCULATES THE SHAPE FUNCTIONS FOR *****
C ***** LOCAL NODES READ FROM LEFT TO RIGHT, BOTTOM TO TOP *****
C ***** IN THE Z-E PLANE. *****
C ***** CALL STATEMENT DEFINITIONS: *****
C *****   Z = VALUE OF THE EXCITATION INPUT *****
C *****   E = VALUE OF THE DATA INPUT *****
C *****   SF = SHAPE FUNCTION VECTOR *****
C *****
C *****

```

```

      IMPLICIT REAL*8(A-H,P-Z)
      DIMENSION SF(8)
      SF(1) = (Z*E + Z*Z + Z*E + Z*Z*E + Z*E*E - 1.00)/4.00
      SF(2) = (1.00 + Z*E + Z*Z*E + Z*Z*E)/2.00
      SF(3) = (-Z*E + Z*E + Z*E*E + Z*Z*E) - Z*E*E - 1.00)/4.00
      SF(4) = (-1.00 + Z*E + Z*E*E + Z*Z*E)/2.00
      SF(5) = (Z*E + Z*Z + Z*E*E + Z*Z*E) - Z*E*E - 1.00)/4.00
      SF(6) = (1.00 + Z*E + Z*Z*E + Z*Z*E)/2.00
      SF(7) = (-Z*E + Z*Z + Z*E*E + Z*E*E - 1.00)/4.00
      SF(8) = (1.00 + Z*E + Z*E*E + Z*E*E)/2.00
      RETURN
      END

```

190

AD-A124 987 FINITE ELEMENT PROGRAM FOR CALCULATING FLOWS IN
TURBOMACHINES WITH RESULTS FOR NASA TASK-1 COMPRESSOR
(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA J A FERGUSON
1987 82 5/16 2814

FINITE ELEMENT PROGRAM FOR CALCULATING FLOWS IN
TURBOMACHINES WITH RESULTS FOR NASA TASK-1 COMPRESSOR
(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA J A PERGUSON
007 00 510 2834

3/3

UNCLASSIFIED

OCT 82

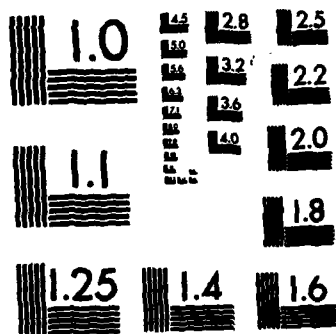
F/G 28/4

NL

END

© 2004 Blackwell Publishing Ltd

Q² =



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

FILE: TURBO FCRTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

1      ,NODE(J,5),NOCCE(J,6),NOCCE(J,7),NOCCE(J,8),NTE(J)
1010  FORMAT (9I5)
1015  CC CONTINUE
      READ IN INLET THERMODYNAMIC CLANTITIES; FLOW RATE, INLET
      U VELOCITY, OUTLET U VELOCITY, RHCT, RHCSTA, PTOT, TTCT, N
      UNITS ARE AS FOLLOWS: FLOW RATE (LBM/SEC)
      VELOCITY (FT/SEC) ; RHOT AND RHCSTA (LBM/CU FT) ;
      FTCT (PSF) ; TT ( DEGREES RANKINE) ; SPEED (RPM)
1024  READ(25,1035)PRES,PT,TEM,TT,RHCSTA,RHCT
1035  FCFMAT(4F11.6,2F12.8)
      READ IN FLUID/GAS CONSTANTS,R(GAS CONSTANT),GAMMA, CP(RTU/LBM-R)
      READ(25,1037)NCC1,CP,RG,G,SPEED,UIINLET
1037  FCFMAT(4F12.7,F9.1,F13.8)
      FIND OMEGA (RAD/SEC)
      WG = SPEED*2.00*3.14159300/60.00
      LINLET = UIINLET*12.000
      COMPLETE THE FIRST ESTIMATES OF UVEL(I) AND RHO(I).
      CC 166 I = 1,NN
      UVEL(I) = UIINLET
      PHO(I) = RHCSTA
      TEM(I) = TEM
      TTOT(I) = TT
      PRESS(I) = PRES
      PTOT(I) = PT
      RHOTT(I) = RHCT
166  CC CONTINUE
      READ NCES WHERE PSI IS SPECIFIED
      NABC = 2 * NCC11 + NROW1 - 2
      CC 190 I = 1,NABC
      READ(25,1020) ABC(I)
1020  FCFMAT(15)
190  CC CONTINUE
      READ IN THE FIRST ESTIMATE OF SYSTEM'S PSI DISTRIBUTION
      CC 103 I = 1,NN
      READ(40,1021) PSI(I)
      PSIC(I) = PSI(I)
1021  FCFMAT(F15.11)
193  CC CONTINUE
      LINLT = LINLET / 12.000
      FIND NCES WHERE F(P,Z) IS SPECIFIED
      NABC = NN - NABC
      PRINT ALL INPUT DATA
      GC TO 1101
      WRITE(NWRITE,1038)TITLE
1038  FCFMAT(17X,10A4)
      WRITE(NWRITE,1040)NN,NE
1040  FCFMAT(11X,10A4)
      WRITE(NWRITE,1041)NRC,NCC1
1041  FCFMAT(5X,10A4)
      WRITE(NWRITE,1042)NCC2,NCC3
1042  FCFMAT(5X,10A4)
      WRITE(NWRITE,1043)NCC4,NCC5
1043  FCFMAT(5X,10A4)
      WRITE(NWRITE,1044)NCC6,NCC7
1044  FCFMAT(5X,10A4)
      WRITE(NWRITE,1045)NCC8,NCC9
1045  FCFMAT(5X,10A4)
      WRITE(NWRITE,1046)NCC10,NCC11
1046  FCFMAT(5X,10A4)
      WRITE(NWRITE,1047)NCC12,NCC13
1047  FCFMAT(5X,10A4)
      WRITE(NWRITE,1048)NCC14,NCC15
1048  FCFMAT(5X,10A4)
      WRITE(NWRITE,1049)NCC16,NCC17
1049  FCFMAT(5X,10A4)
      WRITE(NWRITE,1050)NCC18,NCC19
1050  FCFMAT(5X,10A4)
      WRITE(NWRITE,1051)NCC20,NCC21
1051  FCFMAT(5X,10A4)
      WRITE(NWRITE,1052)NCC22,NCC23
1052  FCFMAT(5X,10A4)
      WRITE(NWRITE,1053)NCC24,NCC25
1053  FCFMAT(5X,10A4)
      WRITE(NWRITE,1054)NCC26,NCC27
1054  FCFMAT(5X,10A4)
      WRITE(NWRITE,1055)NCC28,NCC29
1055  FCFMAT(5X,10A4)
      WRITE(NWRITE,1056)NCC30,NCC31
1056  FCFMAT(5X,10A4)
      WRITE(NWRITE,1057)NCC32,NCC33
1057  FCFMAT(5X,10A4)
      WRITE(NWRITE,1058)NCC34,NCC35
1058  FCFMAT(5X,10A4)
      WRITE(NWRITE,1059)NCC36,NCC37
1059  FCFMAT(5X,10A4)
      WRITE(NWRITE,1060)NCC38,NCC39
1060  FCFMAT(5X,10A4)
      WRITE(NWRITE,1061)NCC40,NCC41
1061  FCFMAT(5X,10A4)
      WRITE(NWRITE,1062)NCC42,NCC43
1062  FCFMAT(5X,10A4)
      WRITE(NWRITE,1063)NCC44,NCC45
1063  FCFMAT(5X,10A4)
      WRITE(NWRITE,1064)NCC46,NCC47
1064  FCFMAT(5X,10A4)
      WRITE(NWRITE,1065)NCC48,NCC49
1065  FCFMAT(5X,10A4)
      WRITE(NWRITE,1066)NCC50,NCC51
1066  FCFMAT(5X,10A4)
      WRITE(NWRITE,1067)NCC52,NCC53
1067  FCFMAT(5X,10A4)
      WRITE(NWRITE,1068)NCC54,NCC55
1068  FCFMAT(5X,10A4)
      WRITE(NWRITE,1069)NCC56,NCC57
1069  FCFMAT(5X,10A4)
      WRITE(NWRITE,1070)NCC58,NCC59
1070  FCFMAT(5X,10A4)
      WRITE(NWRITE,1071)NCC60,NCC61
1071  FCFMAT(5X,10A4)
      WRITE(NWRITE,1072)NCC62,NCC63
1072  FCFMAT(5X,10A4)
      WRITE(NWRITE,1073)NCC64,NCC65
1073  FCFMAT(5X,10A4)
      WRITE(NWRITE,1074)NCC66,NCC67
1074  FCFMAT(5X,10A4)
      WRITE(NWRITE,1075)NCC68,NCC69
1075  FCFMAT(5X,10A4)
      WRITE(NWRITE,1076)NCC70,NCC71
1076  FCFMAT(5X,10A4)
      WRITE(NWRITE,1077)NCC72,NCC73
1077  FCFMAT(5X,10A4)
      WRITE(NWRITE,1078)NCC74,NCC75
1078  FCFMAT(5X,10A4)
      WRITE(NWRITE,1079)NCC76,NCC77
1079  FCFMAT(5X,10A4)
      WRITE(NWRITE,1080)NCC78,NCC79
1080  FCFMAT(5X,10A4)
      WRITE(NWRITE,1081)NCC80,NCC81
1081  FCFMAT(5X,10A4)
      WRITE(NWRITE,1082)NCC82,NCC83
1082  FCFMAT(5X,10A4)
      WRITE(NWRITE,1083)NCC84,NCC85
1083  FCFMAT(5X,10A4)
      WRITE(NWRITE,1084)NCC86,NCC87
1084  FCFMAT(5X,10A4)
      WRITE(NWRITE,1085)NCC88,NCC89
1085  FCFMAT(5X,10A4)
      WRITE(NWRITE,1086)NCC90,NCC91
1086  FCFMAT(5X,10A4)
      WRITE(NWRITE,1087)NCC92,NCC93
1087  FCFMAT(5X,10A4)
      WRITE(NWRITE,1088)NCC94,NCC95
1088  FCFMAT(5X,10A4)
      WRITE(NWRITE,1089)NCC96,NCC97
1089  FCFMAT(5X,10A4)
      WRITE(NWRITE,1090)NCC98,NCC99
1090  FCFMAT(5X,10A4)
      WRITE(NWRITE,1091)NCC100,NCC101
1091  FCFMAT(5X,10A4)
      WRITE(NWRITE,1092)NCC102,NCC103
1092  FCFMAT(5X,10A4)
      WRITE(NWRITE,1093)NCC104,NCC105
1093  FCFMAT(5X,10A4)
      WRITE(NWRITE,1094)NCC106,NCC107
1094  FCFMAT(5X,10A4)
      WRITE(NWRITE,1095)NCC108,NCC109
1095  FCFMAT(5X,10A4)
      WRITE(NWRITE,1096)NCC110,NCC111
1096  FCFMAT(5X,10A4)
      WRITE(NWRITE,1097)NCC112,NCC113
1097  FCFMAT(5X,10A4)
      WRITE(NWRITE,1098)NCC114,NCC115
1098  FCFMAT(5X,10A4)
      WRITE(NWRITE,1099)NCC116,NCC117
1099  FCFMAT(5X,10A4)
      WRITE(NWRITE,1100)NCC118,NCC119
1100  FCFMAT(5X,10A4)
      WRITE(NWRITE,1101)NCC120,NCC121
1101  FCFMAT(5X,10A4)
      WRITE(NWRITE,1102)NCC122,NCC123
1102  FCFMAT(5X,10A4)
      WRITE(NWRITE,1103)NCC124,NCC125
1103  FCFMAT(5X,10A4)
      WRITE(NWRITE,1104)NCC126,NCC127
1104  FCFMAT(5X,10A4)
      WRITE(NWRITE,1105)NCC128,NCC129
1105  FCFMAT(5X,10A4)
      WRITE(NWRITE,1106)NCC130,NCC131
1106  FCFMAT(5X,10A4)
      WRITE(NWRITE,1107)NCC132,NCC133
1107  FCFMAT(5X,10A4)
      WRITE(NWRITE,1108)NCC134,NCC135
1108  FCFMAT(5X,10A4)
      WRITE(NWRITE,1109)NCC136,NCC137
1109  FCFMAT(5X,10A4)
      WRITE(NWRITE,1110)NCC138,NCC139
1110  FCFMAT(5X,10A4)
      WRITE(NWRITE,1111)NCC140,NCC141
1111  FCFMAT(5X,10A4)
      WRITE(NWRITE,1112)NCC142,NCC143
1112  FCFMAT(5X,10A4)
      WRITE(NWRITE,1113)NCC144,NCC145
1113  FCFMAT(5X,10A4)
      WRITE(NWRITE,1114)NCC146,NCC147
1114  FCFMAT(5X,10A4)
      WRITE(NWRITE,1115)NCC148,NCC149
1115  FCFMAT(5X,10A4)
      WRITE(NWRITE,1116)NCC150,NCC151
1116  FCFMAT(5X,10A4)
      WRITE(NWRITE,1117)NCC152,NCC153
1117  FCFMAT(5X,10A4)
      WRITE(NWRITE,1118)NCC154,NCC155
1118  FCFMAT(5X,10A4)
      WRITE(NWRITE,1119)NCC156,NCC157
1119  FCFMAT(5X,10A4)
      WRITE(NWRITE,1120)NCC158,NCC159
1120  FCFMAT(5X,10A4)
      WRITE(NWRITE,1121)NCC160,NCC161
1121  FCFMAT(5X,10A4)
      WRITE(NWRITE,1122)NCC162,NCC163
1122  FCFMAT(5X,10A4)
      WRITE(NWRITE,1123)NCC164,NCC165
1123  FCFMAT(5X,10A4)
      WRITE(NWRITE,1124)NCC166,NCC167
1124  FCFMAT(5X,10A4)
      WRITE(NWRITE,1125)NCC168,NCC169
1125  FCFMAT(5X,10A4)
      WRITE(NWRITE,1126)NCC170,NCC171
1126  FCFMAT(5X,10A4)
      WRITE(NWRITE,1127)NCC172,NCC173
1127  FCFMAT(5X,10A4)
      WRITE(NWRITE,1128)NCC174,NCC175
1128  FCFMAT(5X,10A4)
      WRITE(NWRITE,1129)NCC176,NCC177
1129  FCFMAT(5X,10A4)
      WRITE(NWRITE,1130)NCC178,NCC179
1130  FCFMAT(5X,10A4)
      WRITE(NWRITE,1131)NCC180,NCC181
1131  FCFMAT(5X,10A4)
      WRITE(NWRITE,1132)NCC182,NCC183
1132  FCFMAT(5X,10A4)
      WRITE(NWRITE,1133)NCC184,NCC185
1133  FCFMAT(5X,10A4)
      WRITE(NWRITE,1134)NCC186,NCC187
1134  FCFMAT(5X,10A4)
      WRITE(NWRITE,1135)NCC188,NCC189
1135  FCFMAT(5X,10A4)
      WRITE(NWRITE,1136)NCC190,NCC191
1136  FCFMAT(5X,10A4)
      WRITE(NWRITE,1137)NCC192,NCC193
1137  FCFMAT(5X,10A4)
      WRITE(NWRITE,1138)NCC194,NCC195
1138  FCFMAT(5X,10A4)
      WRITE(NWRITE,1139)NCC196,NCC197
1139  FCFMAT(5X,10A4)
      WRITE(NWRITE,1140)NCC198,NCC199
1140  FCFMAT(5X,10A4)
      WRITE(NWRITE,1141)NCC200,NCC201
1141  FCFMAT(5X,10A4)
      WRITE(NWRITE,1142)NCC202,NCC203
1142  FCFMAT(5X,10A4)
      WRITE(NWRITE,1143)NCC204,NCC205
1143  FCFMAT(5X,10A4)
      WRITE(NWRITE,1144)NCC206,NCC207
1144  FCFMAT(5X,10A4)
      WRITE(NWRITE,1145)NCC208,NCC209
1145  FCFMAT(5X,10A4)
      WRITE(NWRITE,1146)NCC210,NCC211
1146  FCFMAT(5X,10A4)
      WRITE(NWRITE,1147)NCC212,NCC213
1147  FCFMAT(5X,10A4)
      WRITE(NWRITE,1148)NCC214,NCC215
1148  FCFMAT(5X,10A4)
      WRITE(NWRITE,1149)NCC216,NCC217
1149  FCFMAT(5X,10A4)
      WRITE(NWRITE,1150)NCC218,NCC219
1150  FCFMAT(5X,10A4)
      WRITE(NWRITE,1151)NCC220,NCC221
1151  FCFMAT(5X,10A4)
      WRITE(NWRITE,1152)NCC222,NCC223
1152  FCFMAT(5X,10A4)
      WRITE(NWRITE,1153)NCC224,NCC225
1153  FCFMAT(5X,10A4)
      WRITE(NWRITE,1154)NCC226,NCC227
1154  FCFMAT(5X,10A4)
      WRITE(NWRITE,1155)NCC228,NCC229
1155  FCFMAT(5X,10A4)
      WRITE(NWRITE,1156)NCC230,NCC231
1156  FCFMAT(5X,10A4)
      WRITE(NWRITE,1157)NCC232,NCC233
1157  FCFMAT(5X,10A4)
      WRITE(NWRITE,1158)NCC234,NCC235
1158  FCFMAT(5X,10A4)
      WRITE(NWRITE,1159)NCC236,NCC237
1159  FCFMAT(5X,10A4)
      WRITE(NWRITE,1160)NCC238,NCC239
1160  FCFMAT(5X,10A4)
      WRITE(NWRITE,1161)NCC240,NCC241
1161  FCFMAT(5X,10A4)
      WRITE(NWRITE,1162)NCC242,NCC243
1162  FCFMAT(5X,10A4)
      WRITE(NWRITE,1163)NCC244,NCC245
1163  FCFMAT(5X,10A4)
      WRITE(NWRITE,1164)NCC246,NCC247
1164  FCFMAT(5X,10A4)
      WRITE(NWRITE,1165)NCC248,NCC249
1165  FCFMAT(5X,10A4)
      WRITE(NWRITE,1166)NCC250,NCC251
1166  FCFMAT(5X,10A4)
      WRITE(NWRITE,1167)NCC252,NCC253
1167  FCFMAT(5X,10A4)
      WRITE(NWRITE,1168)NCC254,NCC255
1168  FCFMAT(5X,10A4)
      WRITE(NWRITE,1169)NCC256,NCC257
1169  FCFMAT(5X,10A4)
      WRITE(NWRITE,1170)NCC258,NCC259
1170  FCFMAT(5X,10A4)
      WRITE(NWRITE,1171)NCC260,NCC261
1171  FCFMAT(5X,10A4)
      WRITE(NWRITE,1172)NCC262,NCC263
1172  FCFMAT(5X,10A4)
      WRITE(NWRITE,1173)NCC264,NCC265
1173  FCFMAT(5X,10A4)
      WRITE(NWRITE,1174)NCC266,NCC267
1174  FCFMAT(5X,10A4)
      WRITE(NWRITE,1175)NCC268,NCC269
1175  FCFMAT(5X,10A4)
      WRITE(NWRITE,1176)NCC270,NCC271
1176  FCFMAT(5X,10A4)
      WRITE(NWRITE,1177)NCC272,NCC273
1177  FCFMAT(5X,10A4)
      WRITE(NWRITE,1178)NCC274,NCC275
1178  FCFMAT(5X,10A4)
      WRITE(NWRITE,1179)NCC276,NCC277
1179  FCFMAT(5X,10A4)
      WRITE(NWRITE,1180)NCC278,NCC279
1180  FCFMAT(5X,10A4)
      WRITE(NWRITE,1181)NCC280,NCC281
1181  FCFMAT(5X,10A4)
      WRITE(NWRITE,1182)NCC282,NCC283
1182  FCFMAT(5X,10A4)
      WRITE(NWRITE,1183)NCC284,NCC285
1183  FCFMAT(5X,10A4)
      WRITE(NWRITE,1184)NCC286,NCC287
1184  FCFMAT(5X,10A4)
      WRITE(NWRITE,1185)NCC288,NCC289
1185  FCFMAT(5X,10A4)
      WRITE(NWRITE,1186)NCC290,NCC291
1186  FCFMAT(5X,10A4)
      WRITE(NWRITE,1187)NCC292,NCC293
1187  FCFMAT(5X,10A4)
      WRITE(NWRITE,1188)NCC294,NCC295
1188  FCFMAT(5X,10A4)
      WRITE(NWRITE,1189)NCC296,NCC297
1189  FCFMAT(5X,10A4)
      WRITE(NWRITE,1190)NCC298,NCC299
1190  FCFMAT(5X,10A4)
      WRITE(NWRITE,1191)NCC300,NCC301
1191  FCFMAT(5X,10A4)
      WRITE(NWRITE,1192)NCC302,NCC303
1192  FCFMAT(5X,10A4)
      WRITE(NWRITE,1193)NCC304,NCC305
1193  FCFMAT(5X,10A4)
      WRITE(NWRITE,1194)NCC306,NCC307
1194  FCFMAT(5X,10A4)
      WRITE(NWRITE,1195)NCC308,NCC309
1195  FCFMAT(5X,10A4)
      WRITE(NWRITE,1196)NCC310,NCC311
1196  FCFMAT(5X,10A4)
      WRITE(NWRITE,1197)NCC312,NCC313
1197  FCFMAT(5X,10A4)
      WRITE(NWRITE,1198)NCC314,NCC315
1198  FCFMAT(5X,10A4)
      WRITE(NWRITE,1199)NCC316,NCC317
1199  FCFMAT(5X,10A4)
      WRITE(NWRITE,1200)NCC318,NCC319
1200  FCFMAT(5X,10A4)
      WRITE(NWRITE,1201)NCC320,NCC321
1201  FCFMAT(5X,10A4)
      WRITE(NWRITE,1202)NCC322,NCC323
1202  FCFMAT(5X,10A4)
      WRITE(NWRITE,1203)NCC324,NCC325
1203  FCFMAT(5X,10A4)
      WRITE(NWRITE,1204)NCC326,NCC327
1204  FCFMAT(5X,10A4)
      WRITE(NWRITE,1205)NCC328,NCC329
1205  FCFMAT(5X,10A4)
      WRITE(NWRITE,1206)NCC330,NCC331
1206  FCFMAT(5X,10A4)
      WRITE(NWRITE,1207)NCC332,NCC333
1207  FCFMAT(5X,10A4)
      WRITE(NWRITE,1208)NCC334,NCC335
1208  FCFMAT(5X,10A4)
      WRITE(NWRITE,1209)NCC336,NCC337
1209  FCFMAT(5X,10A4)
      WRITE(NWRITE,1210)NCC338,NCC339
1210  FCFMAT(5X,10A4)
      WRITE(NWRITE,1211)NCC340,NCC341
1211  FCFMAT(5X,10A4)
      WRITE(NWRITE,1212)NCC342,NCC343
1212  FCFMAT(5X,10A4)
      WRITE(NWRITE,1213)NCC344,NCC345
1213  FCFMAT(5X,10A4)
      WRITE(NWRITE,1214)NCC346,NCC347
1214  FCFMAT(5X,10A4)
      WRITE(NWRITE,1215)NCC348,NCC349
1215  FCFMAT(5X,10A4)
      WRITE(NWRITE,1216)NCC350,NCC351
1216  FCFMAT(5X,10A4)
      WRITE(NWRITE,1217)NCC352,NCC353
1217  FCFMAT(5X,10A4)
      WRITE(NWRITE,1218)NCC354,NCC355
1218  FCFMAT(5X,10A4)
      WRITE(NWRITE,1219)NCC356,NCC357
1219  FCFMAT(5X,10A4)
      WRITE(NWRITE,1220)NCC358,NCC359
1220  FCFMAT(5X,10A4)
      WRITE(NWRITE,1221)NCC360,NCC361
1221  FCFMAT(5X,10A4)
      WRITE(NWRITE,1222)NCC362,NCC363
1222  FCFMAT(5X,10A4)
      WRITE(NWRITE,1223)NCC364,NCC365
1223  FCFMAT(5X,10A4)
      WRITE(NWRITE,1224)NCC366,NCC367
1224  FCFMAT(5X,10A4)
      WRITE(NWRITE,1225)NCC368,NCC369
1225  FCFMAT(5X,10A4)
      WRITE(NWRITE,1226)NCC370,NCC371
1226  FCFMAT(5X,10A4)
      WRITE(NWRITE,1227)NCC372,NCC373
1227  FCFMAT(5X,10A4)
      WRITE(NWRITE,1228)NCC374,NCC375
1228  FCFMAT(5X,10A4)
      WRITE(NWRITE,1229)NCC376,NCC377
1229  FCFMAT(5X,10A4)
      WRITE(NWRITE,1230)NCC378,NCC379
1230  FCFMAT(5X,10A4)
      WRITE(NWRITE,1231)NCC380,NCC381
1231  FCFMAT(5X,10A4)
      WRITE(NWRITE,1232)NCC382,NCC383
1232  FCFMAT(5X,10A4)
      WRITE(NWRITE,1233)NCC384,NCC385
1233  FCFMAT(5X,10A4)
      WRITE(NWRITE,1234)NCC386,NCC387
1234  FCFMAT(5X,10A4)
      WRITE(NWRITE,1235)NCC388,NCC389
1235  FCFMAT(5X,10A4)
      WRITE(NWRITE,1236)NCC390,NCC391
1236  FCFMAT(5X,10A4)
      WRITE(NWRITE,1237)NCC392,NCC393
1237  FCFMAT(5X,10A4)
      WRITE(NWRITE,1238)NCC394,NCC395
1238  FCFMAT(5X,10A4)
      WRITE(NWRITE,1239)NCC396,NCC397
1239  FCFMAT(5X,10A4)
      WRITE(NWRITE,1240)NCC398,NCC399
1240  FCFMAT(5X,10A4)
      WRITE(NWRITE,1241)NCC400,NCC401
1241  FCFMAT(5X,10A4)
      WRITE(NWRITE,1242)NCC402,NCC403
1242  FCFMAT(5X,10A4)
      WRITE(NWRITE,1243)NCC404,NCC405
1243  FCFMAT(5X,10A4)
      WRITE(NWRITE,1244)NCC406,NCC407
1244  FCFMAT(5X,10A4)
      WRITE(NWRITE,1245)NCC408,NCC409
1245  FCFMAT(5X,10A4)
      WRITE(NWRITE,1246)NCC410,NCC411
1246  FCFMAT(5X,10A4)
      WRITE(NWRITE,1247)NCC412,NCC413
1247  FCFMAT(5X,10A4)
      WRITE(NWRITE,1248)NCC414,NCC415
1248  FCFMAT(5X,10A4)
      WRITE(NWRITE,1249)NCC416,NCC417
1249  FCFMAT(5X,10A4)
      WRITE(NWRITE,1250)NCC418,NCC419
1250  FCFMAT(5X,10A4)
      WRITE(NWRITE,1251)NCC420,NCC421
1251  FCFMAT(5X,10A4)
      WRITE(NWRITE,1252)NCC422,NCC423
1252  FCFMAT(5X,10A4)
      WRITE(NWRITE,1253)NCC424,NCC425
1253  FCFMAT(5X,10A4)
      WRITE(NWRITE,1254)NCC426,NCC427
1254  FCFMAT(5X,10A4)
      WRITE(NWRITE,1255)NCC428,NCC429
1255  FCFMAT(5X,10A4)
      WRITE(NWRITE,1256)NCC430,NCC431
1256  FCFMAT(5X,10A4)
      WRITE(NWRITE,1257)NCC432,NCC433
1257  FCFMAT(5X,10A4)
      WRITE(NWRITE,1258)NCC434,NCC435
1258  FCFMAT(5X,10A4)
      WRITE(NWRITE,1259)NCC436,NCC437
1259  FCFMAT(5X,10A4)
      WRITE(NWRITE,1260)NCC438,NCC439
1260  FCFMAT(5X,10A4)
      WRITE(NWRITE,1261)NCC440,NCC441
1261  FCFMAT(5X,10A4)
      WRITE(NWRITE,1262)NCC442,NCC443
1262  FCFMAT(5X,10A4)
      WRITE(NWRITE,1263)NCC444,NCC445
1263  FCFMAT(5X,10A4)
      WRITE(NWRITE,1264)NCC446,NCC447
1264  FCFMAT(5X,10A4)
      WRITE(NWRITE,1265)NCC448,NCC449
1265  FCFMAT(5X,10A4)
      WRITE(NWRITE,1266)NCC450,NCC451
1266  FCFMAT(5X,10A4)
      WRITE(NWRITE,1267)NCC452,NCC453
1267  FCFMAT(5X,10A4)
      WRITE(NWRITE,1268)NCC454,NCC455
1268  FCFMAT(5X,10A4)
      WRITE(NWRITE,1269)NCC456,NCC457
1269  FCFMAT(5X,10A4)
      WRITE(NWRITE,1270)NCC458,NCC459
1270  FCFMAT(5X,10A4)
      WRITE(NWRITE,1271)NCC460,NCC461
1271  FCFMAT(5X,10A4)
      WRITE(NWRITE,1272)NCC462,NCC463
1272  FCFMAT(5X,10A4)
      WRITE(NWRITE,1273)NCC464,NCC465
1273  FCFMAT(5X,10A4)
      WRITE(NWRITE,1274)NCC466,NCC467
1274  FCFMAT(5X,10A4)
      WRITE(NWRITE,1275)NCC468,NCC469
1275  FCFMAT(5X,10A4)
      WRITE(NWRITE,1276)NCC470,NCC471
1276  FCFMAT(5X,10A4)
      WRITE(NWRITE,1277)NCC472,NCC473
1277  FCFMAT(5X,10A4)
      WRITE(NWRITE,1278)NCC474,NCC475
1278  FCFMAT(5X,10A4)
      WRITE(NWRITE,1279)NCC476,NCC477
1279  FCFMAT(5X,10A4)
      WRITE(NWRITE,1280)NCC478,NCC479
1280  FCFMAT(5X,10A4)
      WRITE(NWRITE,1281)NCC480,NCC481
1281  FCFMAT(5X,10A4)
      WRITE(NWRITE,1282)NCC482,NCC483
1282  FCFMAT(5X,10A4)
      WRITE(NWRITE,1283)NCC484,NCC485
1283  FCFMAT(5X,10A4)
      WRITE(NWRITE,1284)NCC486,NCC487
1284  FCFMAT(5X,10A4)
      WRITE(NWRITE,1285)NCC488,NCC489
1285  FCFMAT(5X,10A4)
      WRITE(NWRITE,1286)NCC490,NCC491
1286  FCFMAT(5X,10A4)
      WRITE(NWRITE,1287)NCC492,NCC493
1287  FCFMAT(5X,10A4)
      WRITE(NWRITE,1288)NCC494,NCC495
1288  FCFMAT(5X,10A4)
      WRITE(NWRITE,1289)NCC496,NCC497
1289  FCFMAT(5X,10A4)
      WRITE(NWRITE,1290)NCC498,NCC499
1290  FCFMAT(5X,10A4)
      WRITE(NWRITE,1291)NCC500,NCC501
1291  FCFMAT(5X,10A4)
      WRITE(NWRITE,1292)NCC502,NCC503
1292  FCFMAT(5X,10A4)
      WRITE(NWRITE,1293)NCC504,NCC505
1293  FCFMAT(5X,10A4)
      WRITE(NWRITE,1294)NCC506,NCC507
1294  FCFMAT(5X,10A4)
      WRITE(NWRITE,1295)NCC508,NCC509
1295  FCFMAT(5X,10A4)
      WRITE(NWRITE,1296)NCC510,NCC511
1296  FCFMAT(5X,10A4)
      WRITE(NWRITE,1297)NCC512,NCC513
1297  FCFMAT(5X,10A4)
      WRITE(NWRITE,1298)NCC514,NCC515
1298  FCFMAT(5X,10A4)
      WRITE(NWRITE,1299)NCC516,NCC517
1299  FCFMAT(5X,10A4)
      WRITE(NWRITE,1300)NCC518,NCC519
1300  FCFMAT(5X,10A4)
      WRITE(NWRITE,1301)NCC520,NCC521
1301  FCFMAT(5X,10A4)
      WRITE(NWRITE,1302)NCC522,NCC523
1302  FCFMAT(5X,10A4)
      WRITE(NWRITE,1303)NCC524,NCC525
1303  FCFMAT(5X,10A4)
      WRITE(NWRITE,1304)NCC526,NCC527
1304  FCFMAT(5X,10A4)
      WRITE(NWRITE,1305)NCC528,NCC529
1305  FCFMAT(5X,10A4)
      WRITE(NWRITE,1306)NCC530,NCC531
1306  FCFMAT(5X,10A4)
      WRITE(NWRITE,1307)NCC532,NCC533
1307  FCFMAT(5X,10A4)
      WRITE(NWRITE,1308)NCC534,NCC535
1308  FCFMAT(5X,10A4)
      WRITE(NWRITE,1309)NCC536,NCC537
1309  FCFMAT(5X,10A4)
      WRITE(NWRITE,1310)NCC538,NCC539
1310  FCFMAT(5X,10A4)
      WRITE(NWRITE,1311)NCC540,NCC541
1311  FCFMAT(5X,10A4)
      WRITE(NWRITE,1312)NCC542,NCC543
1312  FCFMAT(5X,10A4)
      WRITE(NWRITE,1313)NCC544,NCC545
1313  FCFMAT(5X,10A4)
      WRITE(NWRITE,1314)NCC546,NCC547
1314  FCFMAT(5X,10A4)
      WRITE(NWRITE,1315)NCC548,NCC549
1315  FCFMAT(5X,10A4)
      WRITE(NWRITE,1316)NCC550,NCC551
1316  FCFMAT(5X,10A4)
      WRITE(NWRITE,1317)NCC552,NCC553
1317  FCFMAT(5X,10A4)
      WRITE(NWRITE,1318)NCC554,NCC555
1318  FCFMAT(5X,10A4)
      WRITE(NWRITE,1319)NCC556,NCC557
1319  FCFMAT(5X,10A4)
      WRITE(NWRITE,1320)NCC558,NCC559
1320  FCFMAT(5X,10A4)
      WRITE(NWRITE,1321)NCC560,NCC561
1321  FCFMAT(5X,10A4)
      WRITE(NWRITE,1322)NCC562,NCC563
1322  FCFMAT(5X,10A4)
      WRITE(NWRITE,1323)NCC564,NCC565
1323  FCFMAT(5X,10A4)
      WRITE(NWRITE,1324)NCC566,NCC567
1324  FCFMAT(5X,10A4)
      WRITE(NWRITE,1325)NCC568,NCC569
1325  FCFMAT(5X,10A4)
      WRITE(NWRITE,1326)NCC570,NCC571
1326  FCFMAT(5X,10A4)
      WRITE(NWRITE,1327)NCC572,NCC573
1327  FCFMAT(5X,10A4)
      WRITE(NWRITE,1328)NCC574,NCC575
1328  FCFMAT(5X,10A4)
      WRITE(NWRITE,1329)NCC576,NCC577
1329  FCFMAT(5X,10A4)
      WRITE(NWRITE,1330)NCC578,NCC579
1330  FCFMAT(5X,10A4)
      WRITE(NWRITE,1331)NCC580,NCC581
1331  FCFMAT(5X,10A4)
      WRITE(NWRITE,1332)NCC582,NCC583
1332  FCFMAT(5X,10A4)
      WRITE(NWRITE,1333)NCC584,NCC585
1333  FCFMAT(5X,10A4)
      WRITE(NWRITE,1334)NCC586,NCC587
1334  FCFMAT(5X,10A4)
      WRITE(NWRITE,1335)NCC588,NCC589
1335  FCFMAT(5X,10A4)
      WRITE(NWRITE,1336)NCC590,NCC591
1336  FCFMAT(5X,10A4)
      WRITE(NWRITE,1337)NCC592,NCC593
1337  FCFMAT(5X,10A4)
      WRITE(NWRITE,1338)NCC594,NCC595
1338  FCFMAT(5X,10A4)
      WRITE(NWRITE,1339)NCC596,NCC597
1339  FCFMAT(5X,10A4)
      WRITE(NWRITE,1340)NCC598,NCC599
1340  FCF
```

[illegible]

```

C
C      INITIALIZE ALL MATRICES

```

FILE: TURBO FCRTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

CC 110 I = 1, NN
      F(I) = C.CC
      PSI(I) = 0.0, F0
      RMS(I) = C.CC
      UVEL(I) = 0.0, CC
      VVEL(I) = 0.0, CC
      TVEL(I) = C.CC
      PRESS(I) = C.CC
      RC(I) = 0.0, CC
      ZC(I) = 0.0, CC
      B(I) = C.CC
      AFS(I) = C.CC
      ETA(I) = C.CC
      NBC(I) = 0.0
      WFL(I) = C.CC
      RHO(I) = 0.0, CC
      RHON(I) = 0.0, CC
      ENTRCF(I) = C.CC
      F(I) = C.CC
      HS(I) = 0.0, CC
      HF(I) = 0.0, CC
      ALP(I) = 0.0, CC
      TWEL(I) = C.CC
      BE(I) = 0.0, CC
      DEVI(I) = 0.0, CC
      PRAT(I) = C.CC
110  CC CONTINUE
      CC 150 I = 1, NN
            DO 150 J = 1, NN
                  EV(I,J) = C.CC
150  CC CONTINUE
      CC 160 I = 1, NNE
            DO 160 J = 1, NNE
                  EPS(I,J) = 0.0, CC
160  CC CONTINUE
      CC 165 I = 1, NE
            NTE(I) = C
            DO 165 J = 1, NNE
                  NCE(I,J) = 0
165  CC CONTINUE
      RETURN
      ENC

C
C
      SUBROUTINE OUTPUT(X, KK, UVEL, VVEL, TVEL, TWEL, PSI, RC, ZC
1, RHO, RHO1, RMS, TEMP, TTCT, PRESS, PTCT, PFCT, ALP, BE,
      NACD, NE4, NNE4, NROWS)
C *****
C *****
C *****
C *****
C *****
C *****
      IMPLICIT REAL*8(A-H, P-Z)
      INTEGER*4 AREAD, AWRITE, IC, NNCD, NE4, NNE4, NROWS, LIM1, LIM2
      COMMON /ACCOUNT/ ACUL, NCUL, NN, NE
      1, NNE, NNEC, NNEC
      COMMON /LIC/ AREAD, AWRITE
      COMMON /FCCH/ RG, G, CP, FT, TT, WG, WDET, PFCT, RHOSTA,
1LIM1, UCLCT, PSI, F21, F212, F111, CC
      DIMENSION ZC(1), RC(1), PHO(1), H(1), HS(1), PSI(1)
      DIMENSION UVEL(1), VVEL(1), TVEL(1), WFL(1), TWEL(1)
      DIMENSION TEMP(1), TTCT(1), PRESS(1), PTCT(1), PHOT(1)
      DIMENSION ALP(1), BE(1)
C
      IF(IEF.EC.1) GOTO 450
      WRITE(NAWRITE, 1000) KK, X
1600  FORMAT(' PROGRAM TERMINATED ON ITERATION NO. ', I3, /,
1' RESULTS WHICH FOLLOW ARE FOR CONVERGENCE EPSILON = ', D19.12)
C
      GOTO 1310

```

FILE: TLR80 FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

C      GOTO 1104
45C    WRITE(NWRITE,1200)X
1200   FORMAT('STREAM FUNCTION CONVERGENCE CRITERION SATISFIED ON ITERATION NUMBER',I3,'//', 'RESULTS ARE AS FOLLOWS FOR CONVEGENCE 2EPSILON =',D15,12)
C      WRITE(NWRITE,1310)
C      GOTO 1310
C      CHANGE UNITS OF VELOCITY TO FT/SEC
1104   CC 600 I = 1,NN
        UVEL(I) = UVEL(I)/12.00
        VVEL(I) = VVEL(I)/12.00
        TVEL(I) = TVEL(I)/12.00
        TWEL(I) = TWEL(I)/12.00
        ALP(I) = ALP(I) * 57.29578
        WRL(I) = WRL(I) / 144.000
        BE(I) = BE(I) * 57.29578
        H(I) = H(I) / (GC*144.000*BTU)
        HS(I) = HS(I) / (GC*144.000*BTU)
600    CONTINUE
        WRITE(NWRITE,1109)
1109   FORMAT('1')
        WRITE(NWRITE,1110)
1110   FORMAT('1',7777,27X,'FINITE ELEMENT RESULTS',777,
1       '1',A000,5X,'PSI(I)',10X,'VZ',13X,'VR',12X,'R(I)',10X,'DENSITY')
        C = 1
        CC 1122 I = 1,NN
        WRITE(NWRITE,1120)(I,PSI(I),UVEL(I),VVEL(I),AC(I),R-FC(I))
        IF(I.EQ.NN) GOTO 1124
        IC = I
        CC = FLOCAT(IC)/40.
        IF(C.NE.CC) GOTO 1123
        C = C + 1
        WRITE(NWRITE,1053)
        WRITE(NWRITE,1110)
        GOTO 1122
1124   WRITE(NWRITE,1053)
1123   CONTINUE
1122   CONTINUE
1120   FORMAT('1',13,2X,C13.6,2X,D13.6,2X,D13.6,2X,D13.6,2X,C13.6)
1111   WRITE(NWRITE,1121)
1121   FORMAT('1',A000,5X,'WRL(I)',10X,'HT',13X,'VT',12X,'WT',14X,'HS')
        C = 1
        CC 1322 I = 1,NN
        WRITE(NWRITE,1300)(I,WRL(I),H(I),TVEL(I),TWEL(I),HS(I))
1300   FORMAT('1',13,2X,C13.6,2X,D13.6,2X,D13.6,2X,D13.6,2X,D13.6)
        IF(I.EQ.NN) GOTO 1324
        IC = I
        CC = FLOCAT(IC)/40.
        IF(C.NE.CC) GOTO 1323
        C = C + 1
        WRITE(NWRITE,1053)
        WRITE(NWRITE,1121)
        GOTO 1322
1324   WRITE(NWRITE,1053)
10113  FORMAT('1')
1323   CONTINUE
1322   CONTINUE
1421   WRITE(NWRITE,1421)
        FORMAT('1',A000,5X,'TEMP',11X,'TTOT',11X,'PRESS',11X,
1       '1',FTOT',11X,'PHOT')
        C = 1
        CC 1422 I = 1,NN
        WRITE(NWRITE,1400)(I,TEMP(I),TTOT(I),PRESS(I),PTOT(I),PHOT(I))
1400   FORMAT('1',13,2X,C13.6,2X,D13.6,2X,D13.6,2X,D13.6,2X,D13.6)
        IF(I.EQ.NN) GOTO 1424
        IC = I
        CC = FLOCAT(IC)/40.
        IF(C.NE.CC) GOTO 1423
        C = C + 1
        WRITE(NWRITE,1053)

```

196

FILE: TURBO FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

ENC

C
C
C

```
SUBROUTINE RCTC(RC11,VM1,BETA1,BETA2,RMACHR,RMAX1,RMIN1,  
1CEV,FC12,RH01,FH02,T1,TT1,D1,PT1,ALP1,FH0T1,RH0T2,H2,ENT,OPSIRZ,  
2CPS1Z2,CFAC,ALP2,P2,PT2,T2,TT2,TONEG,VM2,RPASS,NMCD,NE4,NE4,  
3NFCWS)
```

C
C
C
C
C
C

```
+++++  
+++++  
++  
++  
+++++  
+++++
```

```
IMPLICIT REAL*8(A-H,O-Z)  
INTEGER*4 NREAC,NWRITE,IC,NMCD,NE4,NE4,NROWS,LIMR,LIMI  
COMMON /ACCOUNT/ NCOL,NCOL1,AN,NE  
1,ANE,NVRC,NANBC  
COMMON /RCM/ RC,G,CP,PT,TT,WC,WDOT,FHGT,RHOST1,  
1C1ALET,LCCLET,PS11,PTU,F212,F111,SC  
COMMON /ACCOUNT/ NROW,MROW1,KK
```

C
C

FIND HPEL,WRL,AND TWEL AT LCC NODES 3,4,5(RCTOR).

```
FC112 = FC11 * PC11  
RC113 = RC11 * PC112  
RC114 = RC11 * PC113  
RC115 = RC11 * PC114  
RC116 = RC11 * PC115  
RC117 = RC11 * PC116  
RC118 = RC11 * PC117  
RC119 = RC11 * PC118  
BETA1 = RCATAN(WG * RC11/ VM1) * 57.29578  
BET1 = BETA1 / 57.29578  
W1 = VM1 / DCCS(BET1)  
RCCN3 = 2.000 * CP * GC * STU  
TRI = T1 + W1 * B1 / (RCCN3 * 144.000)  
RVACHR = CSET((VM1+VM1*(1.00 + DTAN(BET1)*DTAN(BET1))) /  
1(G+GC*RC+TRI*144.000)  
1 RKAPPA = 26.65E8 + C.0691161 * RC112 - 7.8966110-10 * RC118 +  
1 4.00005E-11 * RC119 - 0.121315 * DSIN(RC11) +  
2 3.92005E-4 * DTAN(RC11)  
1 TCVC = C.124E1 - 3.519260-3 * RC11 - 8.560160-5 * RC112  
1 + 5.391440-12 * RC117 - 7.303910-7 * DCCS(RC11) -  
2 5.25533E-5 * DSIN(RC11)  
1 SIG = 7.704673 * 1.227510-3 * RC112 + 1.46E660-8 * PC115  
1 - 1.3388650-3 * CLOG(RC11) - 7.42404E-4 * DSIN(RC11)  
2 + 2.92366210-5 * DTAN(RC11) - 2.12471 * DLOG(RC11)  
1 PHI = 3.771610-4 * RC114 + 2.2770-11 * RC119 -  
1 C.509350E7 * DCCS(RC11) - 0.347410 * DSIN(RC11) +  
2 C.02777612 * DTAN(RC11) - 56.7008 * DLOG(RC11)  
1 TCVC2 = TCVC * TCVC  
1 TCVC3 = TCVC * TCVC2  
1 BETA12 = BETA1 * BETA1  
1 BETA13 = BETA1 * BETA12  
1 BETA14 = BETA1 * BETA13  
1 BETA15 = BETA1 * BETA14  
1 BETA16 = BETA1 * BETA15  
1 SIG2 = SIG * SIG  
1 SIG3 = SIG * SIG2  
1 RPASS = (RMAX1 - RC11) / (RMAX1 - RMIN1)  
1 RPASS2 = RPASS * RPASS  
1 RINCD = BETA1 - RKAPPA  
1 RKISH = C.7000  
1 RKIT = 1.00 * TCVC - 78.06 * TCVC2 + 199.9 * TCVC3  
1 RITEN = SIG * 10.00 * BETA1 - 2.2370-12 * BETA16)  
1 SLCEN = -C.024 * (2.5 - SIG) - 0.002264 * (1.5 - SIG) * CSCRT(DABS(1.8  
1 - SIG)) * BETA1 - 2.14E-8 * 126.43-1.0 / (SIG * SIG) * BETA13  
1 RI20 = RKISH * RITEN + SLOPCN * PHI  
EXPC1 = 2.7054 - 1.1375 * RPASS + 0.4375 * RPASS2
```

FILE: TURBO FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

EXPC2 = ((1.0828 - C.344*RPASS + 1.563*RPASS2)/R*ACHR1)**EXPC1
RICIFF = -2.8 + 2.55*RPASS + (5.275 + 7.5*RPASS - 2.5 * RPASS2)
1 * RMACHR1**XPC2
RREF = R120 + RICIFF
RKCELS = C.7JCC
RKCELT = 4.667*TCVC + 24.45*TCVC2
DELTE1 = (-0.0C123 + 0.0257*SIG + 0.000144*SIG2)*BETA1 +
1 (1.510-4 - 2.550-4 * SIG - 3.102C-4 * SIG2)*BETA12 +
2 (-2.07C + 7.343 * SIG + 3.621 * SIG2)*1.0D-6*BETA13
SLCPM = 0.25 + 7.06C-4 * BETA1 - 1.2*6D-5 * BETA12 + 3.108D-7 *
1 BETA13
ACCN = 3.35 - C.0124 * BETA1 - C.0000984 * BETA12
BCCN = C.0070 - C.00070H * BETA1 + 1.36D-5 * BETA12
CCCLD1 = CEXP(-ACCN*SIG) + (BCCN/SIG2)*(CSIN(3.141593*SIG /
2 (1.2*57.29578)))**2
E = 0.966 - 0.00305 * BETA1 + 6.195D-5 * BETA12 - 1.4789D-6 *
1 BETA13
CEL20 = RKCELS * RKCELT * DELTE1 + (SLCPM/(SIG**3)) * PHI +
1 RIDIFF + CCCLD1
ACCN2 = -1.75 + 2.5*RPASS + RPASS**6.59
BCCN2 = C.29 - 5.5*RPASS + 31.84 * RPASS2 - 57.2*RPASS*RPASS2
1 + 35.15 * RPASS2 * RPASS2
CCCN2 = 1.43 + 55.6 * (RPASS - C.535)**2
CCCLDIF = ACCN2 + BCCN2 * R*ACHR1**CCCN
CELREF = CEL20 + CCCLDIF
DEV = DELREF + (RINCC - RREF)*CCCLD1
BETA2 = BETA1 - PHI - RINCC + DEV
EPS = 1.0D-6
IT = 1
BETA1 = BETA1 / 57.29578
BETA2 = BETA2 / 57.29578
RREF = (RC11 + RC12)/2.0D0
GM1 = G - 1.00
GM11 = 1.00 / GM1
RRC2 = RRC1*0.75
VM2 = (CSQRT(CPSIF2*CPSIF2+DPSIZ2*OPSIZ2)) * 144.0D0 /
1 (RRC2*P2*RC12/12.0D0)
60C ALF2 = CATAN((WG*RC12 - VM2*CTAN(BETA2))/VM2)
DFAC = 1 - (VM2*CCOS(BETA1))/(VM1*CCOS(BETA2)) + ((VM1*CTAN(BETA1)*
1 RC11 - VM2*CTAN(BETA2)*RC12)*CCOS(BETA1))/(2.0D0*SIG*VM1*RBAK)
DFAC2 = DFAC*DFAC
DFAC3 = DFAC2 * DFAC
DFAC5 = DFAC2 * DFAC3
TCMEG1 = (0.00215058 + 0.0304237*DFAC - 0.338447*DFAC2 +
1 C.55549C*DFAC3 - 0.60274C*DFAC5) * (2.0D0*SIG/CCOS(BETA2))
TCMEG2 = (0.00312009 + 0.0319210*DFAC - C.106477*DFAC2 +
1 C.0023378*DFAC3 - 0.0433925*DFAC5) * (2.0D0*SIG/CCOS(BETA2))
TCIFF = (TCMEG1 - TCMEG2)
IF(RPASS.GE.0.30) GOTO 100
IF(OPASS.LE.0.10) GOTO 110
TCMEG = TCMEG1 + (TCMEG1 - TCMEG2) * ((0.3-RPASS)/0.20)**2
11C GOTO 120
TCMEG = TCMEG1
10C GOTO 120
12C TCMEG = TCMEG2
TE1 = T01 + WG*WG * (RC12*RC12 - RC11*RC11) / (BCCN3 * 144.0D0)
PR1 = PT1 * (TE1/IT1)**(G*GM11)
PR2 = PT2 * (TE1/IT1)**(G*GM11)
PE1 = PE1 - TCMEG*(PR1 - P1)
K2 = VM2 / CCOS(BETA2)
TE1 = TE1 - W2*V2 / (BCCN3 * 144.0D0)
P2 = PE2 * (T2 / TE1)**(G*GM11)
RRC2 = (F2*144.00) / (RG * T2)
VM2N = (CSQRT(CPSIF2*CPSIF2+DPSIZ2*OPSIZ2)) * 144.0D0 /
1 (RRC2*P2*RC12/12.0D0)
CTEST = CAPS(VM2 - VM2N)
IT = IT + 1
IF(CTEST.LT.EPS) GOTO 700
VM2 = VM2N
IF(IT.LT.100) GOTO 60C
WRITE(6,65C)
65C FORMAT(' CONVERGENCE NOT REACHED IN 20 ITERATIONS')

```

FILE: TURBO FCRTRAN A1 NAVAL POSTGRADUATE SCHOOL

```
700 CONTINUE
    RMACH2 = VM2N*VM2N*(1.0CC + (DTAN(ALF2))**2)/(G*RG*GC*TT2*144.0CC)
    CLANT = 1.0CC + (GM1/2.00)*RMACH2
    PT2 = P2 * QUANT**((G*GM1))
    TT2 = T2 * QUANT
    RHCT2 = (PT2 * 144.0CC) / (RG * TT2)
    ENT = -RC*LOG(PT2/PT1) * GC * 144.0CC
    RETURN
    EAC
```

```
      SUBROUTINE STAT(RCI1,VM1,BETA1,BETA2,RMACHR,RMAX1,RMIN1,
1    DEV,FCI2,RH01,RH02,T1,TT1,P1,PT1,ALP1,RHCT1,RH0T2,H?,ENT,DPSIR2,
2    DPSI22,CFAC,ALF2,P2,PT2,T2,TT2,TJVEG,VM2,RPASS,NRCD,NE4,NNE4,
3    NRCWS)
```

```
      IMPLICIT REAL*8(A-H,P-Z)
      INTEGER*4 NRCD,ALP1,TT1,PT1,ALP1,RHCT1,RH0T2,H?,ENT,DPSIR2,
      COMMON /ACCUR/ NCOL,NCOL1,NA,NE
      1,NAE,NRCD,ALP1,TT1,PT1,ALP1,RHCT1,RH0T2,H?,ENT,DPSIR2,
      COMMON /FCN/ FC,G,C,PT,TT,KG,WDCT,RHCT,RH0T2,
      1,INLET,LCLET,PSII,RTU,F2I2,F1I1,GC
      COMMON /ACCUR/ NRCD,NAE,NRCD,ALP1,TT1,PT1,ALP1,RHCT,RH0T2,
      1,INLET,LCLET,PSII,RTU,F2I2,F1I1,GC
```

FIND MEEL,WRL,AND TWEL AT ICC NOES 3,4,5(PCTOR).

```
      RCI12 = FCI1 * RCI1
      RCI13 = FCI1 * RCI12
      RCI14 = FCI1 * RCI13
      RCI15 = FCI1 * RCI14
      RCI16 = FCI1 * RCI15
      RCI17 = FCI1 * RCI16
      RCI18 = FCI1 * RCI17
      RCI19 = FCI1 * RCI18
      VM1 = VM1 / DCCSIPETA1
      RCHNA = 2.0CC * CF * GC * BTU
      RMACHR = GC*PT*((VM1*VM1*(1.0CC + DTAN(BETA1)*DTAN(BETA1))) /
      1 (G*GC*G*TT*144.0CC))
      RKAPPA = 64.6522 - 1.89349 * RCI1 + 1.695140-9 * RCI17
      1 - .174026 * DCCS(RCI1) + .43145 * DSIN(RCI1)
      TVC = 0.0188089 + 2.311570-3 * RCI1 + 1.000960-5 * RCI12 -
      17.495435-5 * DCCS(RCI1) + 4.822130-5 * DSIN(RCI1) + 3.340030-7 *
      2 DTAN(RCI1)
      SIG = 4.77577 - 0.350357 * RCI1 + 9.654920-3 * RCI12 -
      15.226630-13 * RCI19 - 1.311640-3 * DSIN(RCI1) - 1.727690-6 * DTAN(RCI1)
      PH1 = 146.588 + 1.111700-10 * RCI1 - .925595 * DCCS(RCI1) +
      1 1.52365 * DSIN(RCI1) + .023427 * DTAN(RCI1) - 45.7749 * LOG(RCI1)
      TVC2 = TVC * TVC
      TVC3 = TVC * TVC2
      BETA1 = BETA1 * 57.29578
      BETA12 = BETA1 * BETA1
      BETA13 = BETA1 * BETA12
      BETA14 = BETA1 * BETA13
      BETA15 = BETA1 * BETA14
      BETA16 = BETA1 * BETA15
      SIG2 = SIG * SIG
      SIG3 = SIG * SIG2
      RPASS = (RMAX1 - RCI1) / (RMAX1 - RMIN1)
      RPASS2 = RPASS * RPASS
      RINCD = BETA1 - RKAPPA
      RKISH = 0.7000
      PKIT = 1.0 * TVC - 78.06 * TVC2 + 199.5 * TVC3
      RTEN = SIG * (0.09 * BETA1 - 2.3370-12 * BETA16)
      SLOPEN = -0.024 * (2.5 - SIG) - 0.002264 * (1.8 - SIG) * DSQRT(CABS(1.8
```

FILE: TUPBO FORTRAN AI NAVAL POSTGRADUATE SCHOOL

```

1-SIG1)*BETA1-2.145D-8*(26.43-1.0/(SIG*SIG))*BETA13
RT2D = RMISH * RKIT * PTEN + SLOPEN * PHI
EXPC1 = 2.7054 - 1.1375*RPASS + 0.4375*RPASS2
EXPC2 = ((1.0828 - 0.344*RPASS + 1.563*RPASS2)/RMACHR)*EXP01
RICIFF = -2.8 + 2.55*RPASS + (5.275 + 7.5*RPASS - 2.5 * RPASS2)
1 * RMACHF**XPC2
PIREF = FI2D + PICIFF
RKCELS = C.700C
RKCEL = 4.667*TCVC + 24.45*TCVC2
CELTFN = (-0.00143 + 0.0257*SIG + 0.000144*SIG2)*BETA1 +
1 (1.51D-4 - 3.53D-4 * SIG - 3.102D-4 * SIG2)*BETA12 +
2 (-2.07C + 7.043 * SIG + 3.621 * SIG2)*1.0D-4*BETA13
SLCPEM = 0.25 + 7.06D-4 * BETA1 - 1.286D-5 * BETA12 + 3.109D-7 *
1 BETA13
ACCN = 3.35 - C.0124 * BETA1 - 0.000064 * BETA12
BCCN = C.007D - C.000708 * BETA1 + 1.36D-5 * BETA12
CCELDI = EXPC1*(-ACCN/SIG) + (BCCN/SIG2)*(DSIN(3.141593*SIG /
2 (1.2*57.29579)))**2
E = C.566 - C.002C5 * BETA1 + 6.195D-5 * BETA12 - 1.4788D-6 *
1 BETA13
CEL2D = FKCELS * FKDELT * DELTEN + (SLOPEN/(SIG**B)) * PHI +
1 RIDIFF + DUELCI
ACCN2 = -1.75 + 2.5*PPASS + RPASS**6.58
BCCN2 = C.29 - 5.55*PPASS + 21.84 * PPASS2 - 57.2*RPASS*PPASS2
1 + 35.15 * RPASS2 * PPASS2
CCCN = .43 + 5.6 * (RPASS - 0.535)**2
CCELDI = ACCN2 + BCCN2 * RMACHF**CCCN
CEL2DF = CEL2D + CCELDI
CEV = DELREF + (FINCO - RIREF)*CCELDI
REAR = (EC11 + RC12)/2.000
EFS = 1.0C-6
IT = 1
BETA1 = BETA1 / 57.29578
BETA2 = BETA2 / 57.29578
GM1 = G - 1.0C
GM11 = 1.0C / GM1
RHC2 = RHC1
IT2 = IT1
V1 = DSCFT(VV1*VV1*(1.000 + CTAN(BETA1)*DTAN(BETA1)))
VM2 = (DSCFT(DPSIR2*OPSI22+DPSI22*OPSI22)) * 144.000 /
1 (RHC2**2*RC12/12.00)
60C V2 = DSCFT(VV2*VV2*(1.000 + CTAN(BETA2)*DTAN(BETA2)))
CFAC = 1 - V2/V1 + (V1*CTAN(BETA1)*RC11
1 - VM2*CTAN(BETA2)*RC12)/(2.0C*SIG*V1*RBAR)
TCMEG = (C.00312084 + 0.031321C*DFC - 0.109476*DFAC2 +
1 C.22337E*DFAC3 - 0.0438925*DFAC5) * (2.0D0*SIG/RCOS(BETA2))
PT2 = PT1 - TCMEG*(PT1 - PT1)
T2 = TT1 - GM1 * (VM2*VM2*(1.0C + (CTAN(ALP2))**2)) / (2.00 *
1 G * RG * GC = 144.000)
P2 = PT2 * (T2 / TT1)**(G*GM11)
RHC2 = (P2*144.0C) / (RG * T2)
VM2N = (DSCFT(DPSIR2*OPSI22+DPSI22*OPSI22)) * 144.000 /
1 (RHC2**2*RC12/12.00)
CTEST = ABS(VM2 - VM2N)
IT = IT + 1
IF(ITEST.LT.EFS) GOTO 700
VM2 = VM2N
IF(IT.LT.100) GOTO 600
WRITE(6,65C)
65C FORMAT(' CONVERGENCE NOT REACHED IN 20 ITERATIONS')
70C RMACH2 = VM2N*VM2N*(1.000+(CTAN(BETA2))**2)/(G*RG*GC*T2*144.00C)
CL/NT = 1.0C0 + (GM1/2.00)*RMACH2
RHC2 = (PT2 * 144.0C) / (RG * T2)
EAT = -RC*CLCC(PT2/PT1) * GC * 144.0C0
RETURN
END

```

C *****
SUBROUTINE RCAT(ZA,EA,W,FELX,KK,LINI,LINR,LIN4)
C *****

FILE: TURBO FCRTRM A1 NAVAL POSTGRADUATE SCHOOL

```

C *****
C **
C **
C *****
C
  IMPLICIT REAL*8(A-H,P-Z)
  INTEGER*4 NR,FAC,NWRITE,IC,NMCD,NF4,NNE4,NROWS,LINR,LIMI
  COMMON /FCCM/ BG,G,CP,PT,TT,KG,WDOCT,PHCT,RHCTSA,
  1UINLET,UCLLET,PSI1,RTU,F212,F111,GC
  COMMON /LIC/ NFAC,NWRITE
  DIMENSION ZA(9),EA(9),W(9)

C
  NREAD = 5
  NWRITE = 6
  RELX = C.24C0C0

C
  INITIALIZE STREAM FUNCTION ITERATION COUNTER ,KK. SET SIZE
  OF REAL*8 AND INTEGER*2 STORAGE

C
  KK = 0
  LINR = 56C0C0
  LIMI = 2C0C0
  LIM4 = 5C0C0

C
  THREE-POINT GAUSSIAN ABSCISSAS
  Z21 = C.774596669241483
  Z22 = C.0C0C0C0C0C0C0C0C0
  ZA(1) = Z21
  ZA(2) = Z21
  ZA(3) = Z21
  ZA(4) = Z22
  ZA(5) = Z22
  ZA(6) = Z22
  ZA(7) = -Z21
  ZA(8) = -Z21
  ZA(9) = -Z21
  EA(1) = Z21
  EA(2) = Z22
  EA(3) = -Z21
  EA(4) = Z21
  EA(5) = Z22
  EA(6) = -Z21
  EA(7) = Z21
  EA(8) = Z22
  EA(9) = -Z21

C
  WEIGHTING VALUES FOR 3 PT. GAUSSIAN QUADRATURE
  W1 = 0.5555555555555556
  W2 = 0.5555555555555556
  W(1) = W1*W1
  W(2) = W2*W1
  W(3) = W1*W1
  W(4) = W2*W1
  W(5) = W2*W2
  W(6) = W2*W1
  W(7) = W1*W1
  W(8) = W2*W1
  W(9) = W1*W1

C
  CONSTANTS USED FOR UNITS CONVERSIONS
  BTL = 77E.20C
  F111 = 12.000C
  F212 = 144.00C
  F313 = F111 * F212
  GC = 32.1740C
  RETURN
  END

```

۱۱۱

三

۳۳۳

4 PLO 004
MPL 004

```

++      +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+
++      |         |         |         |         |         |         |         |         |
++      |         |         |         |         |         |         |         |         |
++      +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+
++
++      THIS SUBROUTINE CREATES A TEKTRONIX 618 PLOT OF THE FOLLOWING PARAMETERS:
++
++          AXIAL VELOCITY AT THE ROTOR INLET
++          RELATIVE FLOW ANGLES AT THE ROTOR INLET
++          AXIAL VELOCITY AT THE STATOR OUTLET
++          RELATIVE FLOW ANGLES AT THE STATOR OUTLET
++          ABSOLUTE FLOW ANGLES AT THE ROTOR INLET
++          ABSOLUTE FLOW ANGLES AT THE ROTOR INLET
++          AXIAL VELOCITY AT THE STATOR INLET
++          ABSOLUTE FLOW ANGLES AT THE STATOR INLET
++          AXIAL VELOCITY AT THE STATOR OUTLET
++          ABSOLUTE FLOW ANGLES AT THE STATOR OUTLET
++          ADIABATIC EFFICIENCY AT THE ROTOR INLET
++          RELATIVE DEVIATION ANGLE AT THE ROTOR OUTLET
++          ABSOLUTE DEVIATION ANGLE AT THE STATOR OUTLET
++          STATIC PRESSURE RATIO FOR THE ROTOR
++          STATIC PRESSURE RATIO FOR THE STATOR
++
++      +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+ +-----+

```

202

```

C *****
C **
C **          PLOT THE ROTOR INLET VELOCITY DISTRIBUTION
C **
C *****
DATA TITL1/'AXIAL VELOCITY PROFILE AT ROTOR INLET'/
DATA TITL2/'AXIAL VELOCITY PROFILE AT ROTOR OUTLET'/
DATA TITL3/'AXIAL VELOCITY PROFILE AT STATOR INLET'/
DATA TITL4/'AXIAL VELOCITY PROFILE AT STATOR OUTLET'/
DATA TITL5/'RELATIVE FLOW ANGLES AT ROTOR INLET'/
DATA TITL6/'RELATIVE FLOW ANGLES AT ROTOR OUTLET'/
DATA TITL7/'ABSOLUTE FLOW ANGLES AT ROTOR OUTLET'/
DATA TITL8/'ABSOLUTE FLOW ANGLES AT STATOR INLET'/
DATA TITL9/'ABSOLUTE FLOW ANGLES AT STATOR OUTLET'/
DATA TITL10/'ADAPTATION EFFICIENCY AT ROTOR INLET'/
DATA TITL11/'DEVIATION ANGLE ROTOR OUTLET'/
DATA TITL12/'DEVIATION ANGLE STATOR OUTLET'/
DATA TITL13/'TOTAL PRESSURE RATIO, ROTOR'/
DATA TITL14/'TOTAL PRESSURE RATIO, STATOR'/
WRITE(6,100)
100  FORMAT(5X,' >DO YOU WISH TO ENTER THE PLOT SEQUENCE? ',/,)
      1 1 = YES; 2 = NO. ',/)
      READ(15,4) NANS
      IF(NANS.EC.2) GOTO 500
      J = NROTC2
      DO 30 I = 1, NRCW1
        ORC(I) = RC(J)
        OVCL(I) = VZ(J)
        CBE(I) = BE(J)
        CALP(I) = ET1(J)
        CRAT(I) = PRAT(J)
        J = J + 1
30    CONTINUE
      NR14 = NRCW1
      CXYL(1) = -50.000
      CXYL(2) = RMIN
      CXYL(3) = 700.000
      CXYL(4) = RMAX
      CALL DSINIT
      CALL GSEFSE
      CALL PLOT('MMGNAN3NWL',NR14,CVEL,ORC,CXYL,37,TITL1)
      CALL PLOT('MMGNAN3OWL',NR14,CVEL,ORC,CXYL,37,TITL1)
      CALL DSTERM
      WRITE(6,110)
110   FORMAT(5X,' >DO YOU WISH TO CONTINUE PLOT SEQUENCE? ',/,)
      1 1 = YES; 2 = NO. ',/)
      READ(15,4) NANS
      IF(NANS.EC.2) GOTO 500
      CALL DSINIT
      CALL GSEFSE
      CALL PLOT('MMGNAN3NWL',NR14,CBE,ORC,CXYL,35,TITL5)
      CALL PLOT('MMGNAN3OWL',NR14,CBE,ORC,CXYL,35,TITL5)
      CALL DSTERM
      WRITE(6,110)
      READ(15,4) NANS
      IF(NANS.EC.2) GOTO 500
      CALL DSINIT
      CALL GSEFSE
      CALL PLOT('MMGNAN3NWL',NR14,CRAT,ORC,CXYL,27,TITL13)
      CALL PLOT('MMGNAN3OWL',NR14,CRAT,ORC,CXYL,27,TITL13)
      CALL DSTERM
      WRITE(6,110)
      READ(15,4) NANS
      IF(NANS.EC.2) GOTO 500
      CALL DSINIT
      CALL GSEFSE
      CALL PLOT('MMGNAN3NWL',NR14,CALP,ORC,CXYL,35,TITL10)
      CALL PLOT('MMGNAN3OWL',NR14,CALP,ORC,CXYL,35,TITL10)
      CALL DSTERM
      WRITE(6,110)
      READ(15,4) NANS
      IF(NANS.EC.2) GOTO 500

```

```

J = J + MRCW + 1
DC 40 I = 1, MRCW1
  OPC(I) = RC(J)
  QVEL(I) = VZ(J)
  CBE(I) = BE(J)
  CALP(I) = CEV1(J)
  J = J + 1
40 CONTINUE
  CALL DSINIT
  CALL GSEFSE
  CALL PLCT('MMGANN3NWL', NR14, CVEL, CRC, CXYL, 38, TITL2)
  CALL PLCT('MMGANN3OWL', NR14, CVEL, CRC, CXYL, 39, TITL2)
  CALL DSTERM
  WRITE(6, 110)
  READ(15, 4) NANS
  IF(NANS.EQ.2) GOTC 500
  CALL DSINIT
  CALL GSEFSE
  CALL PLCT('MMGANN3NWL', NR14, CHE, CRC, CXYL, 36, TITL5)
  CALL PLCT('MMGANN3OWL', NR14, CBE, CRC, CXYL, 36, TITL5)
  CALL DSTERM
  WRITE(6, 110)
  READ(15, 4) NANS
  IF(NANS.EQ.2) GOTC 500
  CALL DSINIT
  CALL GSEFSE
  CALL PLCT('MMGANN3NWL', NR14, CALP, CRC, CXYL, 28, TITL11)
  CALL PLCT('MMGANN3OWL', NR14, CALP, CRC, CXYL, 28, TITL11)
  CALL DSTERM
  WRITE(6, 110)
  READ(15, 4) NANS
  IF(NANS.EQ.2) GOTC 500
  J = J - MRCW1
  DC 41 I = 1, MRCW1
    OPC(I) = OC(J)
    CBE(I) = ALF(J)
    J = J + 1
41 CONTINUE
  CALL DSINIT
  CALL GSEFSE
  CALL PLCT('MMGANN3NWL', NR14, CHE, CRC, CXYL, 36, TITL7)
  CALL PLCT('MMGANN3OWL', NR14, CBE, CRC, CXYL, 36, TITL7)
  CALL DSTERM
  WRITE(6, 110)
  READ(15, 4) NANS
  IF(NANS.EQ.2) GOTC 500
  J = J + MRCW + 1
  DC 50 I = 1, MRCW1
    OPC(I) = OC(J)
    CVEL(I) = VZ(J)
    CPAT(I) = PRAT(J)
    CBE(I) = ALF(J)
    J = J + 1
50 CONTINUE
  CALL DSINIT
  CALL GSEFSE
  CALL PLCT('MMGANN3NWL', NR14, CVEL, CRC, CXYL, 39, TITL3)
  CALL PLCT('MMGANN3OWL', NR14, CVEL, CRC, CXYL, 38, TITL3)
  CALL DSTERM
  WRITE(6, 110)
  READ(15, 4) NANS
  IF(NANS.EQ.2) GOTC 500
  CALL DSINIT
  CALL GSEFSE
  CALL PLCT('MMGANN3NWL', NR14, CBE, CRC, CXYL, 36, TITL8)
  CALL PLCT('MMGANN3OWL', NR14, CBE, CRC, CXYL, 36, TITL8)
  CALL DSTERM
  WRITE(6, 110)
  READ(15, 4) NANS
  IF(NANS.EQ.2) GOTC 500
  CALL DSINIT
  CALL GSEFSE

```

```

CALL PLCT('MNGNAN3NWL',NR14,CBAT,CRC,CXYL,29,TITL14)
CALL PLCT('MNGNAN3OWL',NR14,CBAT,CRC,CXYL,29,TITL14)
CALL DSTERM
WRITE(6,11C)
READ(15,4) NANS
IF(NANS.EC.2) GOTO 500
J = J + MRCW + 1
CC 60 I = 1,MRCW1
   ORC(I) = PC(J)
   OVEL(I) = VZ(J)
   ORE(I) = ALP(J)
   CALP(I) = CEVI(J)
   J = J + 1
60 CONTINUE
CALL DSINIT
CALL GSEFSE
CALL PLCT('MNGNAN3NWL',NR14,CVEL,CRC,CXYL,39,TITL4)
CALL PLCT('MNGNAN3OWL',NR14,CVEL,CRC,CXYL,39,TITL4)
CALL DSTERM
WRITE(6,11C)
READ(15,4) NANS
IF(NANS.EC.2) GOTO 500
CALL DSINIT
CALL GSEFSE
CALL PLCT('MNGNAN3NWL',NR14,CBE,OPC,CXYL,37,TITL3)
CALL PLCT('MNGNAN3OWL',NR14,CBE,OPC,CXYL,37,TITL3)
CALL DSTERM
WRITE(6,11C)
READ(15,4) NANS
IF(NANS.EC.2) GOTO 500
CALL DSINIT
CALL GSEFSE
CALL PLCT('MNGNAN3NWL',NR14,CALP,CRC,CXYL,29,TITL12)
CALL PLCT('MNGNAN3OWL',NR14,CALP,CRC,CXYL,29,TITL12)
CALL DSTERM
50C CONTINUE
RETURN
END

```

```

SUBROUTINE SLINE(RC,PS1,VZ,VQ,ACDE,VM1,RCI1,RCI2,ALP1,NTS1,PHQ1,
1  IFRC,ALO,EL,II,ISTAL,M,PHCTT,TMP,TTOT,PRESS,PTOT,Z,FL,T1,TT1,PT1,
2  RFOCI1,H1,PS1,Z2,PS1Z2,UPSIZ2,FMS,Z1,VZ1,VR1,IC,ENT4OP,ENT1,
3  BRAC,NE4,NR4,ARC5)

```

```

IMPLICIT REAL*8(A-H, P-Z)
INTEGER*4 NREAD, NWRITE, IC, NNC0, NE4, NNE4, NROWS, LIMR, LIMJ
COMMON /ACCOUNT/ NCUL, NCUL1, AN, NE
1, ANE, NNPC, NNBC
COMMON /FCC4/ FG, G, CD, PT, TT, NG, WDCI, F-CT, RHCST4,
1UINLET, UCUL, PSI(1, NTJ), FZ12, F111, JC
COMMON /ACCOUNT/ MROW, MROW1, KK
DIMENSION PC(1), VR(1), E1(3), SF(3), PHC(1), PHOT(1), TEMP(1), TTOT(1)
DIMENSION VZ(1), VL(1), PL(1), TWEL(1), PRESS(1), PTOT(1), R(1)
DIMENSION PSI(1), ALP(1), SE(1), F(1), HS(1), Z1(3), R(8), S(8), ZC(1)
DIMENSION RC(8), ZC(8), RJAC(2, 2), JZ(3), CR(8), OPSIP(3), OPSIZ(3)
DIMENSION ENTCP(1)
DIMENSION INLET(1), NTE(1), NCCF(NE4, 1)

```

205

```

N11 = NODE(11,N)
P = PSI(N11)
IT = 1
K = 1
IF(ISTAT.EQ.1.AND.NTE1.EQ.1) GOTO 700
IF(ISTAT.EQ.1) GOTO 200
RC12 = RC(N11)
Q2 = E(N11)
CHECK TO SEE IF P IS WITHIN THE PRESENT ELEMENT
IF(P.GE.PSI(NODE(K,5)))GOTO140
CHECK NEXT ELEMENT BELOW THE PRESENT ELEMENT.
K = K + 1
GOTO130
140 IF(P.GT.PSI(NODE(K,4)))GOTO170
EL = E1(4)
ER = E1(5)
150 E2 = (EL + ER)/2.00
CALL SFAP(E2,-1.00,SF)
PA = SF(3)*PSI(NODE(K,2)) + SF(4)*PSI(NODE(K,4))
+ SF(5)*PSI(NODE(K,5))
CHECK FOR STREAMLINE CONVERGENCE
EPS = ABS(PA - P)
IF(EPS.LT.1.0-06)GOTO190
IT = IT + 1
IF(IT.GT.15)GOTO190
IF(PA.LT.P)GOTO160
EL = E2
GOTO150
160 ER = E2
GOTO150
170 IF(P.GT.PSI(NODE(K,3)))GOTO135
EL = E1(3)
ER = E1(4)
GOTO150
CHECK NEXT ELEMENT ABOVE PRESENT ELEMENT
K = K - 1
GOTO140
IF CONVERGENCE CRITERIA SATISFIED.
CALCULATE WHIRL AND STATIC ENTHALPY
NK3 = NODE(K,3)
NK4 = NODE(K,4)
NK5 = NODE(K,5)
RC11 = SF(3) * RC(NK3) + SF(4) * RC(NK4) +
SF(5) * RC(NK5)
VZ1 = SF(3) * VZ(NK3) + SF(4) * VZ(NK4) +
SF(5) * VZ(NK5)
VR1 = SF(3) * VR(NK3) + SF(4) * VR(NK4) +
SF(5) * VR(NK5)
VM1 = DSQRT(VZ1 * VZ1 + VR1 * VR1)
ALP1 = SF(3) * ALP(NK3) + SF(4) * ALP(NK4) +
SF(5) * ALP(NK5)
RHO1 = SF(3) * RHO(NK3) + SF(4) * RHO(NK4) +
SF(5) * RHO(NK5)
RHOT1 = SF(3) * RHOT(NK3) + SF(4) * RHOT(NK4) +
SF(5) * RHOT(NK5)
T1 = SF(3) * TEMP(NK3) + SF(4) * TEMP(NK4) +
SF(5) * TEMP(NK5)
TT1 = SF(3) * TTOT(NK3) + SF(4) * TTOT(NK4) +
SF(5) * TTOT(NK5)
PT1 = SF(3) * PTOT(NK3) + SF(4) * PTOT(NK4) +
SF(5) * PTOT(NK5)
P1 = SF(3) * PRESS(NK3) + SF(4) * PRESS(NK4) +
SF(5) * PRESS(NK5)
ENT1 = SF(3) * ENTROP(NK3) + SF(4) * ENTROP(NK4) +
SF(5) * ENTROP(NK5)
H1 = SF(3) * H(NK3) + SF(4) * H(NK4) +
SF(5) * H(NK5)
HS1 = SF(3) * HS(NK3) + SF(4) * HS(NK4) +
SF(5) * HS(NK5)

```

```

1      SF(5) = PS(NK5)
      K = I
      GOTD 210
20C    RC11 = FC(N11)
      VZ1 = VZ(N11)
      VR1 = VR(N11)
      VM1 = CSOFT(VZ1 * VZ1 + VR1 * VR1)
      ALP1 = ALP(N11)
      RHO1 = RHO(N11)
      RHO11 = RHOCTT(N11)
      P1 = PRESS(N11)
      PT1 = PTOT(N11)
      TT1 = TTOT(N11)
      T1 = TEMP(N11)
      H1 = F(N11)
      HS1 = PS(N11)
      GOTC 215
21C    IF(VT1.EQ.1.OR.1STAL.EQ.3) GOTO 300
215    IT = C
      K = I
      C      CHECK TO SEE IF P IS WITHIN THE PRESENT ELEMENT
230    IF(P.GE.PSI(NCDE(K,7)))GOTO240
      C      CHECK NEXT ELEMENT BELOW THE PRESENT ELEMENT.
      K = K + 1
      GOTD 230
24C    IF(P.GT.PSI(NCDE(K,8)))GOTO270
      EL = E1(S)
      ER = E1(7)
25C    ER2 = (ER + SF)/2.CO
      CALL SUBROUTINE(E2,1,DC,SF)
      PA = SF(1)*PSI(NCDE(K,1)) + SF(7)*PSI(NCDE(K,7))
      C      1
      C      + SF(8)*PSI(NCDE(K,8))
      C      CHECK FOR STREAMLINE CONVERGENCE
      EPS = CABS(PA - P)
      IF(EPS.LT.1.0-C6)GOTO250
      IT = IT + 1
      IF(IT.GT.15)GOTO250
      IF(PA.LT.P)GOTO260
      GOTD 250
26C    GOTD 250
      ER = ER2
      GOTD 250
27C    IF(P.GT.PSI(NCDE(K,1)))GOTO285
265    EL = E1(1)
      ER = E1(8)
      GOTD 250
      C      CHECK NEXT ELEMENT ABOVE PRESENT ELEMENT
285    K = K - 1
      GOTD 240
      C
      C      IF CONVERGENCE CRITERIA SATISFIED.
      C      CALCULATE WHIPL AND STATIC ENTHALPY
29C    RC12 = SF(1)*FC(NCDE(K,1)) + SF(7)*FC(NCDE(K,7))
      C      1
      C      + SF(8)*FC(NCDE(K,8))
      C      1
      B2 = SF(1)*E(NCDE(K,1)) + SF(7)*E(NCDE(K,7))
      C      + SF(8)*E(NCDE(K,8))
      C      GOTD TO NEXT ELEMENT.
30C    N1 = NCDE(K,1)
      N2 = NCDE(K,2)
      N3 = NCDE(K,3)
      N4 = NCDE(K,4)
      N5 = NCDE(K,5)
      N6 = NCDE(K,6)
      N7 = NCDE(K,7)
      N8 = NCDE(K,8)
      RC$(1) = FC(N1)
      RC$(2) = FC(N2)
      RC$(3) = FC(N3)
      RC$(4) = FC(N4)
      RC$(5) = FC(N5)
      RC$(6) = FC(N6)

```

SLEROUTINE FCALIF,W,F,2A,EA,VZ,PC,ZC,WRL,VU,NRC,NODE,WU,
INTF,HQ,ENTROP,ITOT,NVCD,NB4,NB6,PROBS)

```

+++++
++
++      THIS SUBROUTINE CALCULATES THE RIGHT-HAND SIDE VECTOR
++      F(R,Z) FROM KNOWN RADIAL DISTRIBUTIONS OF WHIRL,
++      ENTHALPY, PC*ENTHALPY, AND ENTROPY.
++      CALL STATEMENT DEFINITIONS:
++      F = RIGHT HAND SIDE VECTOR
++      N = GAUSSIAN HEIGHT FUNCTION ARRAY
++      H = NOCAL TOTAL ENTHALPY VECTOR
++      ZA = ARRAY OF EXCISE GAUSSIAN VALUES
++      EA = ARRAY OF ETA GAUSSIAN VALUES
++

```

FILE: TLFB0 FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```

C  **      VZ = NODAL AXIAL VELOCITY
C  **      VC = ARRAY OF NODAL X COORDINATES
C  **      ZC = ARRAY OF NODAL Z COORDINATES
C  **      WPL = NODAL ANGULAR MOMENTUM VECTOR
C  **      VU = NODAL ABSOLUTE TANGENTIAL VELOCITY VECTOR
C  **      ABC = NODAL WHERE BOUNDARY CONDITIONS APPLY
C  **      NCCS = ARRAY OF ELEMENTAL NODE ASSIGNMENTS
C  **      NN = NUMBER OF NODES IN THE MESH
C  **      NE = NUMBER OF ELEMENTS IN THE MESH
C  **      VU = NODAL RELATIVE TANGENTIAL VELOCITY VECTOR
C  **      NTE = TYPE OF ELEMENT ARRAY
C  **
C  ****

```

```

C  IMPLICIT REAL*8(A-H,P-Z)
C  INTEGER*4 NREAC, NWRITE, IC, NNC0, NE4, NNE4, NROWS, LIMR, LIMT
C  COMMON /ACOUNT/ NCOL, NCOL1, NA, NE
C  1, NNE, NNEC, NNNHC
C  COMMON /ACOUNT/ MROW, MROW1, KK
C  COMMON /ECC4/ RG, G, CP, PT, TT, WG, WDET, FDET, RHOSTA,
C  1, IALST, LCOLST, PSIT, STJ, F212, F111, F2
C  DIMENSION VC(1), VC(1), F(1), F(1), TTOT(1)
C  DIMENSION VZ(1), VZ(1), WPL(1), WPL(1), SF(8)
C  DIMENSION ABC(1), NTE(1), NNE(NE4,1)
C  DIMENSION W(4), ZA(4), ZA(4), F(8), F(1), ENTROP(1)
C  DIMENSION D(3), E(3), SF(3), RCS(3), ZCS(3), RJAC(2,2)

```

 ZC(1) = 0.0

```

C  DO 50 I = 1, NNE
C     F(I) = 0.0
C  CONTINUE

```

 CYCLE FOR EACH ELEMENT.

```

C  DO 100 II = 1, NE
C     N1 = NCCS(1,1)
C     N2 = NCCS(1,2)
C     N3 = NCCS(1,3)
C     N4 = NCCS(1,4)
C     N5 = NCCS(1,5)
C     N6 = NCCS(1,6)
C     N7 = NCCS(1,7)
C     N8 = NCCS(1,8)
C     RCS(1) = VC(N1)
C     RCS(2) = VC(N2)
C     RCS(3) = VC(N3)
C     RCS(4) = VC(N4)
C     RCS(5) = VC(N5)
C     RCS(6) = VC(N6)
C     RCS(7) = VC(N7)
C     RCS(8) = VC(N8)
C     ZCS(1) = ZC(N1)
C     ZCS(2) = ZC(N2)
C     ZCS(3) = ZC(N3)
C     ZCS(4) = ZC(N4)
C     ZCS(5) = ZC(N5)
C     ZCS(6) = ZC(N6)
C     ZCS(7) = ZC(N7)
C     ZCS(8) = ZC(N8)

```

 CYCLE FOR EACH LOCAL NODE.

```

C  DO 300 J = 1, 8
C     CALL SHAPE(SA(J), ZA(J), SF)
C     CALL JACOB(SA(J), ZA(J), D, F, RCS, ZCS, RJAC)
C     DETJ = RJAC(1,1)*RJAC(2,2) - RJAC(1,2)*RJAC(2,1)
C     FJAC = 1.0 / DETJ
C     DUM1 = RJAC(1,1) / DETJ
C     RJAC(1,1) = RJAC(2,2) / DETJ

```

```

      RJAC(1,2) = -RJAC(1,2) / DETJ
      RJAC(2,1) = -RJAC(2,1) / DETJ
      RJAC(2,2) = DUM1
      FIND NI*LI,NI*VI,T*ETAI,NI*RI,D(WFL)/CR,D(H)/DR
      SUMU = 0.00
      SUMV = C.00
      SUMR = C.00
      CSDR = C.00
      SUMT = C.00
      TDSDR = C.CJ
      DWRLR = 0.00
      DHDR = C.00
      CC 11C KL = 1,ANE
      N41 = NCDE(11,KL)
      SUMU = SUMU + SF(KL)*VZ(N41)
      SUMT = SUMT + SF(KL)*TTOT(N41)
      IF (NTE(11).EQ.2) GOTC105
      SUMV = SUMV + SF(KL)*VU(N41)
      DHCR = DHDR + (RJAC(2,1)*E(KL)
      + RJAC(2,2)*E(KL))*H(N41)
      CSCR = CSDR + (RJAC(2,1)*D(KL)
      + RJAC(2,2)*E(KL))*ENTROP(N41)
      GOTC106
      SUMV = SUMV + SF(KL)*VU(N41)
      DHCR = DHCR + (RJAC(2,1)*E(KL)
      + RJAC(2,2)*E(KL))*H(N41)
      CSCR = CSDR + (RJAC(2,1)*D(KL)
      + RJAC(2,2)*E(KL))*ENTROP(N41)
      SUMR = SUMR + SF(KL)*RC(N41)
      DWFLF = DWRLR + (RJAC(2,1)*D(KL)
      + RJAC(2,2)*E(KL))*RL(N41)
      105
      1
      1
      106
      1
      11C
      1
      CONTINUE
      FIND F$ (NCDE(11,1))
      TDSCF = SUMT + DSDR
      CC 20C I = 1,ANE
      XX = (TDSDR - DHCR + (SUMV/SUMR)*DWRLR)*(SF(1)/SUMU)*DETJ
      F$ (I) = F$ (I) + XX*(J)
      20C
      CC
      CONTINUE
      CC
      CONTINUE
      ASSEMBLE RIGHT HAND SIDE VECTOR.
      F(N1) = F(N1) + F$ (1)
      F(N2) = F(N2) + F$ (2)
      F(N3) = F(N3) + F$ (3)
      F(N4) = F(N4) + F$ (4)
      F(N5) = F(N5) + F$ (5)
      F(N6) = F(N6) + F$ (6)
      F(N7) = F(N7) + F$ (7)
      F(N8) = F(N8) + F$ (8)
      DO 50C I = 1,ANE
      F(I) = C.CC
      50C
      CONTINUE
      RETURN
      50C
      10C
      50C
      10C

```

210

SLRCUT INE DSINO NOT INCLUDED NON-IMSL LIBRARY SUBROUTINE

```

SUBROUTINE STIFF(RC,ZC,R,EM,S,Z,E4,W,NODE,RHO,
1PSI,F,PMS,FM,APC,ANCD,NE4,NNE4,NROWS)
+++++
+++++
+++++
IMPLICIT REAL*8(A-H,P-Z)
INTEGER*4 NREAC,NWPITE,IC,MNCD,NE4,NNE4,NROWS,LIMR,LIMI
COMMON /ACCOUNT/ ACOL,NCOL,AN,FE
1,ANE,NMRC,MNNEC
COMMON /FCCN/ RC,G,C,CF,PT,TT,BG,WDCT,PHCT,PHCSTA,
1UINLET,UOLLET,FSI1,BTU,F212,FY11,GC
COMMON /ACCOUNT/ MRCW,MROWL,KK
COMMON /LIC/ NREAC,NWPITE
DIMENSION ZC(1),PC(1),B(1),RPF(9),RFC(1)
DIMENSION NC(1),NDUR(ME4,1),A(6)
DIMENSION EM(MNCE,1),RIJAC(2,2),PSI(1),F(1),PMS(1)
DIMENSION D(1),E(2),SE(3),Z2(3),EA(9),DNDZ(3)
DIMENSION RC(1,2),ZC3(3),EMI(6),RIJAC(2,2),ONDZ(3)

```

CC 400 I I
N1 = =
N2 = =
N3 = =
N4 = =
N5 = =
N6 = =
N7 = =
N8 = =
RC \$(12)
RC \$(13)
RC \$(14)

```

RC(5) = RC(NS)
RC(6) = RC(NS)
RC(7) = RC(NS)
RC(8) = RC(NS)
ZC(1) = ZC(NS)
ZC(2) = ZC(NS)
ZC(3) = ZC(NS)
ZC(4) = ZC(NS)
ZC(5) = ZC(NS)
ZC(6) = ZC(NS)
ZC(7) = ZC(NS)
ZC(8) = ZC(NS)

```

PERFORM GAUSSIAN QUADRATURE INTEGRATION

```

DO 320 I = 1,6
CALL SHAPE(EA(I),ZA(I),SF)
CALL JACCP(EA(I),ZA(I),D,E,RC, ZC,RJAC)
DETJ = RJAC(1,1)*RJAC(2,2) - RJAC(1,2)*RJAC(2,1)
C
FIND INVERSE OF JACOBIAN
DUM1 = RJAC(1,1) / DETJ
RJAC(1,1) = RJAC(2,2) / DETJ
RJAC(1,2) = -RJAC(1,2) / DETJ
RJAC(2,1) = -RJAC(2,1) / DETJ
RJAC(2,2) = DUM1
DC 321 J = 1,8
CNDZ(J) = RJAC(1,1)*D(J) + RJAC(1,2)*E(J)
CNDR(J) = RJAC(2,1)*D(J) + RJAC(2,2)*E(J)
321 CCNTINUE
IF(II,AE,1) GOTO 130
100 CONTINUE

```

FIND RHC, R, AND B FOR NUMERICAL INTEGRATION.

```

R+CRB = C,CCO
DC 330 L = 1,NNE
N1IL = NCDE(II,L)
RHC(R) = RHC(R) + SF(L)*PHO(N1IL)*RC(N1IL)*B(N1IL)
330 CCNTINUE
SFK = (1.00/(R+CRB))*144.00*12.000

```

```

DO 300 J = 1,NNE
IF(II,AE,1) GOTO 309
300 CCNTINUE
DC 310 K = 1,NNE
EMS(J,K) = EMS(J,K) + W(I) * SFK * (CNDZ(J)*CNDZ(K)
+ CNDR(J)*CNDR(K)) * DETJ
+ 144.000

```

```

1
2
CCNTINUE
CCNTINUE
CCNTINUE

```

ASSEMBLE SYSTEM INFLUENCE MATRIX W/OUT REGARD FOR BOUNDARY CONDITIONS

```

N(1) = N1
N(2) = N2
N(3) = N3
N(4) = N4
N(5) = N5
N(6) = N6
N(7) = N7
N(8) = N8
DO 350 IS = 1,NNE
1 = N(IS)
DC 350 JS = 1,NNE
J = N(JS)
EM(I,J) = EM(I,J) + EMS(IS,JS)
350 CONTINUE

```

ZEROS OUT EMS() FOR NEXT ELEMENT

FILE: TLPBG FCRTRAN A1 NAVAL POSTGRADUATE SCHOOL

C

C

C

CC

C

C

C

3

22

5

3

FILE: TURBO FORTRAN A1 NAVAL POSTGRADUATE SCHOOL

```
      DC 100 I = 1,NN
      IF (PSI(I).EQ.0.00) GOTO 110
      EPS = ABS(PSI(I) - PSI(I))/PSI(I)
      GOTO 120
110    EPS = ABS(PSI(I) - PSI(I))
120    IF (X.GT.EPS) GOTO 100
      X = EPS
100    CONTINUE
      RETURN
      END
```

APPENDIX G

SAMPLE PROGRAM OUTPUT

MEMORY SPACE AVAILABLE :
 REAL 8 = 499 INTEGER = 988 REAL 4 = 424

NASA TASK-1 TRANSONIC COMPRESSOR
 NO. OF NODES = 222 NO. OF ELEMENTS = 63
 NO. OF ROWS = 7 NO. OF COLUMNS = 9

NOCE	Z(I)	SUMMARY OF NODAL COORDINATES R(I)	S(I)	ABS FLOW ANG	REL FLOW ANG
1	0.0	0.1887800+02	0.9100000+00	0.0	0.0
2	0.0	0.1820000+02	0.9100000+00	0.0	0.0
3	0.0	0.1769240+02	0.9100000+00	0.0	0.0
4	0.0	0.1705320+02	0.9100000+00	0.0	0.0
5	0.0	0.1630990+02	0.9100000+00	0.0	0.0
6	0.0	0.1571940+02	0.9100000+00	0.0	0.0
7	0.0	0.1500910+02	0.9100000+00	0.0	0.0
8	0.0	0.1426140+02	0.9100000+00	0.0	0.0
9	0.0	0.1347340+02	0.9100000+00	0.0	0.0
10	0.0	0.1263430+02	0.9100000+00	0.0	0.0
11	0.0	0.1173970+02	0.9100000+00	0.0	0.0
12	0.0	0.1076960+02	0.9100000+00	0.0	0.0
13	0.0	0.9700910+01	0.9100000+00	0.0	0.0
14	0.0	0.8500100+01	0.9100000+00	0.0	0.0
15	0.0	0.7099000+01	0.9100000+00	0.0	0.0
16	0.0	0.5463100+01	0.9100000+00	0.0	0.0
17	0.0	0.3750210+01	0.9100000+00	0.0	0.0
18	0.0	0.1623790+01	0.9100000+00	0.0	0.0
19	0.0	0.1486660+01	0.9100000+00	0.0	0.0
20	0.0	0.1335520+01	0.9100000+00	0.0	0.0
21	0.0	0.1164330+01	0.9100000+00	0.0	0.0
22	0.0	0.9646280+00	0.9100000+00	0.0	0.0
23	0.0	0.7099000+00	0.9100000+00	0.0	0.0
24	0.0	0.4344000+00	0.9100000+00	0.0	0.0
25	0.0	0.1791240+00	0.9100000+00	0.0	0.0
26	0.0	0.1732160+00	0.9100000+00	0.0	0.0
27	0.0	0.1671060+00	0.9100000+00	0.0	0.0
28	0.0	0.1607740+00	0.9100000+00	0.0	0.0
29	0.0	0.1541540+00	0.9100000+00	0.0	0.0
30	0.0	0.1473520+00	0.9100000+00	0.0	0.0
31	0.0	0.1400100+00	0.9100000+00	0.0	0.0
32	0.0	0.1320720+00	0.9100000+00	0.0	0.0
33	0.0	0.1242650+00	0.9100000+00	0.0	0.0
34	0.0	0.1155910+00	0.9100000+00	0.0	0.0
35	0.0	0.1062120+00	0.9100000+00	0.0	0.0
36	0.0	0.9551210+00	0.9100000+00	0.0	0.0
37	0.0	0.8434020+00	0.9100000+00	0.0	0.0
38	0.0	0.7099000+00	0.9100000+00	0.0	0.0
39	0.0	0.4344000+00	0.9100000+00	0.0	0.0
40	0.0	0.1791240+00	0.9100000+00	0.0	0.0
41	0.0	0.1732160+00	0.9100000+00	0.0	0.0
42	0.0	0.1671060+00	0.9100000+00	0.0	0.0
43	0.0	0.1607740+00	0.9100000+00	0.0	0.0
44	0.0	0.1541540+00	0.9100000+00	0.0	0.0
45	0.0	0.1473520+00	0.9100000+00	0.0	0.0
46	0.0	0.1400100+00	0.9100000+00	0.0	0.0
47	0.0	0.1320720+00	0.9100000+00	0.0	0.0
48	0.0	0.1242650+00	0.9100000+00	0.0	0.0
49	0.0	0.1155910+00	0.9100000+00	0.0	0.0
50	0.0	0.1062120+00	0.9100000+00	0.0	0.0
51	0.0	0.9551210+00	0.9100000+00	0.0	0.0
52	0.0	0.8434020+00	0.9100000+00	0.0	0.0
53	0.0	0.7099000+00	0.9100000+00	0.0	0.0
54	0.0	0.4344000+00	0.9100000+00	0.0	0.0
55	0.0	0.1791240+00	0.9100000+00	0.0	0.0
56	0.0	0.1732160+00	0.9100000+00	0.0	0.0
57	0.0	0.1671060+00	0.9100000+00	0.0	0.0
58	0.0	0.1607740+00	0.9100000+00	0.0	0.0
59	0.0	0.1541540+00	0.9100000+00	0.0	0.0
60	0.0	0.1473520+00	0.9100000+00	0.0	0.0
61	0.0	0.1400100+00	0.9100000+00	0.0	0.0
62	0.0	0.1320720+00	0.9100000+00	0.0	0.0
63	0.0	0.1242650+00	0.9100000+00	0.0	0.0
64	0.0	0.1155910+00	0.9100000+00	0.0	0.0
65	0.0	0.1062120+00	0.9100000+00	0.0	0.0
66	0.0	0.9551210+00	0.9100000+00	0.0	0.0
67	0.0	0.8434020+00	0.9100000+00	0.0	0.0
68	0.0	0.7099000+00	0.9100000+00	0.0	0.0
69	0.0	0.4344000+00	0.9100000+00	0.0	0.0
70	0.0	0.1791240+00	0.9100000+00	0.0	0.0
71	0.0	0.1732160+00	0.9100000+00	0.0	0.0
72	0.0	0.1671060+00	0.9100000+00	0.0	0.0
73	0.0	0.1607740+00	0.9100000+00	0.0	0.0
74	0.0	0.1541540+00	0.9100000+00	0.0	0.0
75	0.0	0.1473520+00	0.9100000+00	0.0	0.0
76	0.0	0.1400100+00	0.9100000+00	0.0	0.0
77	0.0	0.1320720+00	0.9100000+00	0.0	0.0
78	0.0	0.1242650+00	0.9100000+00	0.0	0.0
79	0.0	0.1155910+00	0.9100000+00	0.0	0.0
80	0.0	0.1062120+00	0.9100000+00	0.0	0.0
81	0.0	0.9551210+00	0.9100000+00	0.0	0.0
82	0.0	0.8434020+00	0.9100000+00	0.0	0.0
83	0.0	0.7099000+00	0.9100000+00	0.0	0.0
84	0.0	0.4344000+00	0.9100000+00	0.0	0.0
85	0.0	0.1791240+00	0.9100000+00	0.0	0.0
86	0.0	0.1732160+00	0.9100000+00	0.0	0.0
87	0.0	0.1671060+00	0.9100000+00	0.0	0.0
88	0.0	0.1607740+00	0.9100000+00	0.0	0.0
89	0.0	0.1541540+00	0.9100000+00	0.0	0.0
90	0.0	0.1473520+00	0.9100000+00	0.0	0.0
91	0.0	0.1400100+00	0.9100000+00	0.0	0.0
92	0.0	0.1320720+00	0.9100000+00	0.0	0.0
93	0.0	0.1242650+00	0.9100000+00	0.0	0.0
94	0.0	0.1155910+00	0.9100000+00	0.0	0.0
95	0.0	0.1062120+00	0.9100000+00	0.0	0.0
96	0.0	0.9551210+00	0.9100000+00	0.0	0.0
97	0.0	0.8434020+00	0.9100000+00	0.0	0.0
98	0.0	0.7099000+00	0.9100000+00	0.0	0.0
99	0.0	0.4344000+00	0.9100000+00	0.0	0.0
100	0.0	0.1791240+00	0.9100000+00	0.0	0.0
101	0.0	0.1732160+00	0.9100000+00	0.0	0.0
102	0.0	0.1671060+00	0.9100000+00	0.0	0.0
103	0.0	0.1607740+00	0.9100000+00	0.0	0.0
104	0.0	0.1541540+00	0.9100000+00	0.0	0.0
105	0.0	0.1473520+00	0.9100000+00	0.0	0.0
106	0.0	0.1400100+00	0.9100000+00	0.0	0.0
107	0.0	0.1320720+00	0.9100000+00	0.0	0.0
108	0.0	0.1242650+00	0.9100000+00	0.0	0.0
109	0.0	0.1155910+00	0.9100000+00	0.0	0.0
110	0.0	0.1062120+00	0.9100000+00	0.0	0.0
111	0.0	0.9551210+00	0.9100000+00	0.0	0.0
112	0.0	0.8434020+00	0.9100000+00	0.0	0.0
113	0.0	0.7099000+00	0.9100000+00	0.0	0.0
114	0.0	0.4344000+00	0.9100000+00	0.0	0.0
115	0.0	0.1791240+00	0.9100000+00	0.0	0.0
116	0.0	0.1732160+00	0.9100000+00	0.0	0.0
117	0.0	0.1671060+00	0.9100000+00	0.0	0.0
118	0.0	0.1607740+00	0.9100000+00	0.0	0.0
119	0.0	0.1541540+00	0.9100000+00	0.0	0.0
120	0.0	0.1473520+00	0.9100000+00	0.0	0.0
121	0.0	0.1400100+00	0.9100000+00	0.0	0.0
122	0.0	0.1320720+00	0.9100000+00	0.0	0.0
123	0.0	0.1242650+00	0.9100000+00	0.0	0.0
124	0.0	0.1155910+00	0.9100000+00	0.0	0.0
125	0.0	0.1062120+00	0.9100000+00	0.0	0.0
126	0.0	0.9551210+00	0.9100000+00	0.0	0.0
127	0.0	0.8434020+00	0.9100000+00	0.0	0.0
128	0.0	0.7099000+00	0.9100000+00	0.0	0.0
129	0.0	0.4344000+00	0.9100000+00	0.0	0.0
130	0.0	0.1791240+00	0.9100000+00	0.0	0.0
131	0.0	0.1732160+00	0.9100000+00	0.0	0.0
132	0.0	0.1671060+00	0.9100000+00	0.0	0.0
133	0.0	0.1607740+00	0.9100000+00	0.0	0.0
134	0.0	0.1541540+00	0.9100000+00	0.0	0.0
135	0.0	0.1473520+00	0.9100000+00	0.0	0.0
136	0.0	0.1400100+00	0.9100000+00	0.0	0.0
137	0.0	0.1320720+00	0.9100000+00	0.0	0.0
138	0.0	0.1242650+00	0.9100000+00	0.0	0.0
139	0.0	0.1155910+00	0.9100000+00	0.0	0.0
140	0.0	0.1062120+00	0.9100000+00	0.0	0.0
141	0.0	0.9551210+00	0.9100000+00	0.0	0.0
142	0.0	0.8434020+00	0.9100000+00	0.0	0.0
143	0.0	0.7099000+00	0.9100000+00	0.0	0.0
144	0.0	0.4344000+00	0.9100000+00	0.0	0.0
145	0.0	0.1791240+00	0.9100000+00	0.0	0.0
146	0.0	0.1732160+00	0.9100000+00	0.0	0.0
147	0.0	0.1671060+00	0.9100000+00	0.0	0.0
148	0.0	0.1607740+00	0.9100000+00	0.0	0.0
149	0.0	0.1541540+00	0.9100000+00	0.0	0.0
150	0.0	0.1473520+00	0.9100000+00	0.0	0.0
151	0.0	0.1400100+00	0.9100000+00	0.0	0.0
152	0.0	0.1320720+00	0.9100000+00	0.0	0.0
153	0.0	0.1242650+00	0.9100000+00	0.0	0.0
154	0.0	0.1155910+00	0.9100000+00	0.0	0.0
155	0.0	0.1062120+00	0.9100000+00	0.0	0.0
156	0.0	0.9551210+00	0.9100000+00	0.0	0.0
157	0.0	0.8434020+00	0.9100000+00	0.0	0.0
158	0.0	0.7099000+00	0.9100000+00	0.0	0.0
159	0.0	0.4344000+00	0.9100000+00	0.0	0.0
160	0.0	0.1791240+00	0.9100000+00	0.0	0.0
161	0.0	0.1732160+00	0.9100000+00	0.0	0.0
162	0.0	0.1671060+00	0.9100000+00	0.0	0.0
163	0.0	0.1607740+00	0.9100000+00	0.0	0.0
164	0.0	0.1541540+00	0.9100000+00	0.0	0.0
165	0.0	0.1473520+00	0.9100000+00	0.0	0.0
166	0.0	0.1400100+00	0.9100000+00	0.0	0.0
167	0.0	0.1320720+00	0.9100000+00	0.0	0.0
168	0.0	0.1242650+00	0.9100000+00	0.0	0.0
169	0.0	0.1155910+00	0.9100000+00	0.0	0.0
170	0.0	0.1062120+00	0.9100000+00	0.0	0.0
171	0.0	0.9551210+00	0.9100000+00	0.0</	

NODE	Z(I)	SUMMARY OF NODAL COORDINATES		ABS FLOW ANG	REL FLOW ANG
		R(I)	B(I)		
41	C.45000000+01	C.1504490+02	C.91000000+00	0.0	0.0
42	C.45000000+01	C.1469790+02	C.91000000+00	0.0	0.0
43	C.45000000+01	C.1321440+02	C.91000000+00	0.0	0.0
44	C.45000000+01	C.1154180+02	C.91000000+00	0.0	0.0
45	C.45000000+01	C.9561450+01	C.91000000+00	0.0	0.0
46	C.45000000+01	C.7099000+01	C.91000000+00	0.0	0.0
47	C.60000000+01	C.1840300+02	C.91000000+00	0.0	0.0
48	C.60000000+01	C.1783340+02	C.91000000+00	0.0	0.0
49	C.60000000+01	C.1725240+02	C.91000000+00	0.0	0.0
50	C.60000000+01	C.1664460+02	C.91000000+00	0.0	0.0
51	C.60000000+01	C.1601370+02	C.91000000+00	0.0	0.0
52	C.60000000+01	C.1535630+02	C.91000000+00	0.0	0.0
53	C.60000000+01	C.1467070+02	C.91000000+00	0.0	0.0
54	C.60000000+01	C.1395680+02	C.91000000+00	0.0	0.0
55	C.60000000+01	C.1312170+02	C.91000000+00	0.0	0.0
56	C.60000000+01	C.1233270+02	C.91000000+00	0.0	0.0
57	C.60000000+01	C.1152440+02	C.91000000+00	0.0	0.0
58	C.60000000+01	C.1069280+02	C.91000000+00	0.0	0.0
59	C.60000000+01	C.9571010+01	C.91000000+00	0.0	0.0
60	C.60000000+01	C.8426150+01	C.91000000+00	0.0	0.0
61	C.60000000+01	C.7099000+01	C.91000000+00	0.0	0.0
62	C.65020000+01	C.1340250+02	C.91000000+00	0.0	0.0
63	C.65020000+01	C.1724740+02	C.91000000+00	0.0	0.0
64	C.65020000+01	C.1500320+02	C.91000000+00	0.0	0.0
65	C.65020000+01	C.1356690+02	C.91000000+00	0.0	0.0
66	C.65020000+01	C.1316860+02	C.91000000+00	0.0	0.0
67	C.65020000+01	C.1152210+02	C.91000000+00	0.0	0.0
68	C.65020000+01	C.9563810+01	C.91000000+00	0.0	0.0
69	C.65020000+01	C.7099000+01	C.91000000+00	0.0	0.0
70	C.11804000+02	C.1340250+02	C.91000000+00	0.0	0.0
71	C.11804000+02	C.1724740+02	C.91000000+00	0.0	0.0
72	C.11804000+02	C.1500320+02	C.91000000+00	0.0	0.0
73	C.11804000+02	C.1356690+02	C.91000000+00	0.0	0.0
74	C.11804000+02	C.1316860+02	C.91000000+00	0.0	0.0
75	C.11804000+02	C.1152210+02	C.91000000+00	0.0	0.0
76	C.11804000+02	C.9563810+01	C.91000000+00	0.0	0.0
77	C.11804000+02	C.7099000+01	C.91000000+00	0.0	0.0
78	C.11804000+02	C.1316860+02	C.91000000+00	0.0	0.0
79	C.11804000+02	C.1233270+02	C.91000000+00	0.0	0.0
80	C.11804000+02	C.1152440+02	C.91000000+00	0.0	0.0

NODE	Z(I)	SUMMARY OF NODAL COORDINATES		ABS FLOW ANG	REL FLOW ANG
		R(I)	3(I)		
81	0.118040T+02	0.1059930+02	0.7100000+00	0.0	0.0
82	0.118040C+02	0.3508627+01	0.9100000+00	0.0	0.0
83	0.118040C+02	0.8425221+01	0.9100000+00	0.0	0.0
84	0.118040T+02	0.7100000+01	0.7100000+00	0.0	0.0
85	0.128091T+02	0.1834357+02	0.9100000+00	0.0	0.0
86	0.128091C+02	0.1733540+02	0.9100000+00	0.0	0.0
87	0.128091C+02	0.16003940+02	0.9100000+00	0.0	0.0
88	0.128091T+02	0.1472180+02	0.9100000+00	0.0	0.0
89	0.128091C+02	0.1274171+02	0.9100000+00	0.0	0.0
90	0.128091C+02	0.1164730+02	0.9100000+00	0.0	0.0
91	0.128091C+02	0.9784073+01	0.9100000+00	0.0	0.0
92	0.1300000+02	0.7390000+01	0.9100000+00	0.0	0.0
93	0.1300000+02	0.1820000+01	0.9100000+00	0.0	0.0
94	0.1300000+02	0.1743000+01	0.9100000+00	0.0	0.0
95	0.1300000+02	0.1727150+01	0.9100000+00	0.0	0.0
96	0.1300000+02	0.1668430+01	0.9100000+00	0.0	0.0
97	0.1300000+02	0.1607340+01	0.9100000+00	0.0	0.0
98	0.1300000+02	0.1544260+01	0.9100000+00	0.0	0.0
99	0.1300000+02	0.1478380+01	0.9100000+00	0.0	0.0
100	0.1300000+02	0.1400000+01	0.9100000+00	0.0	0.0
101	0.1300000+02	0.1336550+01	0.9100000+00	0.0	0.0
102	0.1300000+02	0.1286975+01	0.9100000+00	0.0	0.0
103	0.1300000+02	0.1217790+01	0.9100000+00	0.0	0.0
104	0.1300000+02	0.1150000+01	0.9100000+00	0.0	0.0
105	0.1300000+02	0.9963310+00	0.9100000+00	0.0	0.0
106	0.1300000+02	0.8384030+00	0.9100000+00	0.0	0.0
107	0.1300000+02	0.7580000+00	0.9100000+00	0.0	0.0
108	0.1300000+02	0.6200000+00	0.9100000+00	0.0	0.0
109	0.1300000+02	0.4722880+00	0.9100000+00	0.0	0.0
110	0.1300000+02	0.3500000+00	0.9100000+00	0.0	0.0
111	0.1300000+02	0.2430000+00	0.9100000+00	0.0	0.0
112	0.1300000+02	0.1355400+00	0.9100000+00	0.0	0.0
113	0.1300000+02	0.1200120+00	0.9100000+00	0.0	0.0
114	0.1300000+02	0.1040070+00	0.9100000+00	0.0	0.0
115	0.1300000+02	0.7300000+00	0.9100000+00	0.0	0.0
116	0.1300000+02	0.1410000+00	0.9100000+00	0.0	0.0
117	0.1300000+02	0.1275672+00	0.9024430+00	0.0	0.0
118	0.1300000+02	0.1171020+00	0.9019640+00	0.0	0.0
119	0.1300000+02	0.1566780+00	0.9014400+00	0.0	0.0
120	0.1300000+02	0.1612860+02	0.9003520+00	0.0	0.0

NODE	Z(I)	SUMMARY OF MODAL COORDINATES		ABS FLOW ANG	REL FLOW ANG
		R(I)	B(I)		
121	0.183295E+02	0.155706D+02	0.900157D+00	0.0	0.0
122	0.182302E+02	0.149415E+02	0.399463D+00	0.0	0.0
123	0.182555E+02	0.143400E+02	0.398637D+00	0.0	0.0
124	0.183190E+02	0.137617E+02	0.397699D+00	0.0	0.0
125	0.181925E+02	0.131014E+02	0.396324E+00	0.0	0.0
126	0.181450E+02	0.124101E+02	0.395385E+00	0.0	0.0
127	0.181095E+02	0.116758E+02	0.393944E+00	0.0	0.0
128	0.180730E+02	0.108321E+02	0.393245E+00	0.0	0.0
129	0.180365E+02	0.100475E+02	0.393021E+00	0.0	0.0
130	0.180000E+02	0.912500E+01	0.391774E+00	0.0	0.0
131	0.179635E+02	0.803050E+02	0.390122E+00	0.0	0.0
132	0.179270E+02	0.704550E+02	0.389033E+00	0.0	0.0
133	0.178905E+02	0.605900E+02	0.388378E+00	0.0	0.0
134	0.178540E+02	0.506190E+02	0.387550E+00	0.0	0.0
135	0.178175E+02	0.406320E+02	0.386440E+00	0.0	0.0
136	0.177810E+02	0.306162E+02	0.385339E+00	0.0	0.0
137	0.177445E+02	0.205324E+02	0.384250E+00	0.0	0.0
138	0.177080E+02	0.104300E+02	0.383041E+00	0.0	0.0
139	0.176715E+02	0.953300E+01	0.382237E+00	0.0	0.0
140	0.176350E+02	0.853300E+02	0.381030E+00	0.0	0.0
141	0.175985E+02	0.753300E+02	0.380199E+00	0.0	0.0
142	0.175620E+02	0.653300E+02	0.379144E+00	0.0	0.0
143	0.175255E+02	0.553300E+02	0.378085E+00	0.0	0.0
144	0.174890E+02	0.453300E+02	0.376920E+00	0.0	0.0
145	0.174525E+02	0.353300E+02	0.375660E+00	0.0	0.0
146	0.174160E+02	0.253300E+02	0.374390E+00	0.0	0.0
147	0.173795E+02	0.153300E+02	0.373120E+00	0.0	0.0
148	0.173430E+02	0.053300E+02	0.371850E+00	0.0	0.0
149	0.173065E+02	0.000000E+00	0.370580E+00	0.0	0.0
150	0.172700E+02	0.000000E+00	0.369310E+00	0.0	0.0
151	0.172335E+02	0.000000E+00	0.368040E+00	0.0	0.0
152	0.171970E+02	0.000000E+00	0.366770E+00	0.0	0.0
153	0.171605E+02	0.000000E+00	0.365500E+00	0.0	0.0
154	0.171240E+02	0.000000E+00	0.364230E+00	0.0	0.0
155	0.170875E+02	0.000000E+00	0.362960E+00	0.0	0.0
156	0.170510E+02	0.000000E+00	0.361690E+00	0.0	0.0
157	0.170145E+02	0.000000E+00	0.360420E+00	0.0	0.0
158	0.169780E+02	0.000000E+00	0.359150E+00	0.0	0.0
159	0.169415E+02	0.000000E+00	0.357880E+00	0.0	0.0
160	0.169050E+02	0.000000E+00	0.356610E+00	0.0	0.0

ACCE	Z(I)	SUMMARY OF NODAL COORDINATES		ABS FLOW ANG	REL FLOW ANG
		R(I)	B(I)		
161	00	C.1035300+02	C.9100000+00	0.0	0.0
162	00	C.1783500+02	C.3900600+00	0.0	0.0
163	00	C.1742110+02	C.8929320+00	0.0	0.0
164	00	C.1655500+02	C.8987750+00	0.0	0.0
165	00	C.1656010+02	C.3025930+00	0.0	0.0
166	00	C.1611230+02	C.8023830+00	0.0	0.0
167	00	C.1585180+02	C.4091430+00	0.0	0.0
168	00	C.1517720+02	C.3978730+00	0.0	0.0
169	00	C.1468740+02	C.3975730+00	0.0	0.0
170	00	C.1418060+02	C.8072400+00	0.0	0.0
171	00	C.1365510+02	C.8068780+00	0.0	0.0
172	00	C.1310850+02	C.8064830+00	0.0	0.0
173	00	C.1253810+02	C.8060510+00	0.0	0.0
174	00	C.1194040+02	C.8055730+00	0.0	0.0
175	00	C.1131130+02	C.8050470+00	0.0	0.0
176	00	C.1064500+02	C.8044370+00	0.0	0.0
177	00	C.1003600+02	C.8038970+00	0.0	0.0
178	00	C.1701650+02	C.3217730+00	0.0	0.0
179	00	C.1615570+02	C.3224070+00	0.0	0.0
180	00	C.1534730+02	C.3230350+00	0.0	0.0
181	00	C.1437330+02	C.3233420+00	0.0	0.0
182	00	C.1324120+02	C.3233610+00	0.0	0.0
183	00	C.1211490+02	C.3229730+00	0.0	0.0
184	00	C.1087250+02	C.3219590+00	0.0	0.0
185	00	C.1783500+02	C.3501100+00	0.0	0.0
186	00	C.1742110+02	C.3518920+00	0.0	0.0
187	00	C.1703760+02	C.3517310+00	0.0	0.0
188	00	C.1642260+02	C.3535620+00	0.0	0.0
189	00	C.1561040+02	C.3541390+00	0.0	0.0
190	00	C.1517420+02	C.3549150+00	0.0	0.0
191	00	C.1431630+02	C.3573620+00	0.0	0.0
192	00	C.1405480+02	C.3576750+00	0.0	0.0
193	00	C.1337860+02	C.3573290+00	0.0	0.0
194	00	C.1331600+02	C.3564730+00	0.0	0.0
195	00	C.1337950+02	C.3565350+00	0.0	0.0
196	00	C.1254430+02	C.3560610+00	0.0	0.0
197	00	C.1199040+02	C.3545500+00	0.0	0.0
198	00	C.1171040+02	C.3549510+00	0.0	0.0
199	00	C.1110000+02	C.3543320+00	0.0	0.0
200	00	C.1783500+02	C.9100000+00	0.0	0.0

NOCE	Z(I)	SUMMARY OF NODAL COORDINATES R(I)	H(I)	ABS FLOW	ANG	REL FLOW	ANG
201	C	2547000+02	C	1704000+02	C	9100000+00	0.0
202	C	2561000+02	C	1520675+02	C	9100000+00	0.0
203	C	2561000+02	C	1533720+02	C	9100000+00	0.0
204	C	2561000+02	C	1436400+02	C	9100000+00	0.0
205	C	2555000+02	C	1533600+02	C	9100000+00	0.0
206	C	2552000+02	C	1231700+02	C	9100000+00	0.0
207	C	2549000+02	C	1113500+02	C	9100000+00	0.0
208	C	2536000+02	C	1783600+02	C	9100000+00	0.0
209	C	2536000+02	C	1744450+02	C	9100000+00	0.0
210	C	2536000+02	C	1704400+02	C	9100000+00	0.0
211	C	2536000+02	C	1663400+02	C	9100000+00	0.0
212	C	2536000+02	C	1433350+02	C	9100000+00	0.0
213	C	2536000+02	C	1574190+02	C	9100000+00	0.0
214	C	2536000+02	C	1533400+02	C	9100000+00	0.0
215	C	2536000+02	C	1409100+02	C	9100000+00	0.0
216	C	2536000+02	C	1440950+02	C	9100000+00	0.0
217	C	2536000+02	C	1352200+02	C	9100000+00	0.0
218	C	2536000+02	C	1241500+02	C	9100000+00	0.0
219	C	2536000+02	C	1241500+02	C	9100000+00	0.0
220	C	2536000+02	C	1244470+02	C	9100000+00	0.0
221	C	2536000+02	C	1177200+02	C	9100000+00	0.0
222	C	2536000+02	C	1117000+02	C	9100000+00	0.0

ELEMENT	NOTES	TYPE OF ELEMENT
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10
11	11	11
12	12	12
13	13	13
14	14	14
15	15	15
16	16	16
17	17	17
18	18	18
19	19	19
20	20	20
21	21	21
22	22	22
23	23	23
24	24	24
25	25	25
26	26	26
27	27	27
28	28	28
29	29	29
30	30	30
31	31	31
32	32	32
33	33	33
34	34	34
35	35	35
36	36	36
37	37	37
38	38	38
39	39	39
40	40	40
41	41	41
42	42	42
43	43	43
44	44	44
45	45	45
46	46	46
47	47	47
48	48	48
49	49	49
50	50	50
51	51	51
52	52	52
53	53	53
54	54	54
55	55	55
56	56	56
57	57	57
58	58	58
59	59	59
60	60	60
61	61	61
62	62	62
63	63	63
64	64	64
65	65	65
66	66	66
67	67	67
68	68	68
69	69	69
70	70	70
71	71	71
72	72	72
73	73	73
74	74	74
75	75	75
76	76	76
77	77	77
78	78	78
79	79	79
80	80	80
81	81	81
82	82	82
83	83	83
84	84	84
85	85	85
86	86	86
87	87	87
88	88	88
89	89	89
90	90	90
91	91	91
92	92	92
93	93	93
94	94	94
95	95	95
96	96	96
97	97	97
98	98	98
99	99	99
100	100	100
101	101	101
102	102	102
103	103	103
104	104	104
105	105	105
106	106	106
107	107	107
108	108	108
109	109	109
110	110	110
111	111	111
112	112	112
113	113	113
114	114	114
115	115	115
116	116	116
117	117	117
118	118	118
119	119	119
120	120	120
121	121	121
122	122	122
123	123	123
124	124	124
125	125	125
126	126	126
127	127	127
128	128	128
129	129	129
130	130	130
131	131	131
132	132	132
133	133	133
134	134	134
135	135	135
136	136	136
137	137	137
138	138	138
139	139	139
140	140	140
141	141	141
142	142	142
143	143	143
144	144	144
145	145	145
146	146	146
147	147	147
148	148	148
149	149	149
150	150	150
151	151	151

INLET THERMODYNAMIC VARIABLES ARE AS FOLLOWS

FLOW RATE = 0.17454×10^{-3} LBM/SEC
TOT DENSITY = 0.851884×10^{-1} LBM CU FT
TOT PRESSURE = 0.175000×10^2 PSI
TOT TEMPERATURE = 0.555000×10^3 DEG RANKINE
ROTATIONAL SPEED = 0.697520×10^4 RPM
INLET U VELOCITY = 0.306922×10^3 FT/SEC
GAS CONSTANT = 0.533000×10^2
RATIO OF SPECIFIC HEATS = 0.140000×10^1
SPECIFIC HEAT AT CONSTANT PRESSURE = 0.240000×10^3
STATIC DENSITY AT INLET = 0.922124×10^{-1}

ACOE, INC.

PS (11)

0.2777890	+02
0.2777940	+02
0.2778000	+02
0.2778050	+02
0.2778100	+02
0.2778150	+02
0.2778200	+02
0.2778250	+02
0.2778300	+02
0.2778350	+02
0.2778400	+02
0.2778450	+02
0.2778500	+02
0.2778550	+02
0.2778600	+02
0.2778650	+02
0.2778700	+02
0.2778750	+02
0.2778800	+02
0.2778850	+02
0.2778900	+02
0.2778950	+02
0.2779000	+02
0.2779050	+02
0.2779100	+02
0.2779150	+02
0.2779200	+02
0.2779250	+02
0.2779300	+02
0.2779350	+02
0.2779400	+02
0.2779450	+02
0.2779500	+02
0.2779550	+02
0.2779600	+02
0.2779650	+02
0.2779700	+02
0.2779750	+02
0.2779800	+02
0.2779850	+02
0.2779900	+02
0.2779950	+02
0.2780000	+02
0.2780050	+02
0.2780100	+02
0.2780150	+02
0.2780200	+02
0.2780250	+02
0.2780300	+02
0.2780350	+02
0.2780400	+02
0.2780450	+02
0.2780500	+02
0.2780550	+02
0.2780600	+02
0.2780650	+02
0.2780700	+02
0.2780750	+02
0.2780800	+02
0.2780850	+02
0.2780900	+02
0.2780950	+02
0.2781000	+02
0.2781050	+02
0.2781100	+02
0.2781150	+02
0.2781200	+02
0.2781250	+02
0.2781300	+02
0.2781350	+02
0.2781400	+02
0.2781450	+02
0.2781500	+02
0.2781550	+02
0.2781600	+02
0.2781650	+02
0.2781700	+02
0.2781750	+02
0.2781800	+02
0.2781850	+02
0.2781900	+02
0.2781950	+02
0.2782000	+02
0.2782050	+02
0.2782100	+02
0.2782150	+02
0.2782200	+02
0.2782250	+02
0.2782300	+02
0.2782350	+02
0.2782400	+02
0.2782450	+02
0.2782500	+02
0.2782550	+02
0.2782600	+02
0.2782650	+02
0.2782700	+02
0.2782750	+02
0.2782800	+02
0.2782850	+02
0.2782900	+02
0.2782950	+02
0.2783000	+02
0.2783050	+02
0.2783100	+02
0.2783150	+02
0.2783200	+02
0.2783250	+02
0.2783300	+02
0.2783350	+02
0.2783400	+02
0.2783450	+02
0.2783500	+02
0.2783550	+02
0.2783600	+02
0.2783650	+02
0.2783700	+02
0.2783750	+02
0.2783800	+02
0.2783850	+02
0.2783900	+02
0.2783950	+02
0.2784000	+02
0.2784050	+02
0.2784100	+02
0.2784150	+02
0.2784200	+02
0.2784250	+02
0.2784300	+02
0.2784350	+02
0.2784400	+02
0.2784450	+02
0.2784500	+02
0.2784550	+02
0.2784600	+02
0.2784650	+02
0.2784700	+02
0.2784750	+02
0.2784800	+02
0.2784850	+02
0.2784900	+02
0.2784950	+02
0.2785000	+02
0.2785050	+02
0.2785100	+02
0.2785150	+02
0.2785200	+02
0.2785250	+02
0.2785300	+02
0.2785350	+02
0.2785400	+02
0.2785450	+02
0.2785500	+02
0.2785550	+02
0.2785600	+02
0.2785650	+02
0.2785700	+02
0.2785750	+02
0.2785800	+02
0.2785850	+02
0.2785900	+02
0.2785950	+02
0.2786000	+02
0.27	

LARGEST EPS FOR ITERATION 1 IS 0.346759559206D+00
 ITERATION NO. 1 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 2 IS 0.204902273624D+00
 ITERATION NO. 2 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 3 IS 0.135922144523D+00
 ITERATION NO. 3 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 4 IS 0.941426824100D-01
 ITERATION NO. 4 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 5 IS 0.662198640489D-01
 ITERATION NO. 5 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 6 IS 0.469498793800D-01
 ITERATION NO. 6 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 7 IS 0.342379259092D-01
 ITERATION NO. 7 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 LARGEST EPS FOR ITERATION 8 IS 0.254317279121D-01
 ITERATION NO. 8 COMPLETE
 STREAM FUNCTION CONVERGENCE NOT YET SATISFIED.
 NEXT ITERATION IS IN PROGRESS
 PROGRAM TERMINATED ON ITERATION NO. 9
 RESULTS WHICH FOLLOW ARE FOR CONVERGENCE EPSILON = 0.189017741652D-01
 STREAM FUNCTION CONVERGENCE CRITERION SATISFIED ON ITERATION NUMBER 9
 RESULTS ARE AS FOLLOWS FOR CONVERGENCE EPSILON = 0.189017741652D-01

FINITE ELEMENT RESULTS

NODE	PSI(1)	VZ	VR	R(1)	DENSITY
1	277794	306522D+03	0.0	1.84780D+02	0.8231240-01
2	25794	306922D+03	0.0	1.82900D+02	0.8231240-01
3	238105E+C2	306522D+03	0.0	0.176824D+02	0.8231240-01
4	218247E+C2	306522D+03	0.0	0.170532D+02	0.8231240-01
5	198421E+C2	306522D+03	0.0	0.163959D+02	0.8231240-01
6	178579E+C2	306522D+03	0.0	0.157194D+02	0.8231240-01
7	158737E+C2	306522D+03	0.0	0.150331D+02	0.8231240-01
8	138895E+C2	306522D+03	0.0	0.142614D+02	0.8231240-01
9	119052E+C2	306522D+03	0.0	0.134774D+02	0.8231240-01
10	992104E+C1	306522D+03	0.0	0.126367D+02	0.8231240-01
11	793683E+C1	306522D+03	0.0	0.117665D+02	0.8231240-01
12	595262E+C1	306522D+03	0.0	0.970061D+01	0.8231240-01
13	396841E+C1	306522D+03	0.0	0.709500E+01	0.8231240-01
14	198421E+C1	306522D+03	0.0	0.468100D+02	0.8231240-01
15	277794	306522D+03	-0.450541D+02	0.175031D+02	0.8231240-01
16	25794	306922D+03	-0.367279D+02	0.162375D+02	0.8231240-01
17	238105E+C2	306522D+03	-0.277484D+02	0.148666D+02	0.8231240-01
18	218247E+C2	306522D+03	-0.211333D+02	0.133552D+02	0.8231240-01
19	198421E+C2	306522D+03	-0.155310D+02	0.116453D+02	0.8231240-01
20	178579E+C2	306522D+03	-0.105989D+02	0.964623D+01	0.8231240-01
21	158737E+C2	306522D+03	-0.005567D+01	0.739800D+01	0.8231240-01
22	138895E+C2	306522D+03	-0.533389D+02	0.548400D+02	0.8231240-01
23	119052E+C2	306522D+03	-0.411451D+02	0.424240D+02	0.8231240-01
24	992104E+C1	306522D+03	-0.292220D+02	0.333150D+02	0.8231240-01
25	793683E+C1	306522D+03	-0.205330D+02	0.267100D+02	0.8231240-01
26	595262E+C1	306522D+03	-0.228207D+02	0.160161D+02	0.8231240-01
27	396841E+C1	306522D+03	-0.209104D+02	0.154154D+02	0.8231240-01
28	277794	306522D+03	-0.173725D+02	0.147752D+02	0.8231240-01
29	25794	306922D+03	-0.153950D+02	0.140010D+02	0.8231240-01
30	238105E+C2	306522D+03	-0.132974D+02	0.132720D+02	0.8231240-01
31	218247E+C2	306522D+03	-0.110533D+02	0.124265D+02	0.8231240-01
32	198421E+C2	306522D+03	-0.933358D+01	0.115561D+02	0.8231240-01
33	178579E+C2	306522D+03	-0.630627D+01	0.106212D+02	0.8231240-01
34	158737E+C2	306522D+03	-0.545475D+01	0.955191D+01	0.8231240-01
35	138895E+C2	306522D+03	-0.355355D+01	0.843382D+01	0.8231240-01
36	119052E+C1	306522D+03	-0.279920D+01	0.709200D+01	0.8231240-01
37	992104E+C1	306522D+03	-0.355273D+01	0.544400D+02	0.8231240-01
38	793683E+C1	306522D+03	-0.128843D+02	0.172872D+02	0.8231240-01

FINITE ELEMENT RESULTS

ACDE	PSI(1)	VZ	VR	R(1)	DENSITY
41	0.196244C+02	0.382613D+03	-C.1169E3D+02	0.160445D+02	0.8C5813D-01
42	0.156204C+02	0.377925D+03	-C.945150D+01	0.146975D+02	0.8C6931D-01
43	0.116615C+02	0.372201D+03	-C.620540C+01	0.132144D+02	0.8C7759D-01
44	0.77510C+01	0.371532D+03	-C.413513D+01	0.115415D+02	0.8C8282D-01
45	0.3E55E2C+01	0.371478D+03	-C.0	0.709930D+01	0.8C8441D-01
46	0.2777895C+02	0.393032D+03	-C.776881D+01	0.179266D+02	0.8C8768D-01
47	0.257762C+02	0.391295D+03	-C.776881D+01	0.179266D+02	0.8C8768D-01
48	0.238778C+02	0.388416D+03	-C.776881D+01	0.179266D+02	0.8C8768D-01
49	0.216530C+02	0.355336D+03	-C.505014D+01	0.160127D+02	0.8C9154D-01
50	0.196321C+02	0.346141D+03	-C.505014D+01	0.160127D+02	0.8C9154D-01
51	0.176225C+02	0.301493D+03	-C.515480D+01	0.152369D+02	0.8C9537D-01
52	0.156231C+02	0.301038D+03	-C.333970D+01	0.146707D+02	0.8C9611D-01
53	0.136341C+02	0.377131D+03	-C.333970D+01	0.131017D+02	0.8C9705D-01
54	0.116541C+02	0.377131D+03	-C.333970D+01	0.131017D+02	0.8C9705D-01
55	0.985640C+01	0.377131D+03	-C.333970D+01	0.131017D+02	0.8C9705D-01
56	0.777230C+01	0.377131D+03	-C.333970D+01	0.131017D+02	0.8C9705D-01
57	0.577230C+01	0.368402D+03	-C.149630D+01	0.105978D+02	0.8C9705D-01
58	0.377230C+01	0.359431D+03	-C.149630D+01	0.105978D+02	0.8C9705D-01
59	0.192720C+01	0.377131D+03	-C.149630D+01	0.105978D+02	0.8C9705D-01
60	0.2777895C+02	0.377131D+03	-C.149630D+01	0.105978D+02	0.8C9705D-01
61	0.237110C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
62	0.190522C+02	0.349020D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
63	0.156144C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
64	0.116112C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
65	0.165951C+01	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
66	0.379367C+01	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
67	0.7777895C+02	0.377131D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
68	0.257210C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
69	0.216066C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
70	0.174449C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
71	0.154453C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
72	0.134925C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
73	0.113773C+02	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
74	0.935883C+01	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01
75	0.730830C+01	0.349150D+03	-C.139940D+00	0.123462D+02	0.8C9705D-01

FINITE ELEMENT RESULTS

ACCE	PSI(11)	VZ	VR	R(11)	DENSITY
1	540672C+01	367249D+03	0.262556D+02	0.105893C+02	0.8092C8D-01
2	349044C+01	0.361666D+03	0.205554D+02	0.842522D+01	0.810413D-01
3	168667E+01	0.325666D+03	0.102820D+02	0.842522D+01	0.810413D-01
4	77725D+02	0.313548D+03	0.549917D-01	0.193935D+02	0.802725D-01
5	36975D+02	0.306752D+03	0.127650D+01	0.172556D+02	0.801903D-01
6	196042C+02	0.306752D+03	0.549917D-01	0.160340D+02	0.801676D-01
7	155084C+02	0.355340D+03	0.147315D+02	0.147218D+02	0.801676D-01
8	114395C+02	0.355340D+03	0.251550D+02	0.132741C+02	0.802552D-01
9	143277E+01	0.386224D+03	0.342357D+02	0.116478D+02	0.804547D-01
10	357495C+01	0.373813D+03	0.538590D+02	0.975477D+01	0.806949D-01
11	77785C+02	0.331067D+03	0.342357D+02	0.739000D+02	0.814850D-01
12	57725C+02	0.396727D+03	0.127650D+01	0.183300D+02	0.802458D-01
13	237615C+02	0.406828D+03	0.109316D+01	0.172715C+02	0.802458D-01
14	217125C+02	0.405470D+03	0.151249D+01	0.166942D+02	0.802458D-01
15	156675C+02	0.407428D+03	0.636444C+01	0.160754C+02	0.799755D-01
16	175183C+02	0.405470D+03	0.111751D+02	0.154430D+02	0.800097D-01
17	155057C+02	0.406902D+03	0.167520D+02	0.147328D+02	0.795853D-01
18	135226C+02	0.403345D+03	0.225335D+02	0.140921D+02	0.800669D-01
19	114878C+02	0.403504D+03	0.244768D+02	0.133658D+02	0.800518D-01
20	545945C+01	0.396113D+03	0.344711D+02	0.125976C+02	0.802199D-01
21	550325C+01	0.3964475D+03	0.535356D+02	0.117755D+02	0.801896D-01
22	3555C+01	0.331696D+03	0.570117D+02	0.109010D+02	0.804997D-01
23	174824C+01	0.344509D+03	0.723255D+02	0.943331C+01	0.805578D-01
24	77785C+02	0.340025D+03	0.906746D+02	0.768000D+01	0.811784D-01
25	22705C+02	0.424285C+03	0.137240D+02	0.139000D+02	0.795529D-01
26	15655C+02	0.430738D+03	0.109719D+02	0.172285D+02	0.795173D-01
27	15596C+02	0.432135D+03	0.109157D+01	0.160371C+02	0.793307D-01
28	15575C+02	0.429470D+03	0.139950D+02	0.149100D+02	0.792348D-01
29	15535C+01	0.421738D+03	0.258424D+02	0.135540D+02	0.78722D-01
30	367674C+01	0.411491D+03	0.856060D+02	0.120334C+02	0.795104D-01
31	77785C+02	0.315501D+03	0.135049D+03	0.104037D+02	0.795323D-01
32	258483C+02	0.443518D+03	0.192040D+02	0.940250D+01	0.801820D-01
33	23233C+02	0.442480D+03	0.205140D+02	0.176900D+02	0.795251D-01
34	23233C+02	0.400200D+03	0.205140D+02	0.176900D+02	0.795251D-01
35	155183C+02	0.407470D+03	0.116161C+01	0.166678D+02	0.785560D-01
36	155183C+02	0.477368D+03	0.305556D+02	0.161288D+02	0.785560D-01

FINITE ELEMENT RESULTS

ACODE	PSI(1)	VZ	VR	R(1)	DENSITY
121	0.177573E+02	0.476950D+03	0.613218D+01	0.155706D+02	0.781006D-01
122	0.156982E+02	0.478171D+03	0.184100D+02	0.149919D+02	0.783561D-01
123	0.136504E+02	0.470579D+03	0.304695D+02	0.143900D+02	0.791420D-01
124	0.116151E+02	0.473304D+03	0.439412D+02	0.137617D+02	0.781485D-01
125	0.960203E+01	0.466745D+03	0.568042D+02	0.131034D+02	0.782950D-01
126	0.761048E+01	0.465301D+03	0.721787D+02	0.124101D+02	0.782768D-01
127	0.613761E+01	0.455113D+03	0.877142D+02	0.116758D+02	0.784963D-01
128	0.471717E+01	0.454324D+03	0.108925E+03	0.108921E+02	0.783820D-01
129	0.330611E+01	0.455533D+03	0.125753D+03	0.100447D+02	0.787703D-01
130	0.183061E+01	0.450216D+03	0.154664D+03	0.912500D+01	0.786641D-01
131	0.277785E+02	0.455333D+03	-0.511848D+02	0.130305D+02	0.851432D-01
132	0.238348E+02	0.452176D+03	-0.268113D+02	0.170850D+02	0.847555D-01
133	0.197638E+02	0.452176D+03	-0.209776D+01	0.150450D+02	0.850586D-01
134	0.166195E+02	0.454773D+03	0.243723D+02	0.150183D+02	0.861083D-01
135	0.115716E+02	0.451673D+03	0.480709D+02	0.138622E+02	0.859906D-01
136	0.761269E+01	0.445420D+03	0.726211D+02	0.125162D+02	0.854598D-01
137	0.373938E+01	0.442517D+03	0.134154D+03	0.11241D+02	0.844684D-01
138	0.277795E+02	0.433517D+03	0.147570E+02	0.863300E+01	0.836565D-01
139	0.257195E+02	0.434645D+03	-0.331715D+02	0.178710D+02	0.800113D-01
140	0.237334E+02	0.452668D+03	-0.531875D+02	0.174329D+02	0.800170D-01
141	0.217203E+02	0.452668D+03	-0.332253D+02	0.169436D+02	0.801725D-01
142	0.196233E+02	0.455330D+03	-0.155548D+02	0.165215D+02	0.804006D-01
143	0.175490E+02	0.451705D+03	0.216717D+02	0.160470E+02	0.842765D-01
144	0.154518E+02	0.435470D+03	0.184922D+02	0.155578E+02	0.847018D-01
145	0.134518E+02	0.435516D+03	0.430655E+02	0.150539E+02	0.844563D-01
146	0.114805E+02	0.432220D+03	0.527213D+02	0.145235E+02	0.841465D-01
147	0.952998E+01	0.427784D+03	0.626725D+02	0.138210D+02	0.837028D-01
148	0.752998E+01	0.425152D+03	0.7316461D+02	0.134220D+02	0.822515D-01
149	0.565146E+01	0.430902D+03	0.832118D+02	0.122164D+02	0.815957D-01
150	0.378472E+01	0.430015D+03	0.773581D+02	0.108765D+02	0.801073D-01
151	0.188737E+01	0.430157D+03	0.116828D+03	0.108765D+02	0.801073D-01
152	0.277785E+02	0.403475D+03	-0.649058E+01	0.101410D+02	0.819782D-01
153	0.237334E+02	0.478478D+03	0.218113D+01	0.15865D+02	0.804262D-01
154	0.196018E+02	0.469219D+03	0.156332D+02	0.160710D+02	0.840034D-01
155	0.154173E+02	0.457100D+03	0.283310D+02	0.151140D+02	0.840084D-01
156	0.113298E+02	0.441905D+03	0.425226D+02	0.140925D+02	0.835273D-01
157	0.761269E+01	0.431513D+03	0.605547D+02	0.129701D+02	0.825513D-01
158	0.373938E+01	0.425520D+03	0.816221D+02	0.119752D+02	0.814030D-01

FINITE ELEMENT RESULTS

ACDE	PSI(I)	VZ	VR	R(I)	DENSITY
161	0.277789E+02	0.417646E+03	0.101248D+03	0.103930D+02	0.900633D-01
162	0.277789E+02	0.478634E+03	-0.675492D+01	0.178336E+02	0.515078D-01
163	0.277789E+02	0.485155D+03	-0.421942D+00	0.174211E+02	0.502381D-01
164	0.237425E+02	0.509131E+03	0.109074D+02	0.165600D+02	0.854759D-01
165	0.216637E+02	0.494303E+03	0.147020D+02	0.165600D+02	0.532575D-01
166	0.155759E+02	0.485736E+03	0.160210D+02	0.161123D+02	0.934066D-01
167	0.174917E+02	0.486268D+03	0.200143D+02	0.158118D+02	0.536727D-01
168	0.154091E+02	0.490478D+03	0.248823D+02	0.151772D+02	0.534122D-01
169	0.133786E+02	0.471376D+03	0.301735E+02	0.146874E+02	0.932072D-01
170	0.113878E+02	0.466029D+03	0.359072D+02	0.141806D+02	0.528849D-01
171	0.942447E+01	0.460245D+03	0.437066D+02	0.136551E+02	0.525452D-01
172	0.748007E+01	0.458066D+03	0.526157D+02	0.131085D+02	0.920361D-01
173	0.557745E+01	0.451983D+03	0.602933D+02	0.125381D+02	0.316443D-01
174	0.369556E+01	0.462827E+03	0.707446D+02	0.119404D+02	0.506951D-01
175	0.181501E+01	0.446247E+03	0.843481D+02	0.113113D+02	0.504555D-01
176	0.277789E+02	0.435433E+03	0.105571D+03	0.106450D+02	0.395333D-01
177	0.237504E+02	0.510663E+03	-0.337745D+01	0.178336E+02	0.870875D-01
178	0.156134E+02	0.521489E+03	0.430486D+01	0.176166D+02	0.848954D-01
179	0.155076E+02	0.521594E+03	0.998054D+01	0.161357E+02	0.827179D-01
180	0.114764E+02	0.504070D+03	0.279675D+02	0.152462D+02	0.849123D-01
181	0.753846E+01	0.495503E+03	0.181034D+02	0.142785D+02	0.887375D-01
182	0.266650E+01	0.506208D+03	0.492025D+02	0.1321145D+02	0.881460D-01
183	0.277789E+02	0.495435D+03	0.974175D+02	0.103725D+02	0.365557D-01
184	0.277789E+02	0.542637D+03	-0.554737D-12	0.178336E+02	0.926671D-01
185	0.258302E+02	0.550581D+03	-0.232366E+01	0.174413D+02	0.816462D-01
186	0.228733E+02	0.571703D+03	-0.516124E+01	0.170376D+02	0.803077D-01
187	0.214999E+02	0.559488D+03	-0.556618D+01	0.162400D+02	0.828558D-01
188	0.155451E+02	0.551970D+03	-0.751032D+01	0.161565D+02	0.859956D-01
189	0.157911E+02	0.552209D+03	-0.533673D+01	0.157693D+02	0.845056D-01
190	0.137800E+02	0.544012D+03	-0.460532D+01	0.153163D+02	0.843690D-01
191	0.117395E+02	0.541332D+03	0.299014D+01	0.148548E+02	0.846645D-01
192	0.980828E+01	0.526333E+03	0.377482D+01	0.143866D+02	0.845058D-01
193	0.784318E+01	0.523030D+03	0.902277E+01	0.138753D+02	0.847922D-01
194	0.784318E+01	0.533651D+03	0.150317E+02	0.133753D+02	0.841757D-01
195	0.392631E+01	0.550649D+03	0.242730E+02	0.122904D+02	0.835332D-01
196	0.195719E+01	0.547955D+03	0.521546D+02	0.117104D+02	0.830268D-01
197	0.0	0.556189D+03	0.892649D+02	0.111030D+02	0.818927D-01
198	0.0	0.492100D+03	0.0	0.178336E+02	0.545902D-01
199	0.277789E+02				
200					

FINITE ELEMENT RESULTS

NCDE	PSI(I)	VZ	VR	R(I)	DENSITY
201	236547E+02	0.520660D+03	0.528668D+01	0.170409D+02	0.935245D-01
202	0.152024E+02	0.481502D+03	0.105126D+02	0.162057D+02	0.972542D-01
203	0.111625E+02	0.463783D+03	0.144856D+02	0.153297D+02	0.977364D-01
204	0.125865E+01	0.450237D+03	0.164231D+02	0.133921D+02	0.977770D-01
205	0.346617E+01	0.423253D+03	0.163626D+02	0.123175D+02	0.977812D-01
206	0.27789E+02	0.438623D+03	0.132814D+02	0.111350D+02	0.962319D-01
207	0.256411E+02	0.450280D+03	0.137515D+01	0.118260D+02	0.940117D-01
208	0.235499E+02	0.462044D+03	0.0	0.174445D+02	0.942072D-01
209	0.213603E+02	0.482338D+03	0.0	0.170441D+02	0.932102D-01
210	0.190597E+02	0.510770D+03	0.0	0.166340D+02	0.929104D-01
211	0.170597E+02	0.540991D+03	0.0	0.162135D+02	0.926200D-01
212	0.149486E+02	0.489100D+03	0.0	0.157819D+02	0.91740D-01
213	0.128979E+02	0.473557D+03	0.0	0.153381D+02	0.913731D-01
214	0.109035E+02	0.453359D+03	0.0	0.148811D+02	0.904830D-01
215	0.85174E+01	0.430412D+03	0.0	0.144065D+02	0.896513D-01
216	0.651703E+01	0.443613E+03	0.0	0.139221E+02	0.87706D-01
217	0.518260E+01	0.426766D+03	0.0	0.134129D+02	0.875736D-01
218	0.39183E+01	0.426275D+03	0.0	0.128919D+02	0.829559D-01
219	0.162651E+01	0.392516D+03	0.0	0.123447D+02	0.691367D-01
220	0.0	0.362656D+03	0.0	0.117720D+02	0.998775D-01
221	0.0	0.0	0.0	0.111700D+02	0.954872D-01
222	0.0	0.0	0.0	0.0	0.0

2

I

WT

二

1

3

[illegible][illegible]

HT	0.	1	2	3	4	5	6	7	8	9	
0.	1332000	+03	0.	1332000	+03	0.	1332000	+03	0.	1332000	+03
1	1332000	+03	1	1332000	+03	1	1332000	+03	1	1332000	+03
2	1332000	+03	2	1332000	+03	2	1332000	+03	2	1332000	+03
3	1332000	+03	3	1332000	+03	3	1332000	+03	3	1332000	+03
4	1332000	+03	4	1332000	+03	4	1332000	+03	4	1332000	+03
5	1332000	+03	5	1332000	+03	5	1332000	+03	5	1332000	+03
6	1332000	+03	6	1332000	+03	6	1332000	+03	6	1332000	+03
7	1332000	+03	7	1332000	+03	7	1332000	+03	7	1332000	+03
8	1332000	+03	8	1332000	+03	8	1332000	+03	8	1332000	+03
9	1332000	+03	9	1332000	+03	9	1332000	+03	9	1332000	+03

[illegible][illegible]

235

238

239

ADPE	TEMP	ITCI	PRESS	PTOT	R40T
882	537070	555000	1628	175000	8840
883	540031	555000	1631	175000	8840
884	543461	555000	1654	175000	8840
885	547897	555000	1661	175000	8840
886	541887	555000	1607	175000	8840
887	541777	555000	1607	175000	8840
888	541924	555000	1607	175000	8840
889	542050	555000	1615	175000	8840
890	542201	555000	1622	175000	8840
891	541887	555000	1644	175000	8840
892	541121	555000	1609	175000	8840
893	541156	555000	1609	175000	8840
894	541250	555000	1602	175000	8840
895	541405	555000	1602	175000	8840
896	541364	555000	1604	175000	8840
897	541734	555000	1607	175000	8840
898	542730	555000	1616	175000	8840
899	542573	555000	1618	175000	8840
900	542730	555000	1615	175000	8840
901	544683	555000	1633	175000	8840
902	543997	555000	1599	175000	8840
903	543901	555000	1599	175000	8840
904	543541	555000	1535	175000	8840
905	543541	555000	1535	175000	8840
906	543541	555000	1588	175000	8840
907	540227	555000	1592	175000	8840
908	541716	555000	1607	175000	8840
909	541919	555000	1531	175000	8840
910	541641	555000	1562	175000	8840
911	543731	555000	1556	175000	8840
912	543617	555000	1549	175000	8840
913	543602	555000	1549	175000	8840

ADDE	TFMP	TOUT	PRE SS	PIN1	RHO1
161	57166CL+C3	0.608685D+03	0.190568D+02	237375D+02	0.105362D+00
162	58961UL+C3	0.619392D+03	0.199745D+02	237177D+02	0.103455D+00
163	58785UL+C3	0.621793D+03	0.196933D+02	236457D+02	0.102854D+00
164	58705UL+C3	0.622166D+03	0.194701D+02	236374D+02	0.103054D+00
165	584005UL+C3	0.615287D+03	0.200121D+02	243319D+02	0.105465D+00
166	583232UL+C3	0.614397D+03	0.201220D+02	243218D+02	0.106422D+00
167	58269UL+C3	0.614397D+03	0.201466D+02	243218D+02	0.105875D+00
168	582271UL+C3	0.613333D+03	0.201466D+02	243218D+02	0.106646D+00
169	58140UL+C3	0.613333D+03	0.199856D+02	241022D+02	0.106374D+00
170	580496UL+C3	0.612481D+03	0.198947D+02	241022D+02	0.106173D+00
171	579137UL+C3	0.612481D+03	0.197289D+02	241022D+02	0.105998D+00
172	577864UL+C3	0.611445D+03	0.196018D+02	239470D+02	0.105830D+00
173	577447UL+C3	0.611445D+03	0.193009D+02	238383D+02	0.105651D+00
174	575557UL+C3	0.609551D+03	0.192034D+02	238124D+02	0.105480D+00
175	571330UL+C3	0.608995D+03	0.190184D+02	237375D+02	0.105362D+00
176	571868UL+C3	0.619392D+03	0.186177D+02	236457D+02	0.103455D+00
177	57270UL+C3	0.622166D+03	0.181265D+02	236374D+02	0.102717D+00
178	571877UL+C3	0.615332D+03	0.183866D+02	241726D+02	0.106250D+00
179	571592UL+C3	0.614415D+03	0.183866D+02	241726D+02	0.105767D+00
180	570086UL+C3	0.612422D+03	0.186129D+02	241150D+02	0.105986D+00
181	566422UL+C3	0.611127D+03	0.182416D+02	237705D+02	0.105086D+00
182	566422UL+C3	0.608415D+03	0.178954D+02	235920D+02	0.104742D+00
183	567647UL+C3	0.619392D+03	0.173650D+02	235712D+02	0.102914D+00
184	564714UL+C3	0.621255D+03	0.171644D+02	234927D+02	0.102142D+00
185	563597UL+C3	0.622166D+03	0.172345D+02	233559D+02	0.102305D+00
186	561592UL+C3	0.618603D+03	0.173345D+02	233444D+02	0.104576D+00
187	561357UL+C3	0.615609D+03	0.174607D+02	240900D+02	0.105078D+00
188	560891UL+C3	0.614415D+03	0.175586D+02	241520D+02	0.106118D+00
189	561922UL+C3	0.614415D+03	0.175156D+02	241050D+02	0.105986D+00
190	561555UL+C3	0.613922D+03	0.176058D+02	240038D+02	0.105634D+00
191	562416UL+C3	0.613421D+03	0.175648D+02	239290D+02	0.105315D+00
192	560410UL+C3	0.612969D+03	0.174514D+02	238567D+02	0.105149D+00
193	560410UL+C3	0.612969D+03	0.174738D+02	237855D+02	0.104926D+00
194	561715UL+C3	0.611982D+03	0.175758D+02	237338D+02	0.104732D+00
195	557858UL+C3	0.611245D+03	0.171802D+02	236564D+02	0.104561D+00
196	557777UL+C3	0.610070D+03	0.171105D+02	236564D+02	0.104317D+00
197	552933UL+C3	0.609085D+03	0.167603D+02	234535D+02	0.103122D+00
198	552907UL+C3	0.619392D+03	0.169746D+02	235712D+02	0.102314D+00

AOCE
2012
2002
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020

TEMP
594118L+03
594982L+03
54971L+03
555339L+03
55455L+03
594042L+03
596071L+03
5581C7L+03
597734L+03
597010L+03
552048L+03
593948L+03
593978L+03
54306L+03
555337L+03
55868L+03
55424L+03
55575L+03
55781L+03
55554L+03
557698L+03

TTCI
621133D+03
615428D+03
614308D+03
613283D+03
612363D+03
61C593D+03
608685D+03
619292D+03
621251D+03
621539D+03
617805D+03
615353D+03
614704D+03
614234D+03
613634D+03
613225D+03
612759D+03
612319D+03
611752D+03
61C930D+03
609795D+03
60E685D+03

PRESS
207420D+02
214179D+02
215237D+02
215461D+02
215511D+02
216718D+02
2180C8D+02
208558D+02
205319D+02
210659D+02
212673D+02
211043C+02
215052D+02
215445D+02
216054D+02
2161C5D+02
217062L+02
218415D+02
220097D+02

PTOT
236139D+02
2401061D+02
240732D+02
239055D+02
237798L+02
236373D+02
234595D+02
235712C+02
236343D+02
240014D+02
241154D+02
241455C+02
239690D+02
238766L+02
238275D+02
237635D+02
23706D+02
236278D+02
235372C+02
234535D+02

RHOT
102512D+00
105825D+00
105872D+00
105313D+00
104874D+00
104503D+00
104122D+00
102814D+00
102146D+00
102150D+00
104959D+00
105878D+00
106125D+00
105827D+00
105515D+00
105281D+00
105055D+00
104852D+00
104650D+00
104450D+00
104281D+00
104122D+00

245

>DC YOU WISH TO ENTER THE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

>DC YOU WISH TO CONTINUE PLOT SEQUENCE?
1 = YES; 2 = NO.

INITIAL DISTRIBUTION LIST

	<u>No. Copies</u>
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1
4. Director, Turbopropulsion Laboratory, Code 67Sf Naval Postgraduate School Monterey, California 93940	20
5. LT J. Ferguson Naval Air Systems Command AIR-536 Washington, D.C. 20361	2
6. T. H. Okiishi Department of Mechanical Engineering Iowa State University Ames, Iowa 50011	1