UNCLASSIFIED

AD NUMBER: AD0853042
LIMITATION CHANGES
TO: Approved for public release; distribution is unlimited.
FROM: Distribution authorized to US Government Agencies only; Export Controlled; 1 May 1969. Other requests shall be referred to Air Force Weapons Laboratory, Kirtland AFB, NM 87117.
AUTHORITY
AFWL ltr dtd 30 Nov 1971

THIS PAGE IS UNCLASSIFIED

AFWL-TR-67-141, Vol V

USER'S MANUAL

Volume V

TURBULENT BOUNDARY LAYER PROGRAM

(REBOUND)

H. J. Fivel M. J. Schaffer

McDonnell Douglas Corporation
St Louis, Missouri 63166

Jimmie D. Young
Capt USAF

AD853042

TECHNICAL REPORT NO. AFWL-TR-67-141, Vol V

May 1969

DDC
RECEIVED
MAY 29 1969
B

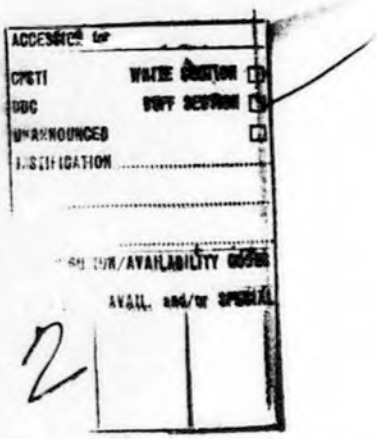
AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base
New Mexico

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of AFWL (WLDE), Kirtland AFB, NM, 87117.

460

BLANK PAGE

AFWL-TR-67-141, Vol V



AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base
New Mexico

When U. S. Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is made available for study with the understanding that proprietary interests in and relating thereto will not be impaired. In case of apparent conflict or any other questions between the Government's rights and those of others, notify the Judge Advocate, Air Force Systems Command, Andrews Air Force Base, Washington, D. C. 20331.

DO NOT RETURN THIS COPY. RETAIN OR DESTROY.

USER'S MANUAL

Volume V

Turbulent Boundary Layer Program

(REBOUND)

H. J. Fivel M. J. Schaffer

McDonnell Douglas Corporation
St Louis, Missouri 63166

Jimmie D. Young
Captain USAF

TECHNICAL REPORT NO. AFWL-TR-67-141, Vol V

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of AFWL (WLDE), Kirtland AFB, NM, 87117. Distribution is limited because of the technology discussed in the report.

FOREWORD

This report was prepared under Program Element 6.24.05.06.F, Project 5791, Task 27, and is a revision of an unpublished document prepared by McDonnell Douglas Corporation, St Louis, Missouri, under Contract AF 29(601)-7225.

Inclusive dates of research were June 1966 through October 1967. This report was submitted by the Air Force Weapons Laboratory Project Officer, Captain Jimmie D. Young (WLDE), on 27 March 1969.

The authors wish to acknowledge Mr. R. L. Cowperthwaite of McDonnell Douglas Astronautics Company and Sgt M. D. McEwan of the Air Force Weapons Laboratory for allowing the use of their material describing the subroutines treating the ablation mass loss, and Lt R. C. Buggeln for his material describing the chemical equilibrium routine.

This technical report is published to document the methods developed under this contract. The Air Force assumes no responsibility for the technical content of this report and makes no guarantees of the accuracy or adequacy of the program described herein.

Information in this report is embargoed under the US Export Control Act of 1949, administered by the Department of Commerce. This report may be released by departments or agencies of the US Government to departments or agencies of foreign governments with which the United States has defense treaty commitments, subject to approval of AFWL (WLDE), Kirtland AFB, NM, 87117.

Jimmie D Young

JIMMIE D. YOUNG
Captain, USAF
Project Officer

CARL F. DAVIS
Lt Colonel, USAF
Chief, Electronics Branch

for GEORGE C. DARBY, JR.
Colonel, USAF
Chief, Development Division

ABSTRACT

(Distribution Limitation Statement No. 2)

This report is one of six volumes providing user's manuals for a series of computer programs developed for the Air Force Weapons Laboratory by McDonnell Douglas Corporation. The program described in this volume treats the inviscid flow field, ablation mass loss, and turbulent boundary layer on a spherically blunted conical body. The principal advantage of this formulation is the use of a real gas equation of state for the ablation products/air mixture. This report describes the theory and calculation procedure for the program and presents sample cases for clarifying the data input. The program is written in FORTRAN VI computer language and has been run on both the CDC 6400 and CDC 6600 computers.

This page intentionally left blank.

CONTENTS

Section		Page
I	INTRODUCTION	1
II	THEORY	2
	Introduction	2
	Shock Shape	2
	Inviscid Flow Fields Profiles by the Streamtube Method	4
	Mass Loss Calculation	8
	Boundary Layer Equations	9
	Equilibrium Chemical Composition	11
	Numerical Solution	11
III	MAINLINE COMPUTER PROGRAM	18
	Introduction	18
	Program Structure and Operation	18
	Program REBOUND	24
	Segment ONE	26
	Segment TWO	67
	Segment THREE	74
	Segment FOUR	97
	Segment FIVE	103
	Segment SIX	115
	Segment SEVEN	163
	Segment EIGHT	171
	Segment NINE	177
IV	SUBROUTINES	196
	Subroutine GINPUT	196
	Subroutine VALUES	223
	Subroutine DISCOT	224

CONTENTS (cont'd)

Section		Page
	Subroutine ARDC69	244
	Subroutine HANSEN	257
	Subroutine DISCT	279
	Subroutine EXPX	282
	Subroutine DIRECT	285
	Subroutine CHEMS	288
	Subroutine DDUMP	295
	Subroutine SHOCKR	298
	Subroutine RGAS	308
	Subroutine GAMSTG	311
	Subroutine GAMCOF	313
	Subroutine INPUT	316
	Subroutine FCALC	332
	Subroutine ISNTRP	333
	Subroutine M367	337
	Subroutine ABMSLS	345
	Subroutine ATTACK	367
	Subroutine PCHEK	376
	Subroutine TCHEK	380
V	PROGRAM INPUT-OUTPUT	384
	Introduction	384
	Input	384
	Output	409

CONTENTS (cont'd)

Section		Page
	APPENDIX	430
	Mixing Program	430
	REFERENCES	447
	DISTRIBUTION	449

ILLUSTRATIONS

Figure		Page
1	Shock Angle Correlation for Sharp Cones with Real Gas Effects	5
2	Shock Wave and Body Geometry	6
3	Boundary Layer Calculation Procedure	12
4	General Flow of Computation	21
5	Illustration of Interpolation Scheme	227
6	Illustration of Two-Dimensional Table Interpolation	230
7	Ratio of Local Flow Speed to Free Stream Flow Speed	368
8	Ratio of Local Temperature to Free Stream Temperature	369
9	Ratio of Local Pressure to Free Stream-Pressure	370

TABLES

Table		Page
I	Curve Fit Constants for Shock Shape Calculations	4
II	Numbered Common in the Boundary Layer Program	19
III	Program Deck Structure	20
IV	Example of Complete Set of Data Cards	203
V	Example of Partial Set of Data Cards	205
VI	Altitude Layers for Atmosphere Model	246
VII	Constants for Atmosphere Model	246
VIII	Specific Layer Constants	247
IX	Data Structure	403

SECTION I

INTRODUCTION

The analysis and computer program described in this report permits the calculation of turbulent boundary layer properties on conical reentry vehicles entering the earth's atmosphere. An inviscid flow solution based on the streamtube concept is included in the program for providing edge conditions to the boundary layer solution. Ablation mass loss is normally considered; however, "clean air" calculations can be treated. The turbulent boundary layer solution generates profiles of temperature, density, chemical and ionic specie concentration, and electron-neutral collision frequency using an equilibrium thermochemistry calculation procedure as an equation of the state for the ablation products/air mixture. This equation of state which permits the use of real gas thermodynamic properties is the principal innovation and advantage of this formulation.

Some parts of this program were first formulated as separate programs and have been thoroughly documented in separate reports. A brief explanation of these incorporated programs is given in this document; however, the reader is directed to the appropriate reference for any additional detail he may wish to obtain

This report contains a discussion of the applicable theory, a description of the program, listings of each portion of the program, sample input and output listings, and a detailed discussion of the input form and variables. An appendix is included which describes a turbulent mixing model that was investigated as part of this study.

SECTION II

THEORY

1. INTRODUCTION

This section gives a description of some of the equations and the overall analysis used in generating the turbulent boundary layer profiles. Included in this section are discussions of the shock shape prediction, the mass loss calculations, the turbulent boundary layer solution, and the equilibrium thermochemistry routine which serves as an equation of state. Two parts of this analysis, the mass loss calculation and the equilibrium thermochemistry calculation, are mentioned only briefly. The theory for these two items can be found in Volumes I and II of this report (Refs. 1, 2).

2. SHOCK SHAPE

The prediction of the shock shape and the properties behind the shock provide the boundary conditions for the streamtube method of generating the inviscid flow field profiles. This shock shape is used with an oblique shock routine to determine the pressure and flow angle behind the shock. The shock shape subprogram calculates the geometry of the shock in terms of a shock angle at a particular shock radius.

To make the method valid for the deformed nose, it is necessary to use a correlation which can predict the effect of arbitrary nose blunting. The "blast wave" methods of Cheng (Ref. 3) and Mirels and Thornton (Ref. 4) are of this type. They treat the effect of blunting as an impulsive addition of energy, proportional to nose drag, to a slender cone flow field. These methods provide satisfactory correlation parameters for spherically blunted cone pressures and shock shapes.

Upon comparison with data from Blunt Body and Method of Characteristics computer programs, it was found that Mirels and Thornton's theory is in closer agreement near the nose than Cheng's theory, and Cheng's theory better predicts the turning to conical flow from the blast wave shock near the nose.

The parameters are

Mirels and Thornton:

$$\bar{y} = \frac{\sqrt{2}\, y\theta_c}{d_N \sqrt{C_{D_N}}} \qquad \bar{X} = \frac{KX\theta_c^2}{d_N \sqrt{C_{D_N}}} \qquad (1)$$

Cheng:

$$\bar{\bar{y}} = \frac{y\theta_c}{\sqrt{\epsilon}\, d_N \sqrt{C_{D_N}}} \qquad \bar{\bar{X}} = \frac{X\theta_c^2}{\sqrt{\epsilon}\, d_N \sqrt{C_{D_N}}} \qquad (2)$$

The parameters, y(shock radius), X(axial location), θ_c(cone angle), d_N(nose diameter), and C_{D_N} (nose drag coefficient), are used in the same way in both correlations. Real gas effects are included differently, however. In the Mirels-Thornton parameters, K is a function of the shock layer ratio of specific heats, γ_s, behind a normal shock, and the Cheng parameter, ϵ, is the density ratio across the normal shock.

The nose shape for other than a spherical cap is fitted with an ellipse to find the nose diameter, d_N, and the bluntness ratio, B_b. The Newtonian pressure drag integral for the nose is obtained either from the deformation program results (discussed in reference 5) or calculated for an undeformed spherical cap, and corrected to a drag coefficient, C_{D_N}, based on the nose frontal area. The equivalent effective ratio of specific heats, γ_E, at a given flight condition is used to compute the normal shock density ratio, ϵ, for Cheng's parameters. The shock layer ratio of specific heats, γ_s, is found by table look-up from a tabulation presented in subroutine GAMSTG (see Section IV) and is used to determine the K for the Mirels-Thornton parameter.

The shock shape correlations are represented in the program by curve-fitted equations. The Mirels-Thornton correlation is presented in the form

$$\bar{y} = A_1 \bar{X}^{B_1}$$

in the region from the nose to the beginning of conical flow, with the coefficients A_1 and B_1 fitted as a function of the bluntness ratio, B_b. Downstream of $\bar{\bar{x}} = 0.1$, Cheng's correlation is fitted as a parabola,

$$\bar{\bar{y}} = A_2\bar{\bar{X}}_2 + B_2\bar{\bar{X}} + C_2$$

This curve is joined to a linear shock found by using a least squares fit to data presented in reference 6. Values for the constants are given in table I, and the data for the linear shock is shown in figure 1. The output of this subprogram is a table of shock shape, shock radius, pressure behind the shock, and flow angle behind the shock for use in the streamtube program.

Table I

CURVE FIT CONSTANTS FOR SHOCK SHAPE CALCULATIONS

Cheng	Mirels-Thornton
$A_1 = 0.475 + 0.385/BB$	$A_2 = 0.24389$
$B_1 = 0.235 + 0.340/(BB+1)$	$B_2 = 0.659056$
BB = Bluntness Ratio	$C_2 = 0.20412$

3. INVISCID FLOW FIELD PROFILES BY THE STREAMTUBE METHOD

The streamtube program uses the conservation of mass and energy and a knowledge of the shock shape to predict the inviscid flow flied profiles. It is assumed that the pressure from the body to the shock and the flow angle from the body to the shock both vary linearly with mass flow rate.

From the shock geometry, figure 2, the location (x_b,y) of the intersection of a streamline which crosses the shock at (x_s,y_s) with the normal to the axis (x_b,y_b) is given by mass conservation as

$$\text{Total mass flow through the shock from the axis to } y_s = \text{Total mass flow from the body to } y \qquad (3)$$

Expressed in equation form

$$\dot{M}_s = \rho_\infty U_\infty \pi y_s^2 = 2\pi \int_{y_b}^{y} \rho U y \, dy \qquad (4)$$

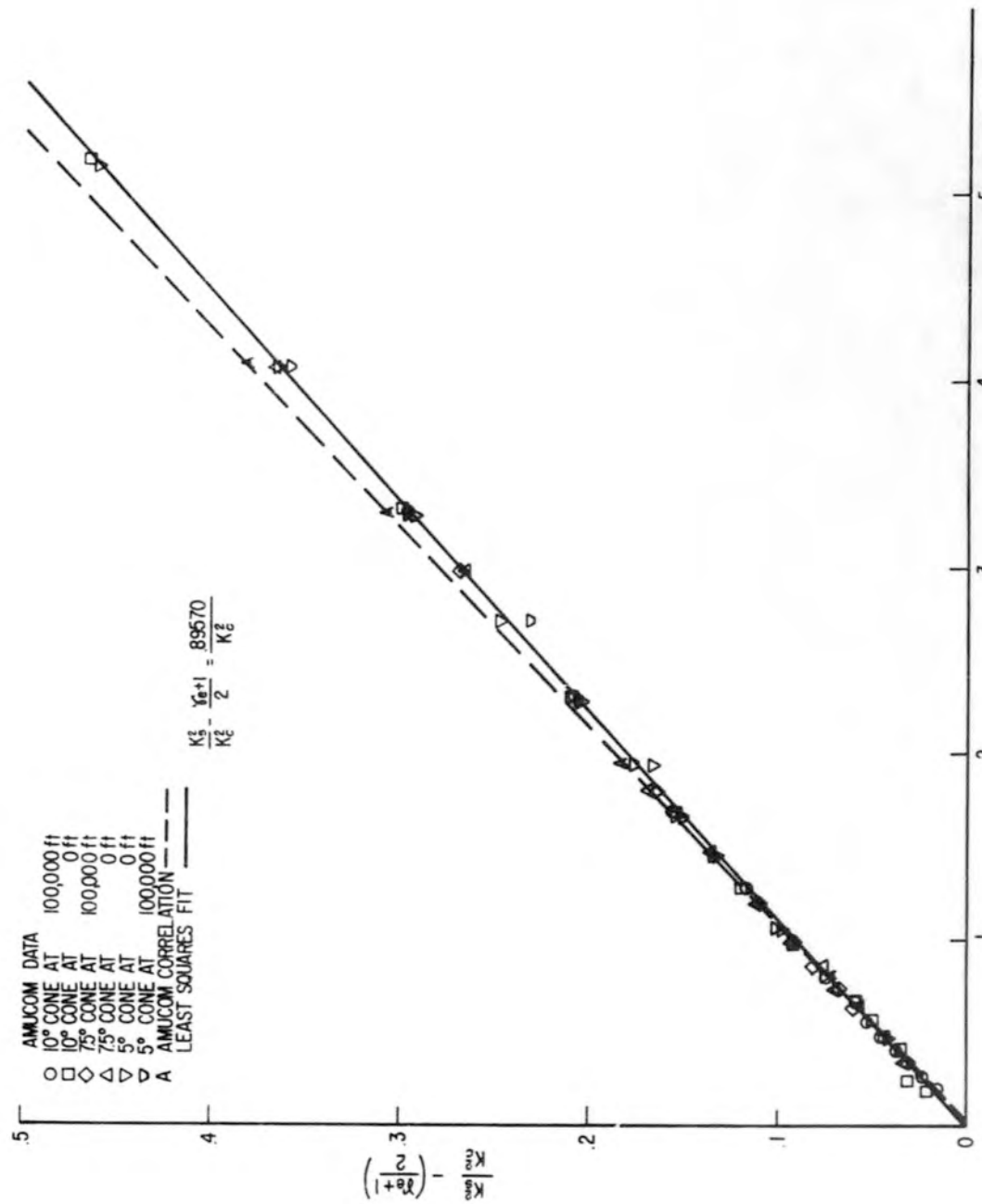


Figure 1. Shock Angle Correlation for Sharp Cones with Real Gas Effects

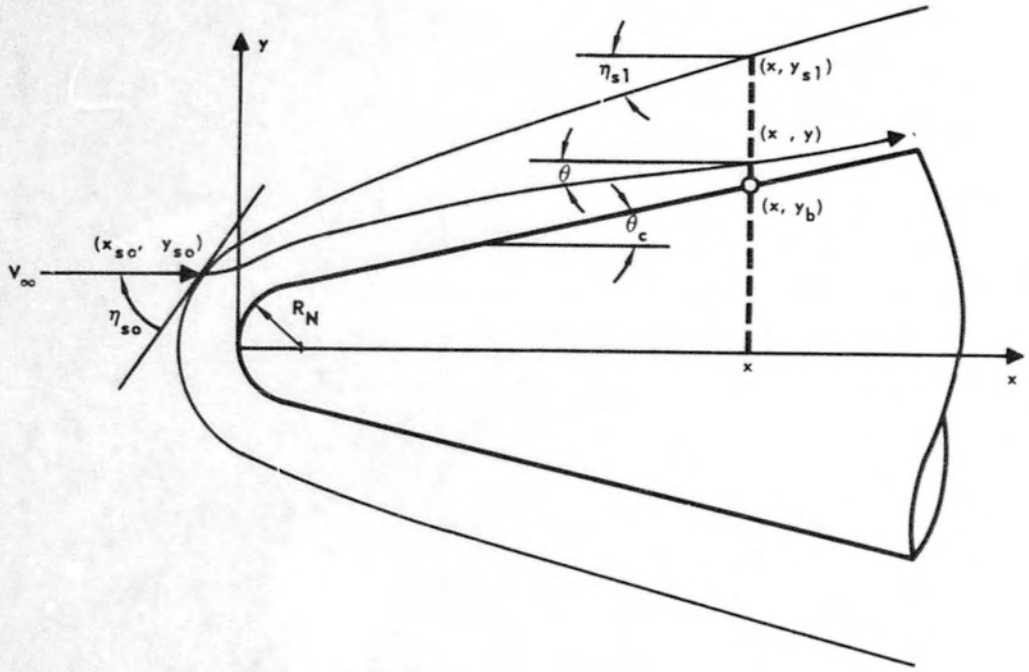


Figure 2. Shock Wave and Body Geometry

Differentiating the above equation

$$d\dot{M} = 2\pi \rho U y \, dy \tag{5}$$

and then integrate to obtain the y coordinate

$$y^2 - y_b^2 = \frac{1}{\pi} \int_0^{\dot{M}_s} \frac{d\dot{M}}{\rho U} \tag{6}$$

The local mass flow is found by first computing the entropy behind the bow shock at the point y_s, defined by

$$y_s^2 = \frac{\dot{M}_s}{\rho_\infty U_\infty \pi} \tag{7}$$

The entropy is for a normal shock with a normal velocity component, $U_N = U \sin \eta_s$. The pressure of each point in the inviscid flow associated with an entropy and mass flow rate is found from

$$P_{LOCAL} = P_b + \left(P_s - P_b\right) \frac{\dot{M}}{\dot{M}_s} \tag{8}$$

At the local pressure and entropy, the enthalpy and density are found. The enthalpy is then used to find the velocity at this streamline and flow angle, θ_{LOCAL}, as

$$V = \sqrt{2\left(h_{TOTAL} - h\right)} \tag{9}$$

and the component of this local velocity along the axis, given by

$$U = V \cos \theta_{LOCAL} \tag{10}$$

where

$$\theta_{LOCAL} = \theta_c + \left(\theta_s - \theta_c\right) \frac{\dot{M}}{\dot{M}_s} \tag{11}$$

With this local velocity and local density, the local mass flow is computed and used to determine the appropriate y-coordinate for the inviscid flow field profiles.

4. MASS LOSS CALCULATION

The mass loss resulting from an ablating body, or the material injection into the boundary layer, is discussed in detail in Volume II of this report (Ref. 1). The calculations will only be briefly discussed here.

Use is made of an aerodynamic heating routine to predict the heating rate, q, to the surface of a vehicle at the location of the desired boundary layer calculations. If the boundary layer solution is to be valid, the heating must be in the turbulent region. The injection rate, $\dot{M}$ (or surface mass loss rate), into the boundary layer is obtained by dividing the heating rate by an effective heat of ablation, Q*. Integrating the mass loss rate over the entire vehicle surface forward of the calculation point results in the injected gas mass flow that is compared with the boundary layer value from equation (26). In obtaining the integrated mass loss, consideration is given to whether laminar or turbulent heating applies, and the appropriate value is used accordingly.

Analytically, the variation of the heating rate along the surface for any flight condition is a function of length. Once the heating rate has been determined at some reference station, the heating rate at any other point in the proper regime can be calculated. Hence, for the turbulent region,

$$q = q_{REF} \left(\frac{X_{REF}}{X} \right)^{0.2} \tag{12}$$

and in the laminar region,

$$q = q_{REF} \left(\frac{X_{REF}}{X} \right)^{0.5} \tag{13}$$

An attempt to take transient effects into account is made by comparing the local heating rate with the radiation leaving the surface at the surface temperature. No integration is performed on that part of the surface where the local convective heating rate is less than the radiation heat loss. For a high performance reentry ballistic vehicle, the surface usually heats up rapidly during the early phase of a trajectory, and transient effects are small.

5. BOUNDARY LAYER EQUATIONS

The turbulent boundary layer calculations are based on the work of Spalding, as discussed in references 7 and 8.

The Crocco enthalpy profile is used as derived from the energy balance equation when assuming unity Prandtl and Lewis numbers. The Crocco expression yields the enthalpy-velocity relationship through the boundary layer, and may be written in the form

$$h = h_W + \left(h_{AW} - h_W\right)\left(\frac{U}{U_e}\right) - \left(h_{AW} - h_e\right)\left(\frac{U}{U_e}\right)^2 \qquad (14)$$

where

h is the enthalpy

h_W is the wall enthalpy

h_{AW} is the adiabatic wall enthalpy

U is the flow velocity

U_e is the boundary layer edge velocity

h_e is the boundary layer edge enthalpy

The adiabatic wall enthalpy, h_{AW}, is defined

$$h_{AW} = h_e + 0.9\left(\frac{U_e}{2gJ}\right) \qquad (15)$$

where

g is the gravitational constant

J is the mechanical equivalent of heat

The concentration profile of injected ablation material under similar assumptions is

$$f = f_W\left(1 - \frac{U}{U_e}\right) \qquad (16)$$

where

f is the mass fraction of ablation gas to total gas

f_W is this fraction at the wall

The velocity profile as a function of position is the other equation needed. The expression is obtained by integrating the Prandtl mixing length law, which is

$$\tau = \rho K^2 y^2 \left(\frac{dU}{dy}\right)^2 \tag{17}$$

where

τ is shear stress

ρ is density

y is distance normal to the surface

K is a mixing length constant

U is the velocity

Rearranging to give position as a function of velocity

$$y = \frac{\mu_W}{\left(\tau_W \rho_W\right)^{1/2}} \frac{1}{E} \exp\left[\frac{KU_e}{\left(\tau_W/\rho_W\right)^{1/2}} \int_0^{U/U_e} \frac{\rho/\rho_W{}^{1/2}}{\left(1 + BU/U_e\right)^{1/2}} d\left(\frac{U}{U_e}\right)\right] \tag{18}$$

where

E is taken as 12 and K as 0.4

and

μ_W is the viscosity at the wall

τ_W is the shear stress at the wall

ρ is the local density

ρ_W is the density at the wall

B is the injection parameter and is defined as

$$B = \frac{\dot{M}_{INJ} U_e}{\tau_W} \tag{19}$$

B is also related to the mass fraction of the injected gas at the wall through

$$f_W = \frac{B}{1 + B} \tag{20}$$

6. EQUILIBRIUM CHEMICAL COMPOSITION

Volume I of this series (Ref. 2) presents a detailed discussion of the equilibrium thermochemical program and calculations. The complete program is used at the end of the boundary layer calculations to generate the temperature, electron concentration, and collision frequency profiles, as well as the high temperature equilibrium chemical composition, for the boundary layer enthalpy profile. A smaller version of the program, capable of handling only 25 species as opposed to 70 species in the complete version, is used in the main boundary layer calculations to provide the real gas equation of state. The basic computations of the two versions are exactly the same.

The original computer program, to calculate the equilibrium composition of an arbitrary high temperature gas mixture, including the contributions of ionized species, is based on the computer program developed by JPL (Ref. 9). The program has been modified to include a greater number of species and elements, to enable interpolation in the thermal property tables, and to iterate on the primary independent state variables of temperature and density when either the pressure and temperature, pressure and density, or pressure and enthalpy are specified.

The chemical equilibrium composition of a mixture is determined by applying the principle of conservation of mass of the elemental species and applying the second law of thermodynamics. For a constant mass system in thermal equilibrium at an arbitrary temperature and pressure, the chemical composition of that system is such that the Gibbs free energy is a minimum.

7. NUMERICAL SOLUTION

a. General

The preceding discussions have shown the methods of determining the boundary layer and inviscid flow field properties. This section discusses the way in which the above-mentioned independent calculations, along with an independent determination of mass loss and mass loss rate, are used in two iteration loops to arrive at a total flow field solution. The boundary layer

equations are solved numerically by iterating on boundary layer edge conditions and wall mass fraction until the ablation or injected gas and the total gas flows are conserved. Figure 3 presents a diagram of the calculation procedure.

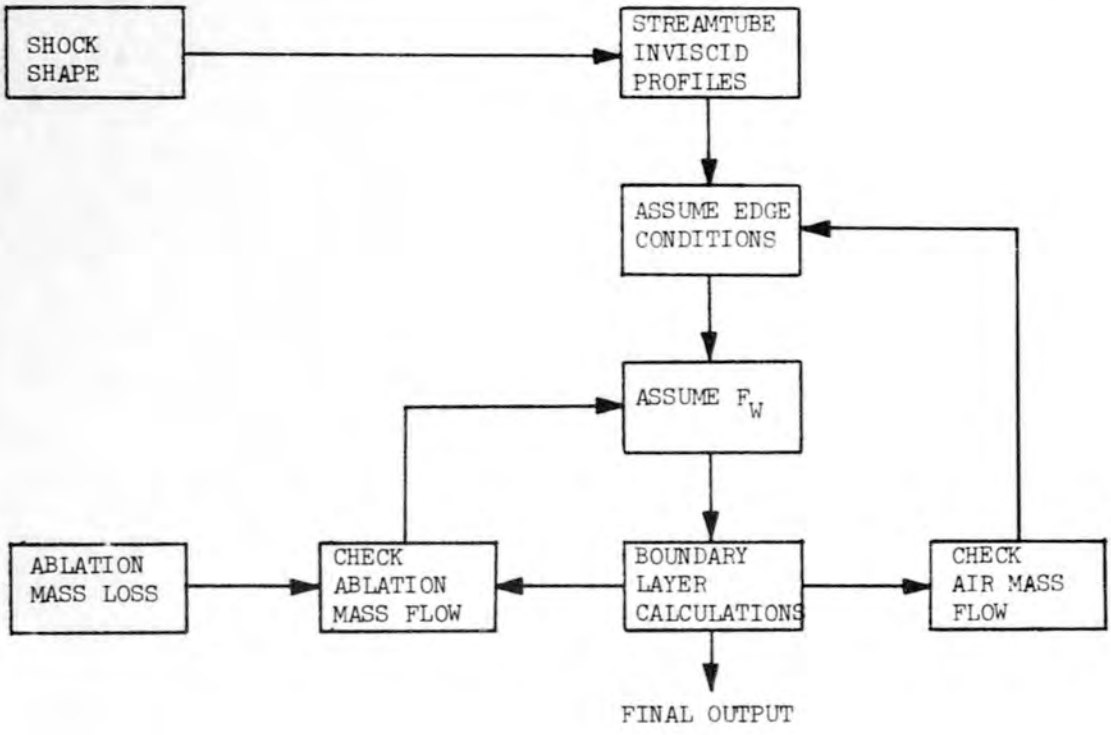


Figure 3. Boundary Layer Calculation Procedure

The edge conditions are chosen from inviscid flow properties calculated by the streamtube method. Starting with the inviscid field profile point next to the body as the first set of boundary layer edge conditions, the boundary layer calculations are performed and the air flow determined. The boundary layer air flow is then compared to that of the inviscid field, which for the first point happend to be zero. The process is then repeated for the next point on the inviscid profile, and so on until the mass flows match or the generated mass flow curves cross. The boundary layer calculations up to this point are carried out with the ideal gas relationships. When the matching or crossover point is found, the boundary layer for that point is recalculated with the real gas equation of state. The air flows are compared again and a new matching or

crossover point is determined for real gas boundary layer properties. One final iteration is necessary: interpolation for the edge conditions at the crossover point to provide air mass flow agreements between the boundary layer and the inviscid field.

The temperature, electron concentration, and collision frequency for the resultant boundary layer profiles are then computed with the aid of the equilibrium thermochemistry program. This particular version of the thermochemistry program, capable of handling up to 70 chemical species, including ionized species, is referred to as the "large" version, as opposed to the smaller version used to provide the real gas capability in the boundary layer calculations themselves.

b. Specific Calculations of Boundary Layer and Inviscid Field Matching

For a particular set of edge conditions, a value of the mass fraction at the wall, f_W, is assumed. B is determined from equation (20), and τ_W, in turn, from equation (19). For a clean air boundary layer, B is zero, and τ_W, the clean air shear stress at the wall, is obtained from a routine in the weight loss subroutine.

The density, ρ_W, the molecular weight of the gas mixture at the wall, M_{MIX}, and the enthalpy at the wall, h_W, are determined for the particular mass fraction of the mixture, the wall temperature, and boundary layer pressure through use of a small modified version of the chemical equilibrium program.

Assuming the molecular weight of air, M_{AIR}, to be 28.962, the molecular weight of the injected gas at the wall, M_{INJ}, is calculated from the expression

$$\frac{1}{M_{MIX}} = \frac{f_W}{M_{INJ}} + \frac{1 - f_W}{M_{AIR}} \tag{21}$$

A specific heat for air is calculated from

$$C_{AIR} = \frac{h_e}{T_e} \tag{22}$$

where

h_e is the edge enthalpy

T_e is the edge temperature

The specific heat of the injected material, C_{INJ}, is calculated from

$$h_W = T_W \left[f_W C_{INJ} + \left(1 - f_W\right) C_{AIR} \right] \tag{23}$$

where T_W is the temperature at the wall.

An ideal gas equation of state based on these specific heat values for the injected gas-air mixtures is

$$\frac{\rho}{\rho_W} = \frac{T_W M_{AIR} C_{AIR} \left[1 + f \left(\frac{C_{INJ}}{C_{AIR}} - 1 \right) \right]}{M_{MIX}\, h \left[1 + f \left(\frac{M_{AIR}}{M_{INJ}} - 1 \right) \right]} \tag{24}$$

where ρ/ρ_W is the ratio of local density to the density at the wall. This equation of state is used in the initial phases of the iteration process. For the final answer a detailed thermochemistry calculation is used to obtain a more accurate real gas equation of state.

Next, the adiabatic wall enthalpy is determined from equation (15), and the boundary layer enthalpy profile generated, using equation (14), for values of the velocity ratio from 0 to 1.0, in even increments of 0.01. When the ideal gas solution is applied, the density profile given by equation (24) is calculated for the corresponding values of enthalpy, after the specific heats and molecular weights are obtained. The density profile for the real gas solution is obtained from the equilibrium thermochemistry calculations, with the enthalpies as independent variables.

The particular density profile is used in equation (18) to generate the velocity profile. The integral in equation (18) is numerically evaluated by the trapezoidal rule. The choice of the step size is discussed in Section II.7.c.

The total amount of injected material appearing in the boundary layer at the desired location is obtained by integrating the point mass flow across the boundary layer.

$$W_{INJ} = 2\pi R \int_0^{\delta} f\rho U \, dy \tag{25}$$

where

R is the local vehicle radius

y is the distance from the vehicle wall

δ is the y value at the boundary layer edge

In terms of the above variables

$$W_{INJ} = 2\pi R \left(f_W \rho_W U_e\right) \int_0^{\delta} \left(\frac{U}{U_e}\right)\left(\frac{\rho}{\rho_W}\right)\left(1 - \frac{U}{U_e}\right) dy \qquad (26)$$

The total gas flow in the boundary layer is obtained by a similar integration:

$$W_{TOTAL} = 2\pi R \left(\rho_W U_e\right) \int_0^{\delta} \left(\frac{U}{U_e}\right)\left(\frac{\rho}{\rho_W}\right) dy \qquad (27)$$

The displacement thickness, δ^*, is the distance the inviscid flow is moved away from the vehicle surface because of the boundary layer flow. It is calculated from the following expression:

$$W_{TOTAL} = 2\pi R \, \rho_e U_e \, (\delta - \delta^*) \qquad (28)$$

The integrals, equations (26) and (27), to determine the boundary layer mass flows are also numerically evaluated by the trapezoidal rule.

If the injected gas mass flow as calculated from the boundary layer program in equation (26) does not correspond to the ablation weight loss rate (Section II.a), the injected gas mass fraction is changed and the calculations repeated until the computed mass loss rate and the integrated ablation gas flow in the boundary layer agree. This procedure is repeated for each chosen boundary layer edge condition.

c. Choice of Integration Step Size

Except for the integration across the boundary layer to obtain mass flow, the independent variable throughout the boundary layer analysis is the ratio of velocity at a point in the boundary layer to the edge velocity,

varying from zero at the wall to 1 at the edge (see equations (14), (16), and (18)). When the calculations on the computer were initially checked out, the velocity ratio was specified in increments of 0.01, making 101 points altogether. It soon becomes obvious that the same number of points cannot be used in making the real gas computations with the chemistry routine, since the total machine running time per case would become prohibitive. Therefore, the number of points generated for the chemistry calculations was reduced.

It was empirically determined that with the choice of 16 points for the velocity ratio in increments of

0.10 between 0.0 and 0.8,

0.05 between 0.8 and 0.9, and

0.02 between 0.9 and 1.0,

an integration error of no larger than 5 percent resulted from taking the fewer points.

d. Calculation Procedure Summary

The major steps in defining the turbulent boundary profiles are listed below. For a given flight condition (i.e., velocity and altitude) and a particular vehicle geometry, the program performs the following steps:

(a) Determines the shock shape surrounding the vehicle, the body pressure, and the conditions behind the shock at the desired body station.

(b) Uses the shock shape parameters to generate inviscid field flow profiles (velocity, density, enthalpy, and mass flow).

(c) With the aid of a mass loss program, determines the rate of material injection at the body station, and the total amount of injected material.

(d) Generates the turbulent boundary layer enthalpy-velocity relationship, using conditions at the wall, and in the inviscid flow field for the boundary layer velocity profile.

(e) Using ideal gas relationships between enthalpy and density, calculates the boundary layer velocity profile.

(f) Integrates the profiles to obtain total gas flow, air flow, and injected material flow.

(g) Adjusts the mass fraction of injected material at the wall and repeats steps (d), (e), and (f) until the boundary layer value of injected material corresponds to the value determined from mass loss program (c).

(h) Picks new inviscid flow boundary layer edge conditions and repeats steps (d), (e), (f), and (g) until the boundary layer air flow agrees with inviscid field air flow up to the boundary layer edge.

(i) Regenerates the boundary layer profiles (e) with a real gas solution. The real gas density-enthalpy relationships are obtained with the aid of a small version of a high-temperature equilibrium thermochemistry program.

(j) Repeats steps (d) through (h) with real gas calculations.

(k) Generates the temperature, electron concentration, and collision frequency profiles from the resultant profile, using a large version of a high-temperature equilibrium thermochemistry program.

SECTION III

MAINLINE COMPUTER PROGRAM

1. INTRODUCTION

This section contains a description of the main portion of the turbulent boundary layer computer program. The general program structure and operation is discussed in the next paragraph. The succeeding paragraphs contain a detailed discussion of each of the main segments in the program. However, the subroutines which are required and which perform many of the vital computations in this program are discussed individually in Section IV of this report.

2. PROGRAM STRUCTURE AND OPERATION

The computer program consists of a short main program, used to initiate the computation, followed by nine segments. The computations within each segment are performed through a "chaining" process, controlled mainly in Segment ONE by certain logic statements. Communication between segments is maintained through various numbered COMMON packages. The functions of these COMMON blocks are described in table II. The actual deck structure (including the subroutines for each segment) is illustrated in table III.

Figure 4 illustrates the general flow of computation in the program. Program GO initiates the computation by setting the variable K1 = 1 and calling Segment ONE. K1 is the variable used in Segment ONE to control which other segment will be called. After Segment ONE is entered, the first call is to Segment TWO (K1 = 1). Segment TWO loads the symbolic case data for the computation to be performed, increments K1 to a value of 2, and returns control to Segment ONE. If multiple chemical composition cases are to be run, the call is to Segment EIGHT, otherwise Segment THREE is called. This segment calculates the shock shape and properties behind an oblique shock for a hemispherically blunted cone. Control is returned to Segment ONE, which calls either Segment FOUR or FIVE. If the ablation mass loss is to be computed, Segment FOUR is called. If the local and integrated mass loss is input as data, Segment FOUR is bypassed and Segment FIVE is called. This segment performs the streamtube calculations in the inviscid flow field which serves as edge conditions for

Table II

NUMBERED COMMON IN THE BOUNDARY LAYER PROGRAM

Block no.	Function
/234/	Common used by subroutines CHEMS and INPUT to communicate data between the two subroutines in SEGMENT ONE.
/367/	Common used by the mass loss program in SEGMENT FOUR for communication between the subroutines in SEGMENT FOUR.
/456/	Common used by the data input subroutines DIRECT, DDUMP, and GINPUT in SEGMENTS TWO and FOUR.
/666/	Common used by subroutine CHEMS in SEGMENT ONE to store the chemistry library data.
/716/	Common used by the boundary layer program in SEGMENT ONE to communicate between the main program and the subroutines in SEGMENT ONE.
/567/	Common used by SEGMENT FIVE.
/1/	Common used by SEGMENT THREE to communicate between the main program and subroutine SHOCKR.
/222/	Common used to transmit variables calculated in SEGMENT ONE to SEGMENT SIX.

Table III

PROGRAM DECK STRUCTURE

Segment	Required subroutines	Block data
GO	None	None
ONE	HANSEN	BLK005
	INPUT	BLK002
	FCALC	
	VALUES	
	CHEMS	
	DISCT	
	EXPX	
	DISCOT	
TWO	DIRECT	
	DDUMP	
	GINPUT	
	ARDC59	
THREE	HANSEN	BLK002
	SHOCKR	COEFF
	RGAS	
	GAMSTG	
	GAMCOF	
	DISCOT	
FOUR	M367	PRMTAB
	ABMSLS	BLK002
	ATTACK	BLK001
	DIRECT	BLK003
	DDUMP	BLK004
	GINPUT	BLK005
	DISCOT	
	HANSEN	
	ARDC59	
	PCHEK	
	TCHEK	
FIVE	HANSEN	BLK002
	DISCOT	ENTRPY
	ISNTRP	SHOCK
SIX	DISCT	
	EXPX	
SEVEN		
EIGHT	INPUT	
	FCALC	
NINE	EXPX	PERM
	DISCT	

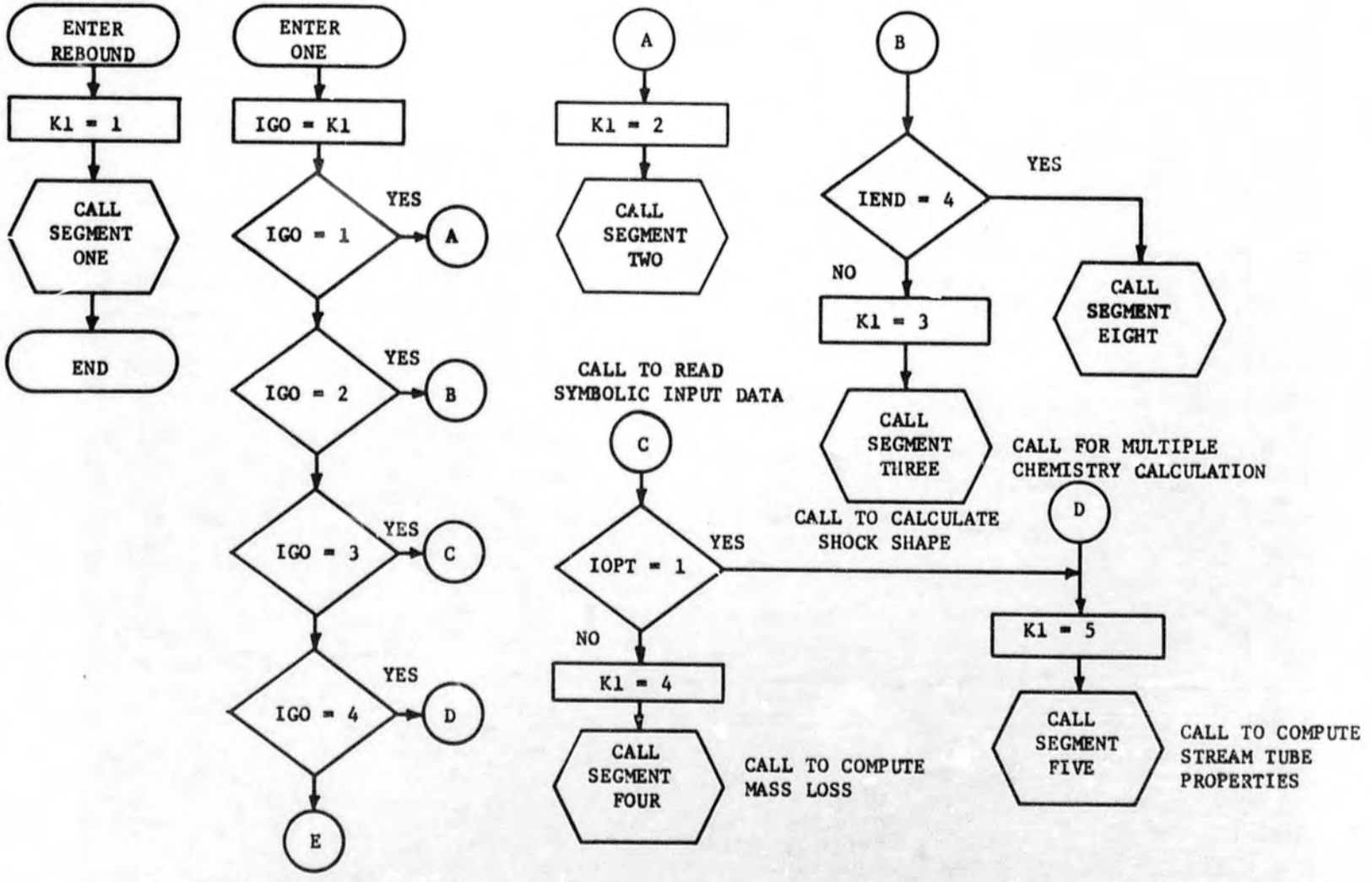


Figure 4. General Flow of Computation

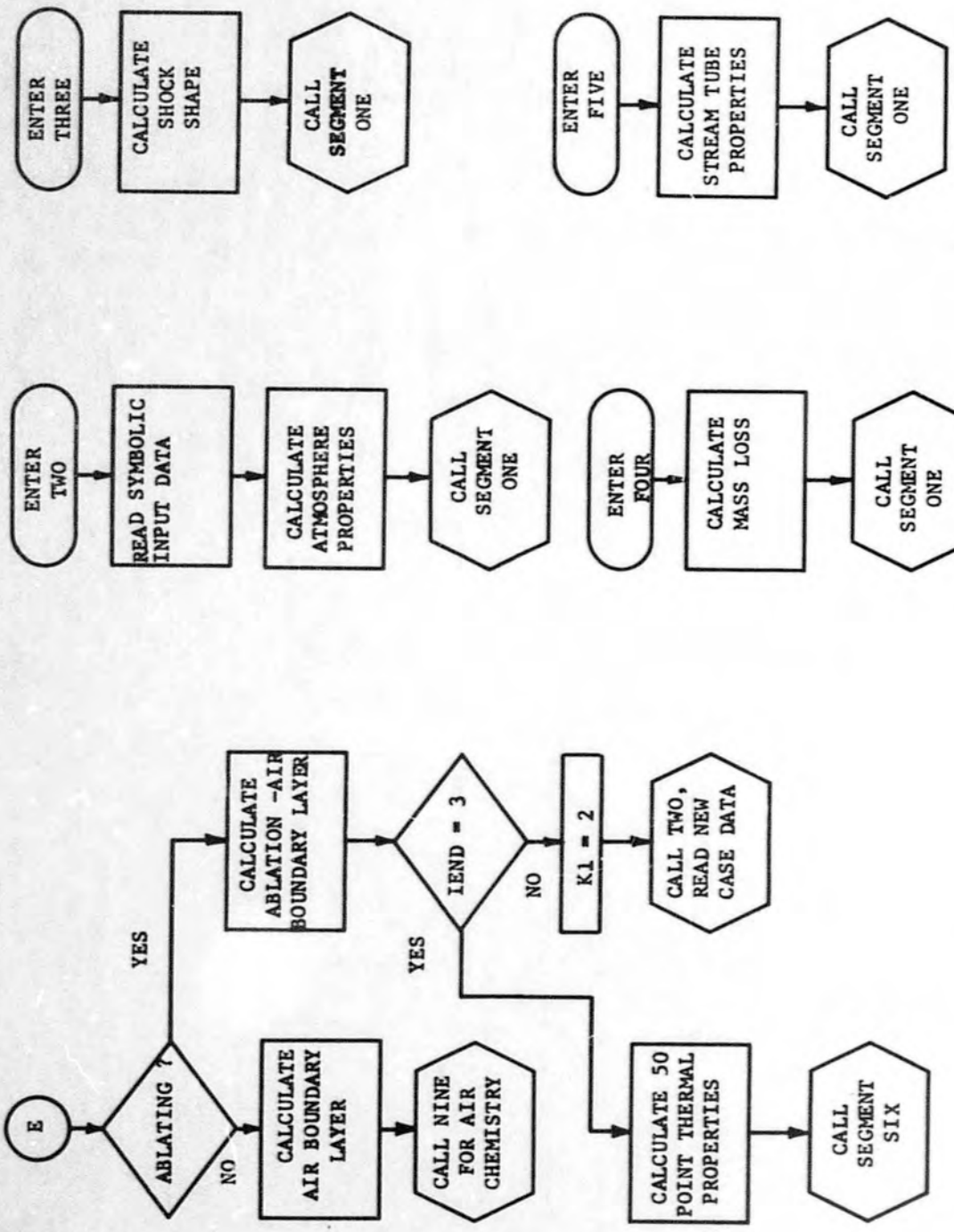


Figure 4 (cont'd). General Flow of Computation

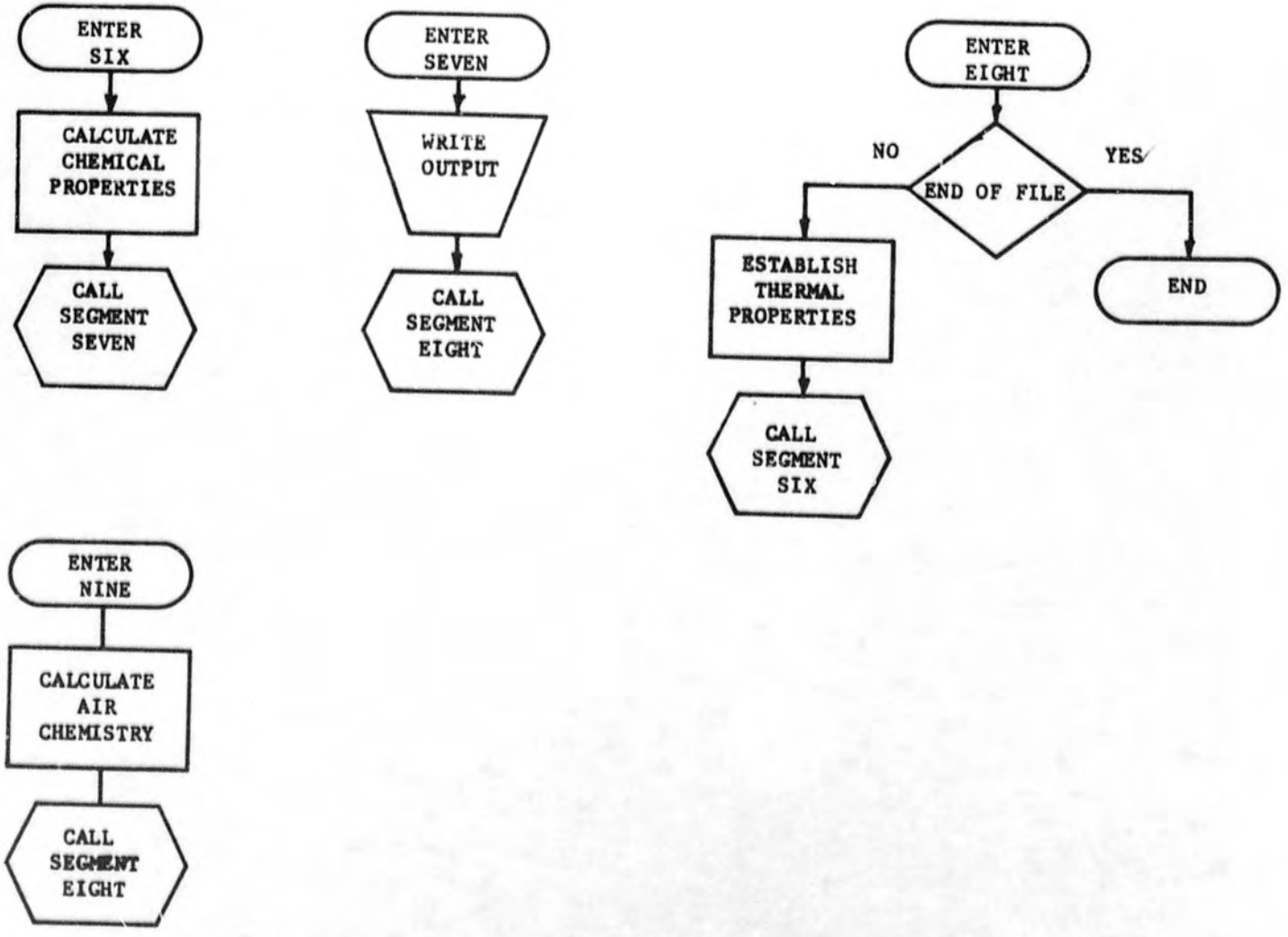


Figure 4 (cont'd). General Flow of Computation

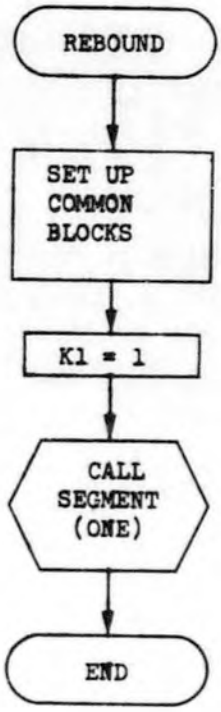
the boundary layer equations. Again, control is returned to Segment ONE. Segment ONE now determines if ablation is occurring. If ablation is occurring, Segment ONE calculates the boundary layer thermal properties at 10 points through the boundary layer. If multiple boundary layer cases are being considered (IEND = 1 or 2), new data is read in by calling Segment TWO. If a complete boundary layer solution is desired (IEND = 3), the thermal properties at 50 points are determined, and Segment SIX is called. Segment SIX then calculates the chemical species present in the boundary layer at each profile point. Segment SEVEN provides the profile output, then calls Segment EIGHT which determines if the program is finished. If ablation had not been occurring, Segment ONE would have calculated boundary layer thermal properties based on clean air. Segment NINE would then have been used to calculate the clean air chemical species. The output would follow the same order as described above.

As mentioned earlier, Segment EIGHT can be entered directly from Segment ONE. For this option (IEND = 4) Segment EIGHT sets up the boundary layer thermal profiles using certain input data, and then calls Segment SIX to perform the chemistry calculations. Multiple chemistry cases can be run using this option.

3. PROGRAM REBOUND

a. Function

This program is a dummy main program which simply calls Segment ONE. Such a dummy main program is necessary because the main program of a segmented job cannot be recalled by another segment. PROGRAM GO contains common packages which are the same length as the common packages in the segments.



b. Listing

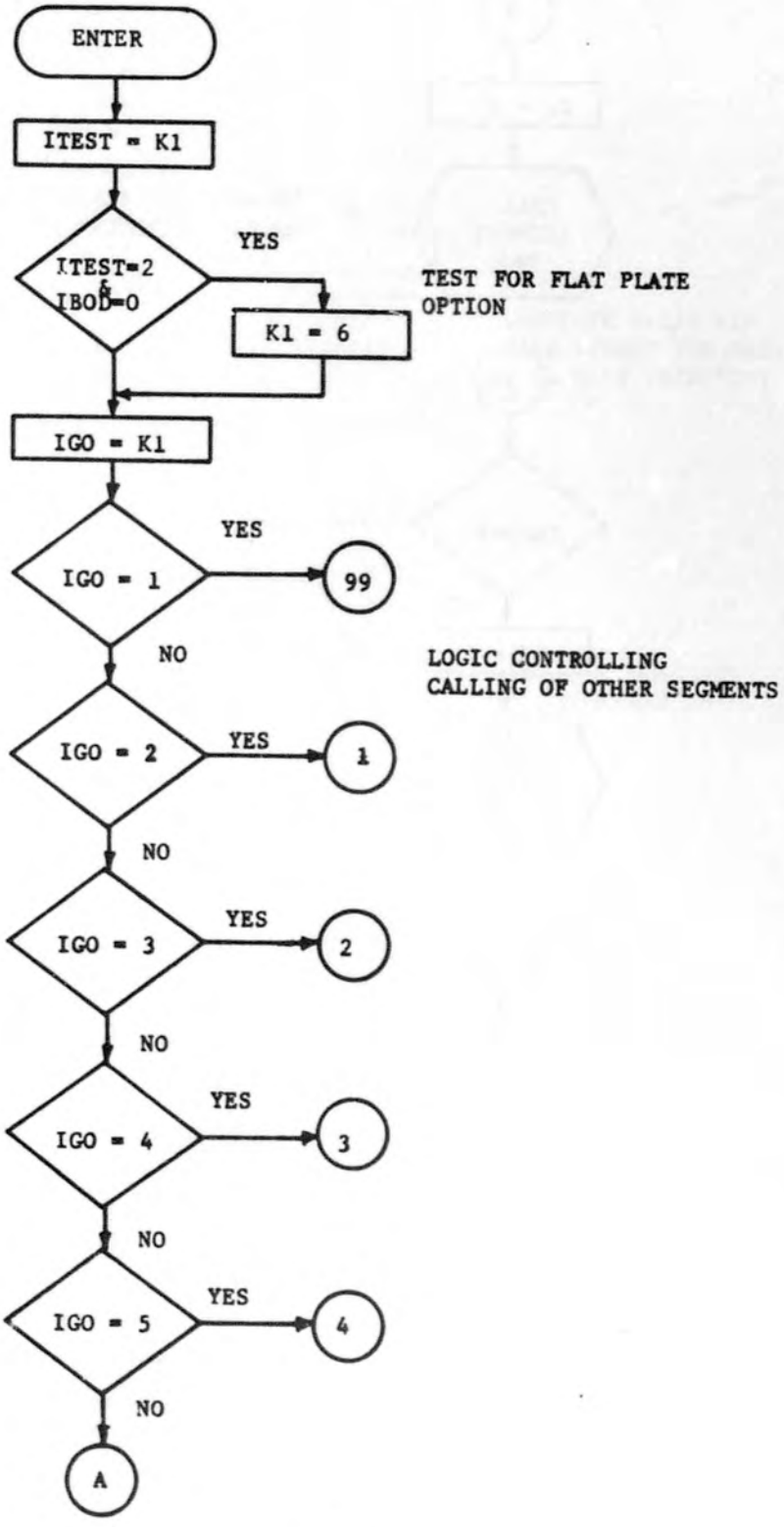
```
      PROGRAM REBOUND(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTOUT,
     *   TAPE22)
      COMMON /234/ SSSS(3960)
      COMMON /333/ JFS(150)
      COMMON /367/ AM367(150)
      COMMON / 456/ INTAP,IOTAP,INT,NMN,M,NGR
      COMMON /666/ SSS(292)
      COMMON /716/    TITLE,   IOPT,   IBOD,   IEND, NABL1, NABL2, IREAD,
     1   KODE,IHFMIS, KODEW,
     2   ANTL,     RN,    ALT,    VEL, XINCH,     Y1,   SMXB,   YMXB, DYDX1,
     3   CP10,     TW,     EM, EMDOT,     WA, WABL1, WABL2,    YBB,
     4     K1,      N,     P1,   RHO1,    ZMD,     ZS,      P,     YB,     VM,
     5    UEV,    HEH, RHOER, EMDTT,     TO,   RHOW,     HW,XMASSO, EMDTO,
     6   VRTO,    I18,    I19,      R,    TAU,     CI ,NTOTAL,NWHICH,   IJJJ,
     7      L
      COMMON /567/ PS,THS,PSTRM(40)
      COMMON /1/ SOS,TATMOS,RHO2
      COMMON /222/ Y,FS,RHOOW,TRS,CFS,ELS,HZ,HAW,HE,RHW,TITL(5),
     * C5,C6,UE,B,CIR
      DIMENSION TITLE (5), AA(1)
      DIMENSION TRS(101),CFS(101),ELS(101)
      DIMENSION HH(101)
      DIMENSION ZMD(40),ZS(40)
      DIMENSION RHOOW(101), Y(101)
      DIMENSION UEV(40), HEH(40), RHOER(40), EMDTT(40)
      K1 = 1
      CALL CHAIN (ONE)
      END
```

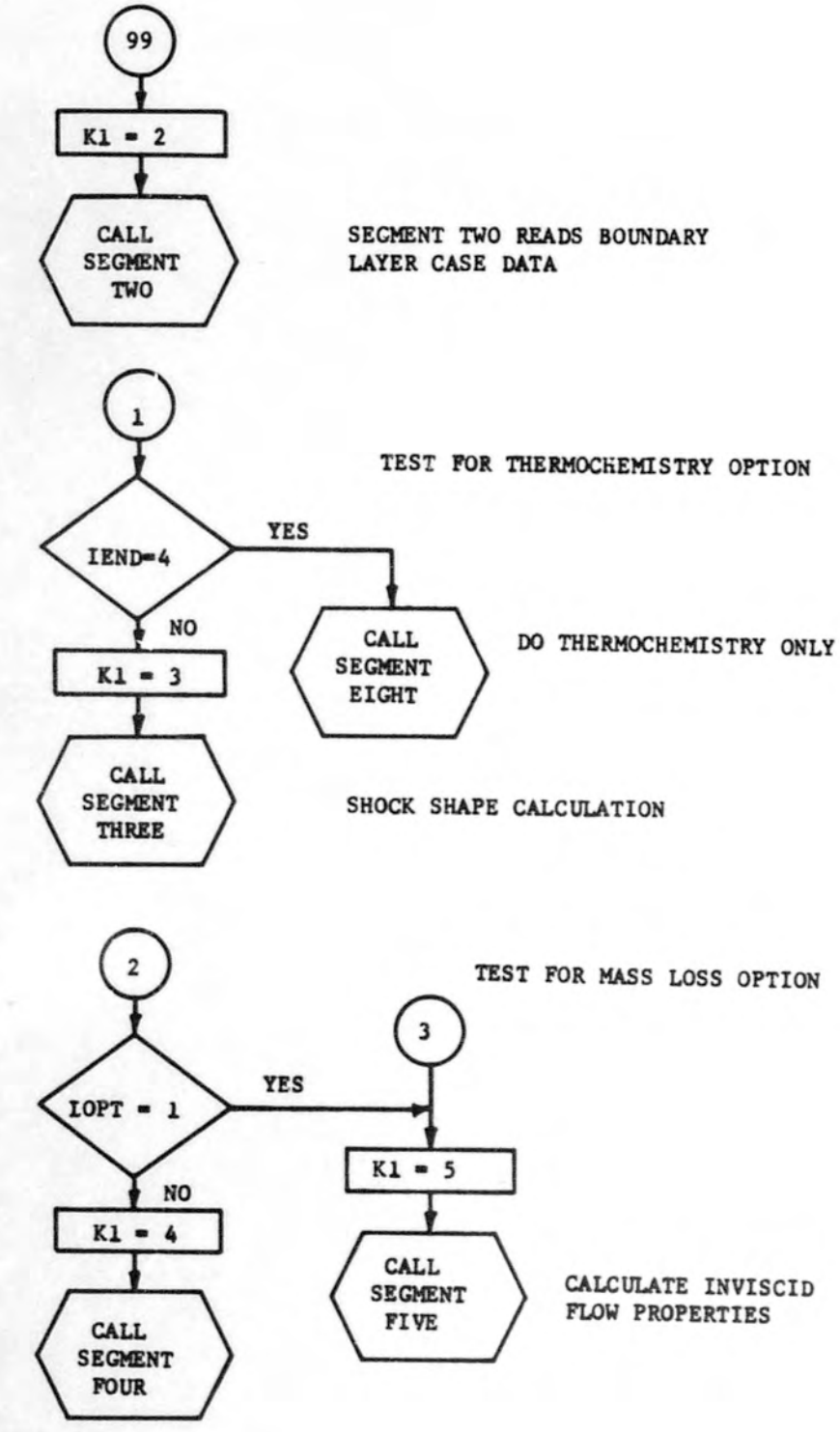
4. SEGMENT ONE

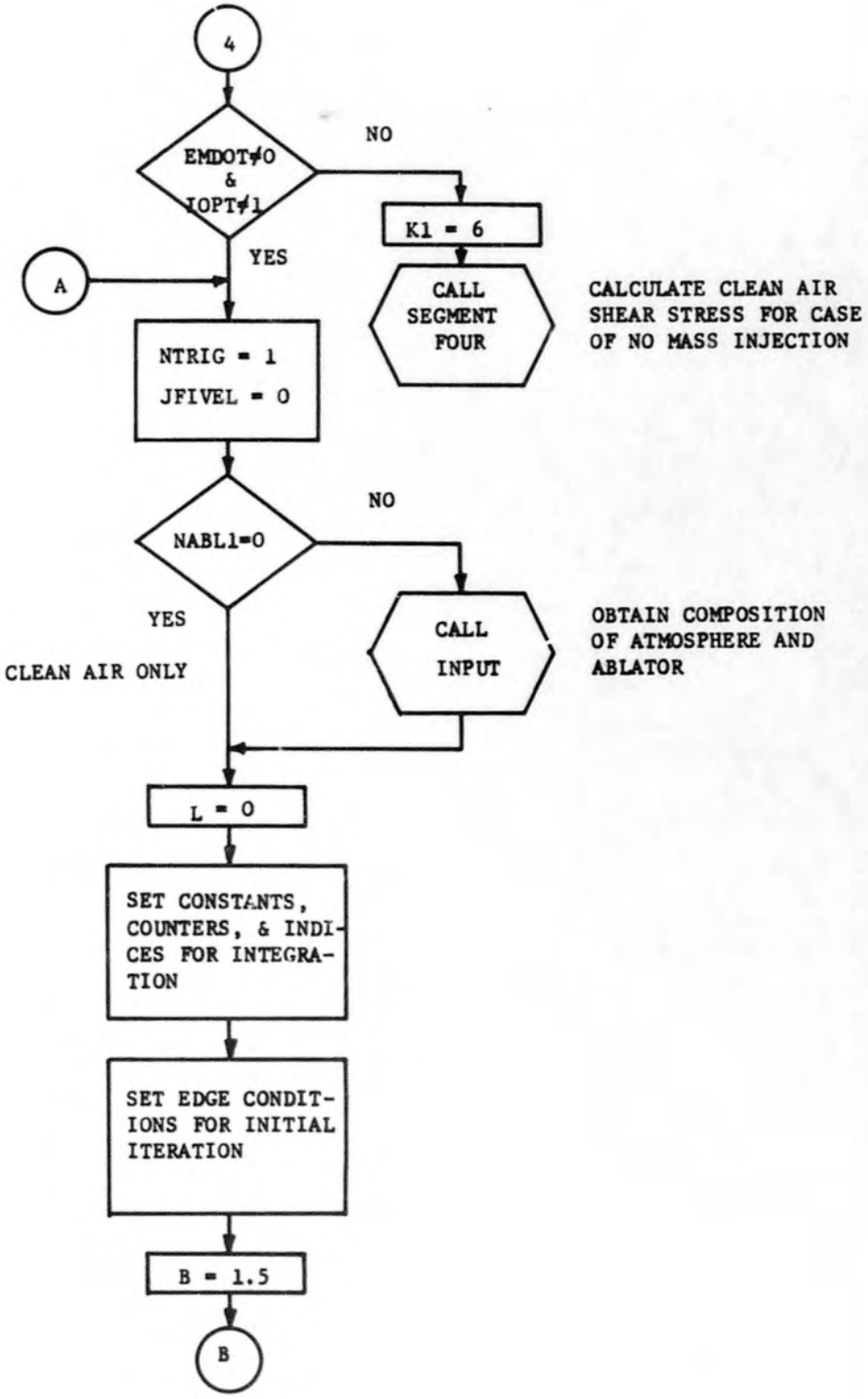
a. Function

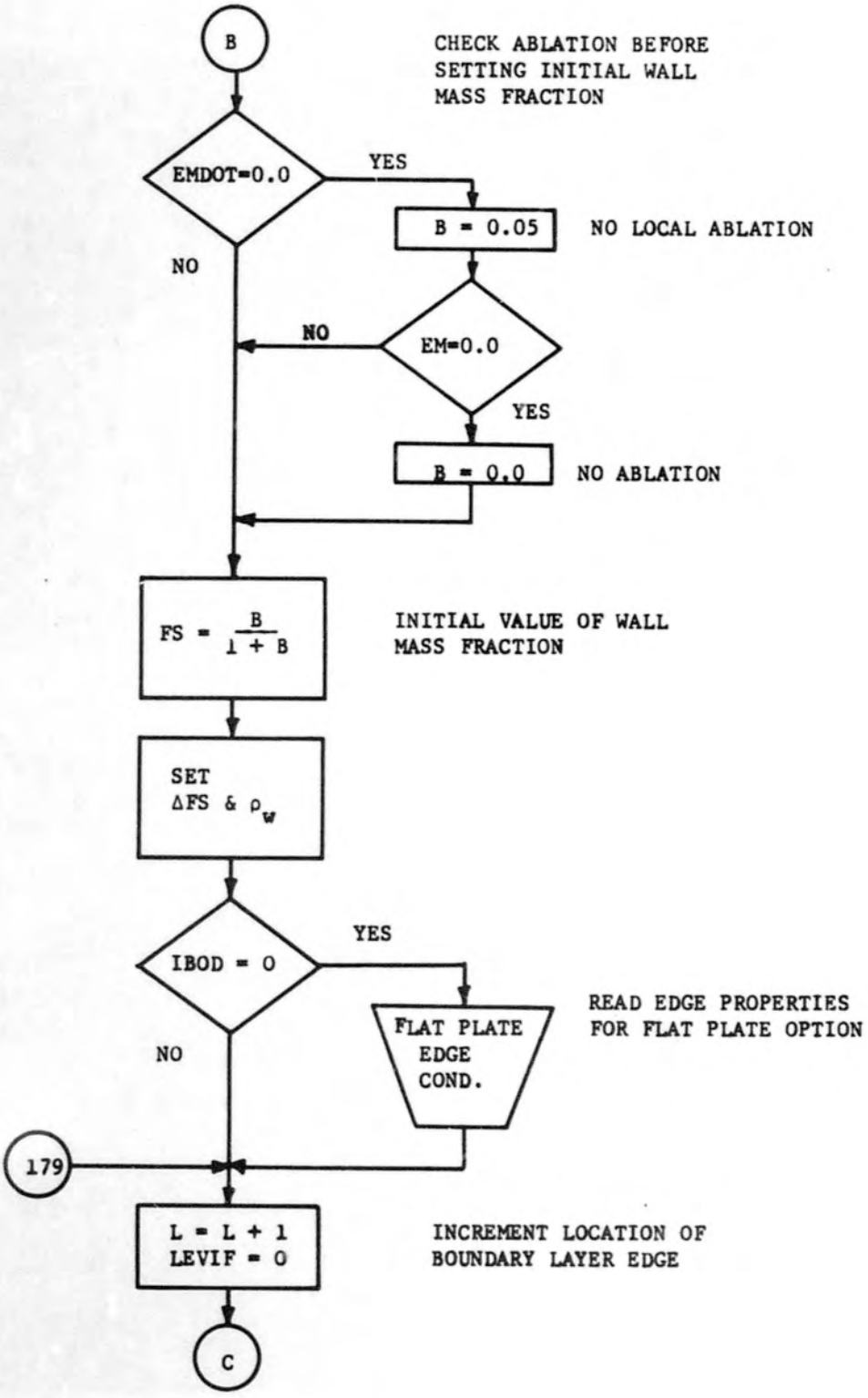
The boundary layer calculations described in Section II are performed in this segment. However, before the boundary layer solution is begun, this segment controls the order in which the other segments are called to read data or to determine the inviscid flow field and ablation mass loss. The boundary layer solution is started by assuming the wall values of pressure and velocity are the boundary layer edge values, and the wall mass fraction is 0.6. An ideal gas equation of state is used, and the injected gas flow rate is determined by integrating the mass fraction, density, and velocity through the boundary layer. If this integrated mass flow does not match the known injected mass flow, the wall mass fraction is adjusted until agreement is reached. Next, the air mass flow is checked. If this value does not agree with the inviscid air flow, the boundary layer edge is moved until both the ablation gas and air mass flow are conserved. When this condition is obtained, the ideal gas equation of state is replaced by a real gas equation of state (the equilibrium chemistry routine), and the process is repeated until the ablation and air flow rates are conserved. This real gas solution uses 16 points for the numerical integration, as was discussed in Section II. Once the solution converges, the program either begins a new solution or prepares an input tape used by Segments SIX or NINE to calculate the thermodynamic properties of the contaminated or uncontaminated (clean air) boundary layer gas. If the large chemistry calculations are to be performed, only one boundary layer solution may be run at a time.

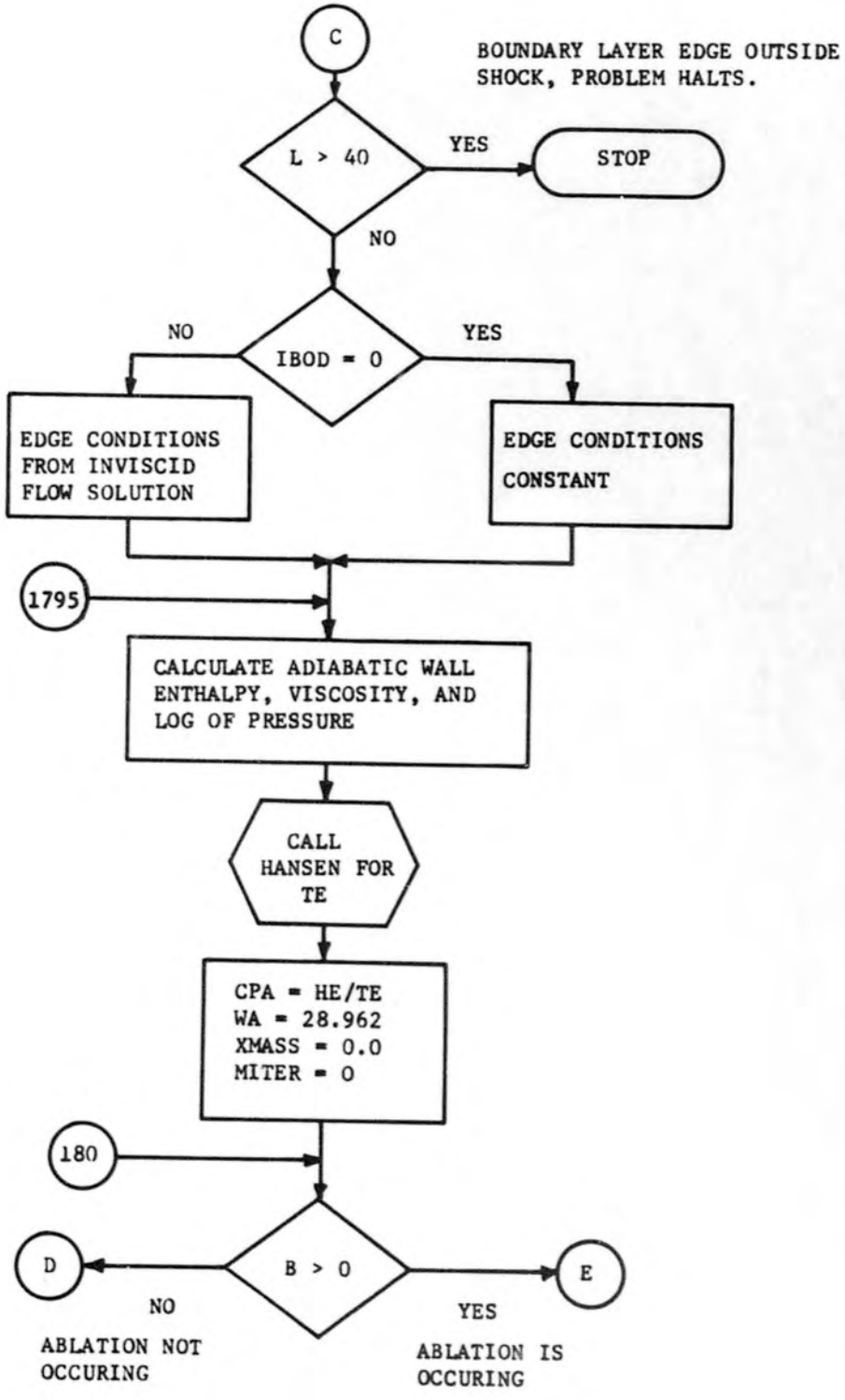
b. Flow Chart

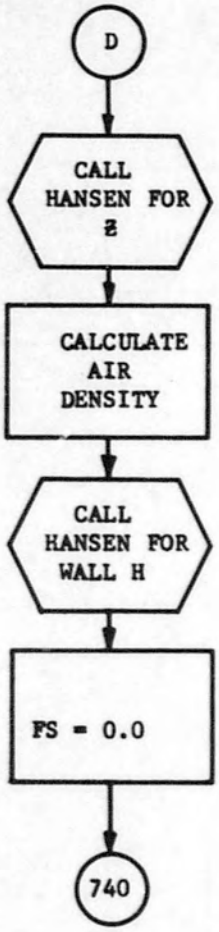

ENTER
ITEST = K1
ITEST=2
&
IBOD=0
YES
K1 = 6
TEST FOR FLAT PLATE OPTION
IGO = K1
IGO = 1
YES
99
NO
LOGIC CONTROLLING CALLING OF OTHER SEGMENTS
IGO = 2
YES
1
NO
IGO = 3
YES
2
NO
IGO = 4
YES
3
NO
IGO = 5
YES
4
NO
A


99
K1 = 2
CALL SEGMENT TWO
SEGMENT TWO READS BOUNDARY LAYER CASE DATA
1
TEST FOR THERMOCHEMISTRY OPTION
IEND=4
YES
NO
CALL SEGMENT EIGHT
DO THERMOCHEMISTRY ONLY
K1 = 3
CALL SEGMENT THREE
SHOCK SHAPE CALCULATION
2
TEST FOR MASS LOSS OPTION
3
IOPT = 1
YES
NO
K1 = 5
K1 = 4
CALL SEGMENT FIVE
CALCULATE INVISCID FLOW PROPERTIES
CALL SEGMENT FOUR
CALCULATE MASS LOSS WITH SEGMENT FOUR


4
EMDOT≠0
&
IOPT≠1
NO
K1 = 6
CALL
SEGMENT
FOUR
CALCULATE CLEAN AIR
SHEAR STRESS FOR CASE
OF NO MASS INJECTION
A
YES
NTRIG = 1
JFIVEL = 0
NABL1=0
NO
YES
CLEAN AIR ONLY
CALL
INPUT
OBTAIN COMPOSITION
OF ATMOSPHERE AND
ABLATOR
L = 0
SET CONSTANTS,
COUNTERS, & INDI-
CES FOR INTEGRA-
TION
SET EDGE CONDIT-
IONS FOR INITIAL
ITERATION
B = 1.5
B

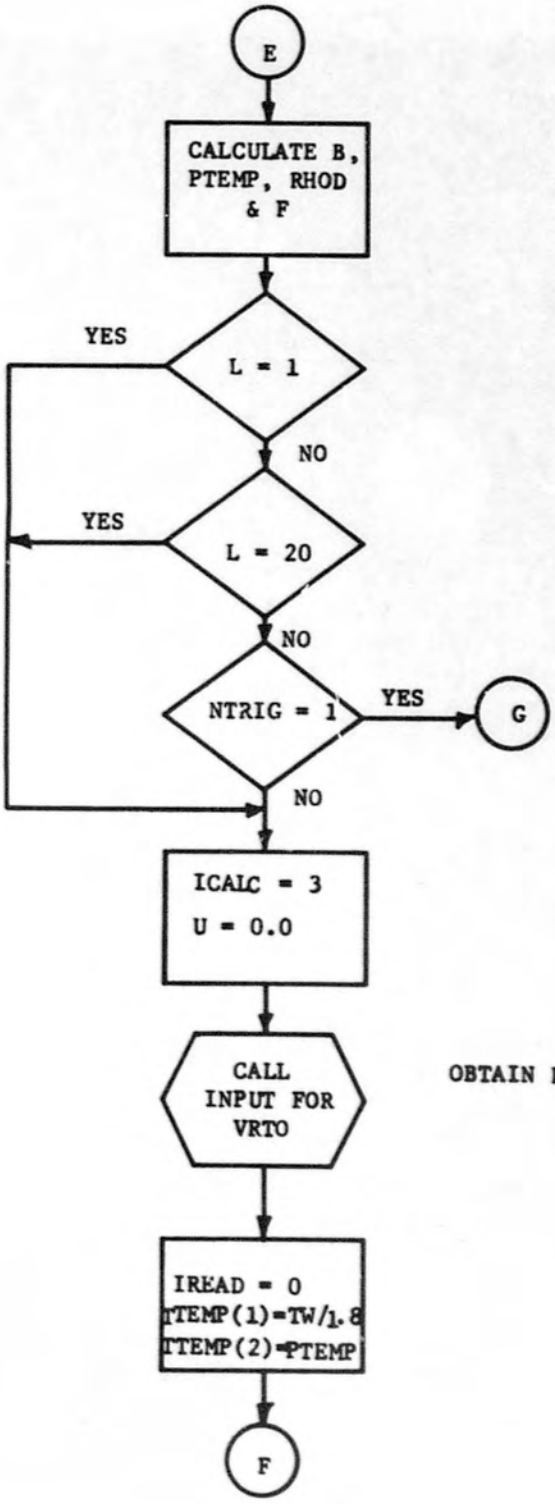

B
CHECK ABLATION BEFORE SETTING INITIAL WALL MASS FRACTION
EMDOT=0.0
YES
B = 0.05
NO LOCAL ABLATION
NO
NO
EM=0.0
YES
B = 0.0
NO ABLATION
FS = B / 1 + B
INITIAL VALUE OF WALL MASS FRACTION
SET ΔFS & ρw
IBOD = 0
YES
FLAT PLATE EDGE COND.
READ EDGE PROPERTIES FOR FLAT PLATE OPTION
NO
179
L = L + 1
LEVIF = 0
INCREMENT LOCATION OF BOUNDARY LAYER EDGE
C


C
BOUNDARY LAYER EDGE OUTSIDE SHOCK, PROBLEM HALTS.
L > 40
YES
STOP
NO
NO
IBOD = 0
YES
EDGE CONDITIONS FROM INVISCID FLOW SOLUTION
EDGE CONDITIONS CONSTANT
1795
CALCULATE ADIABATIC WALL ENTHALPY, VISCOSITY, AND LOG OF PRESSURE
CALL HANSEN FOR TE
CPA = HE/TE
WA = 28.962
XMASS = 0.0
MITER = 0
180
D
B > 0
E
NO
YES
ABLATION NOT OCCURING
ABLATION IS OCCURING

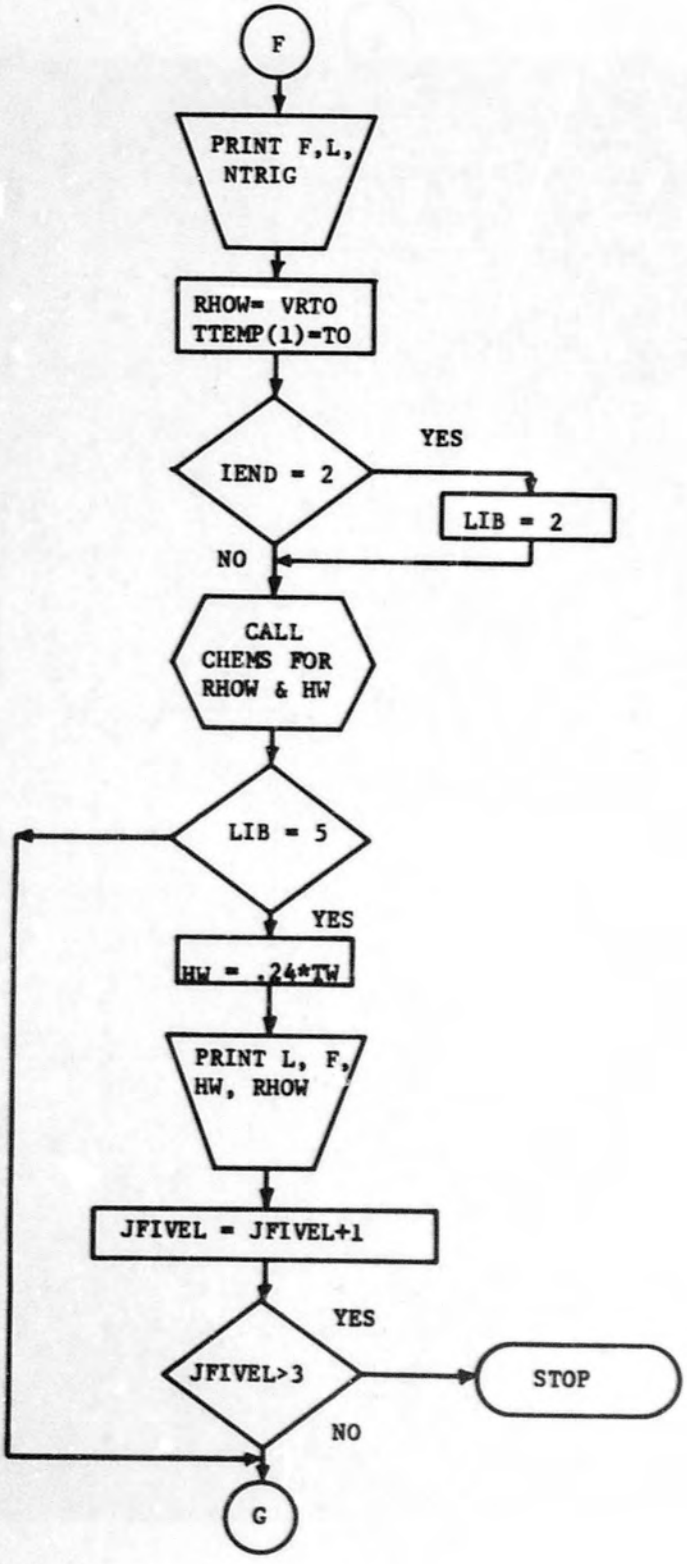


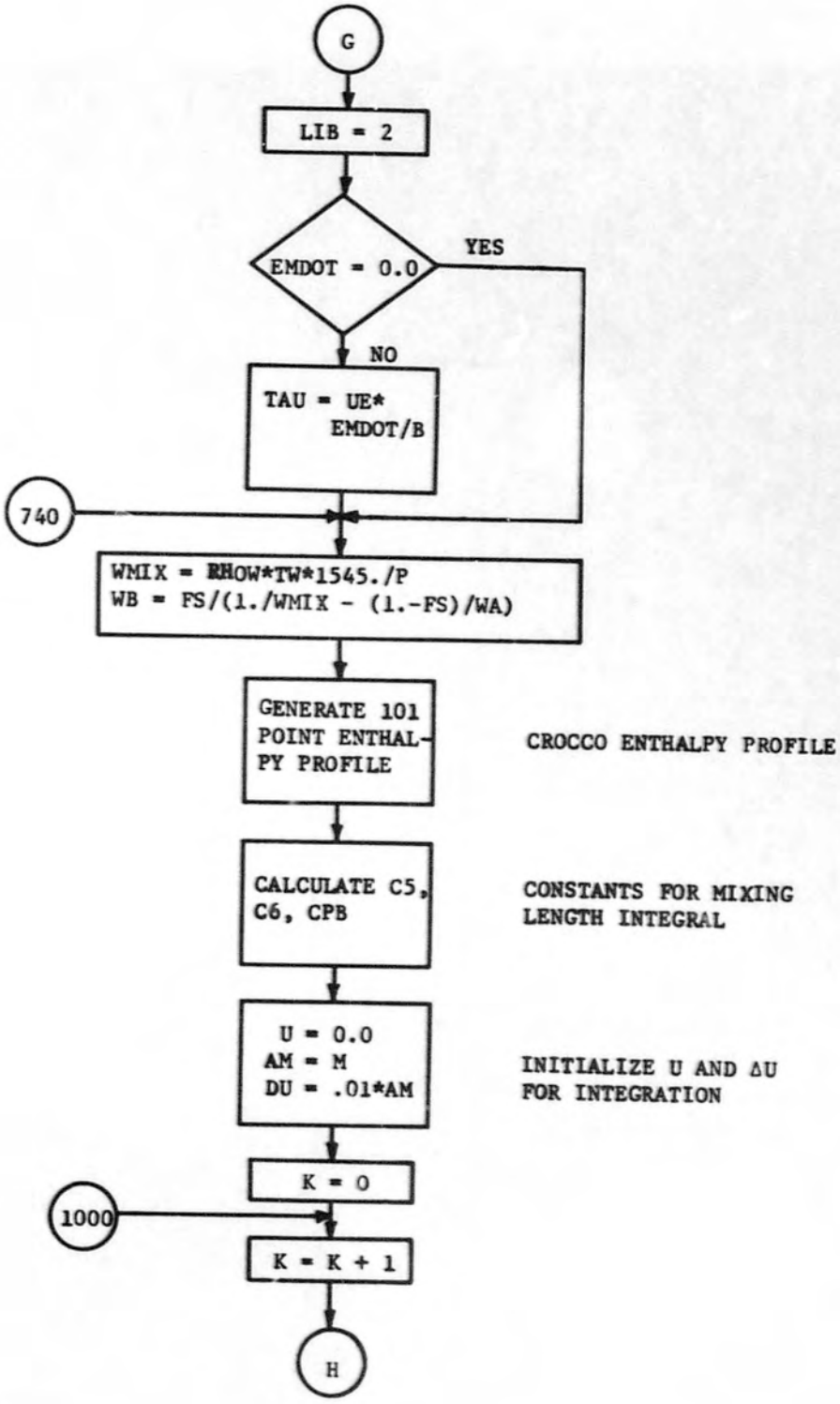
OBTAIN AIR COMPRESS-ABILITY FACTOR

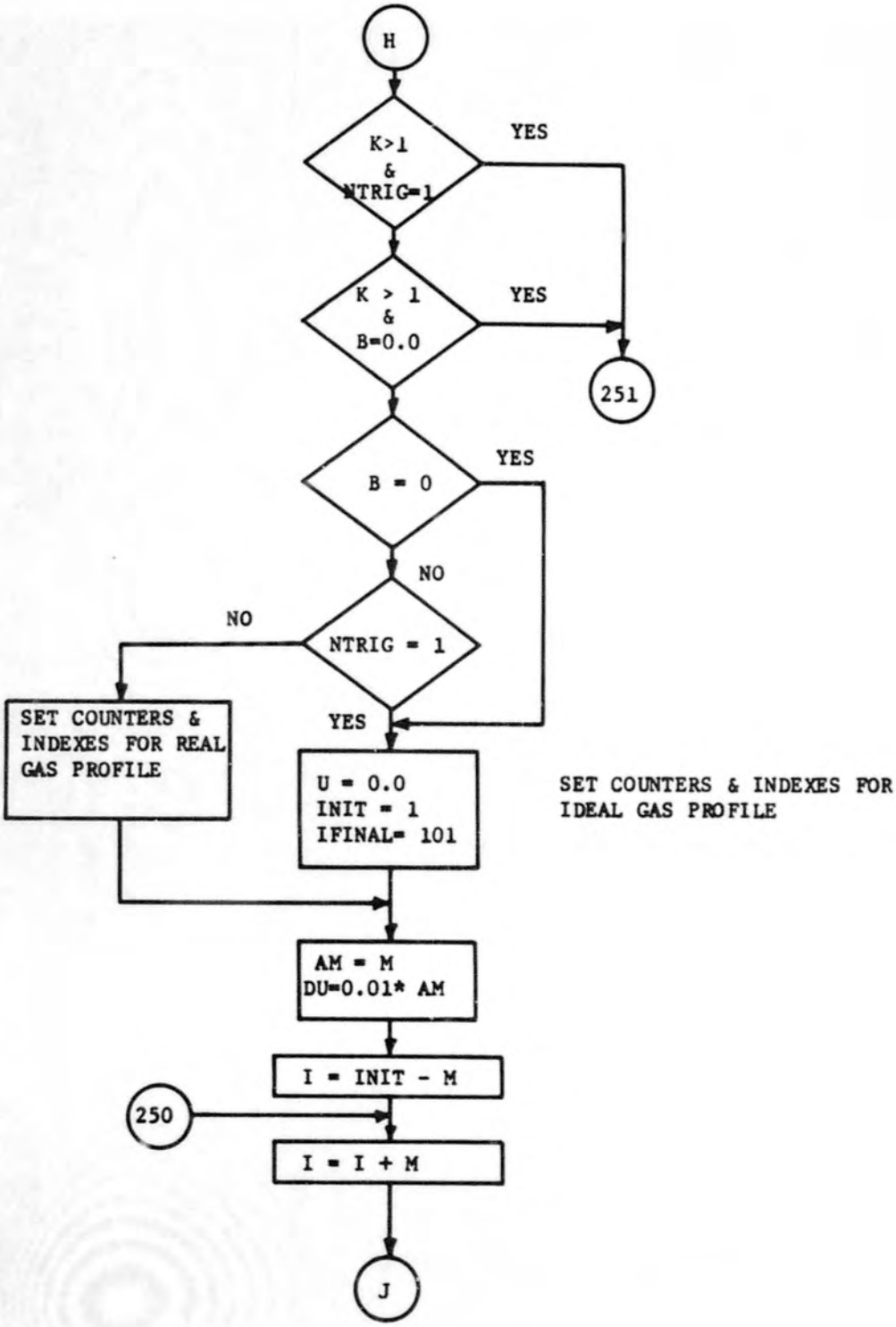
OBTAIN AIR ENTHALPY AT WALL CONDITIONS

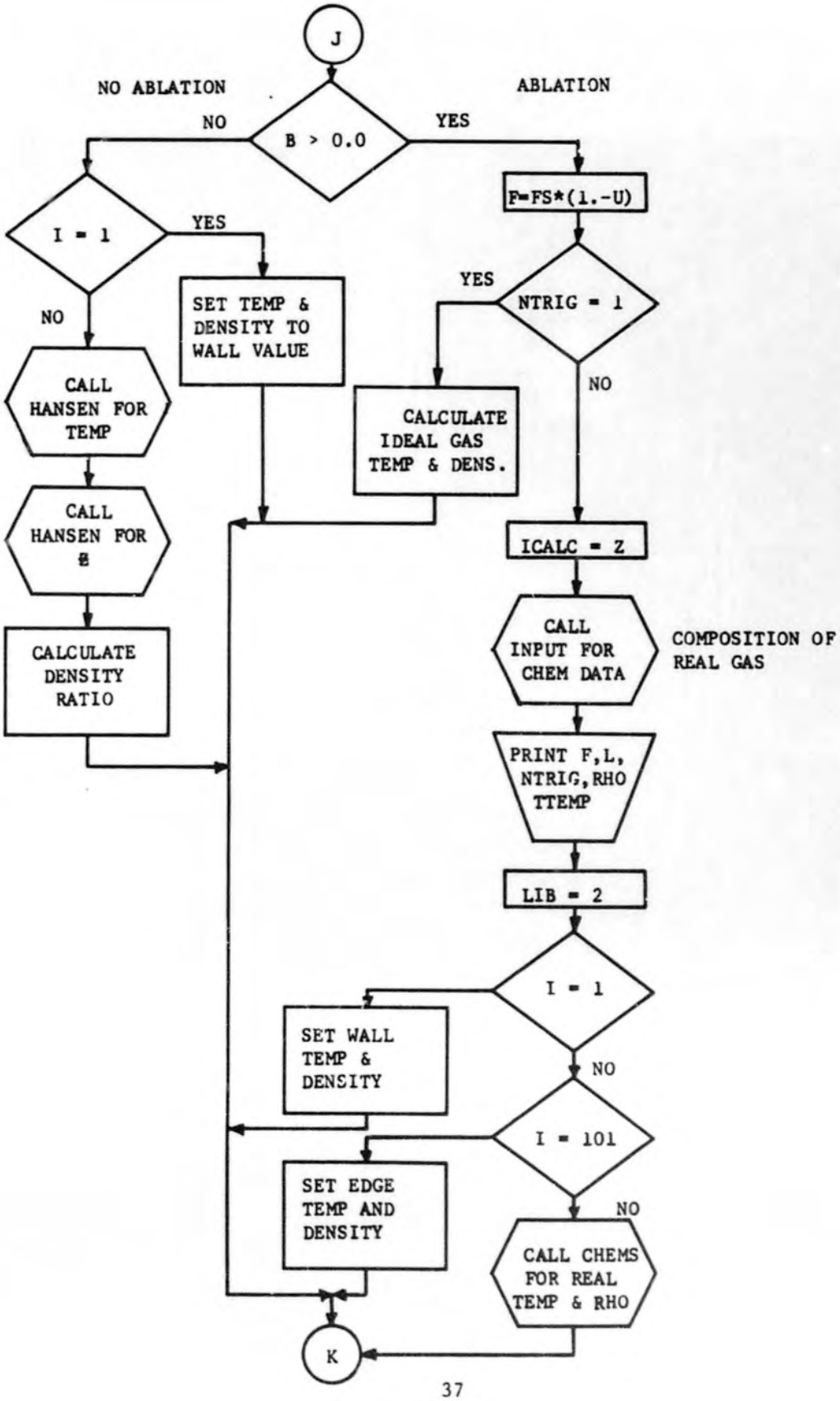


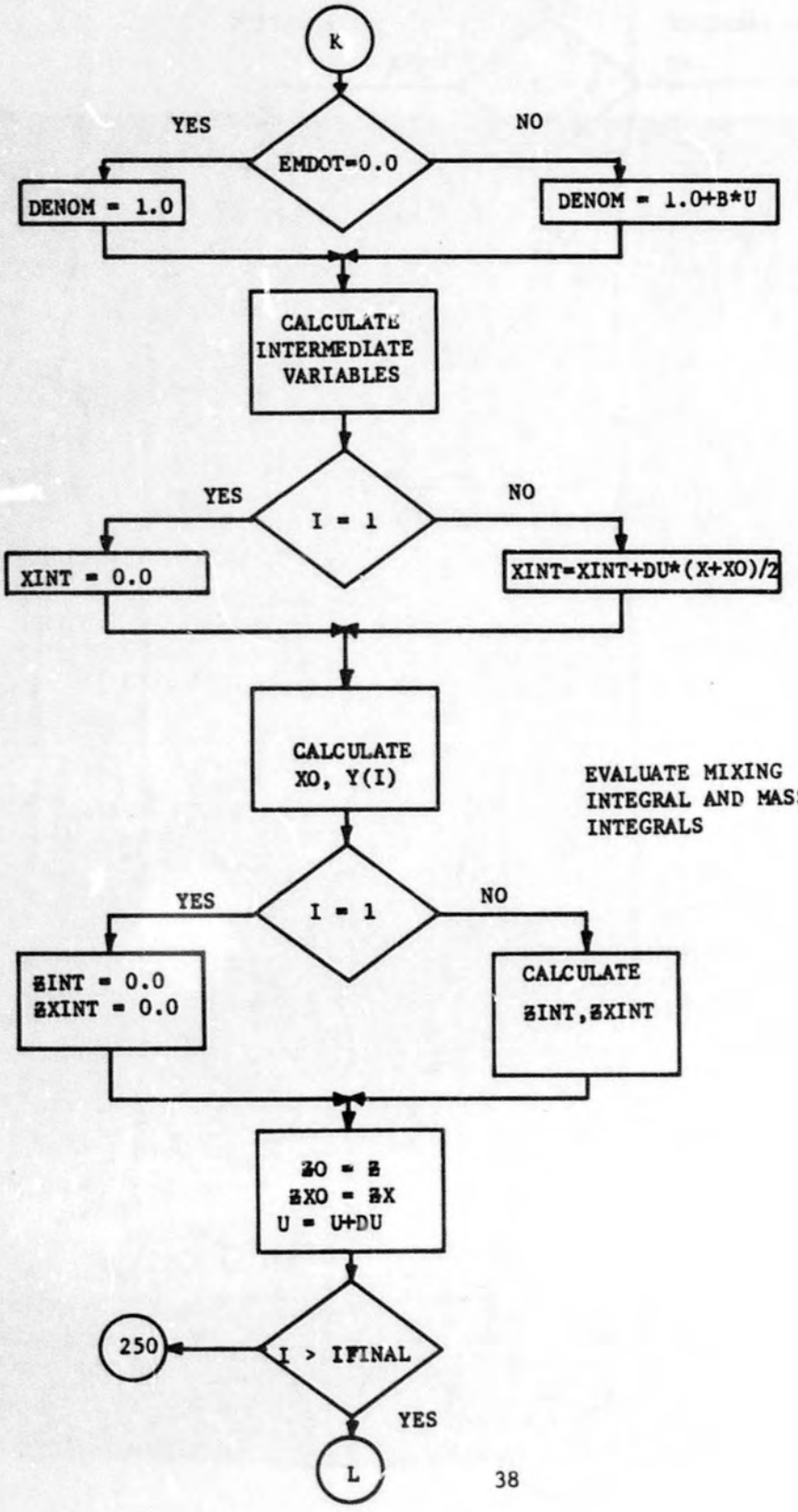
OBTAIN DENSITY RATIO

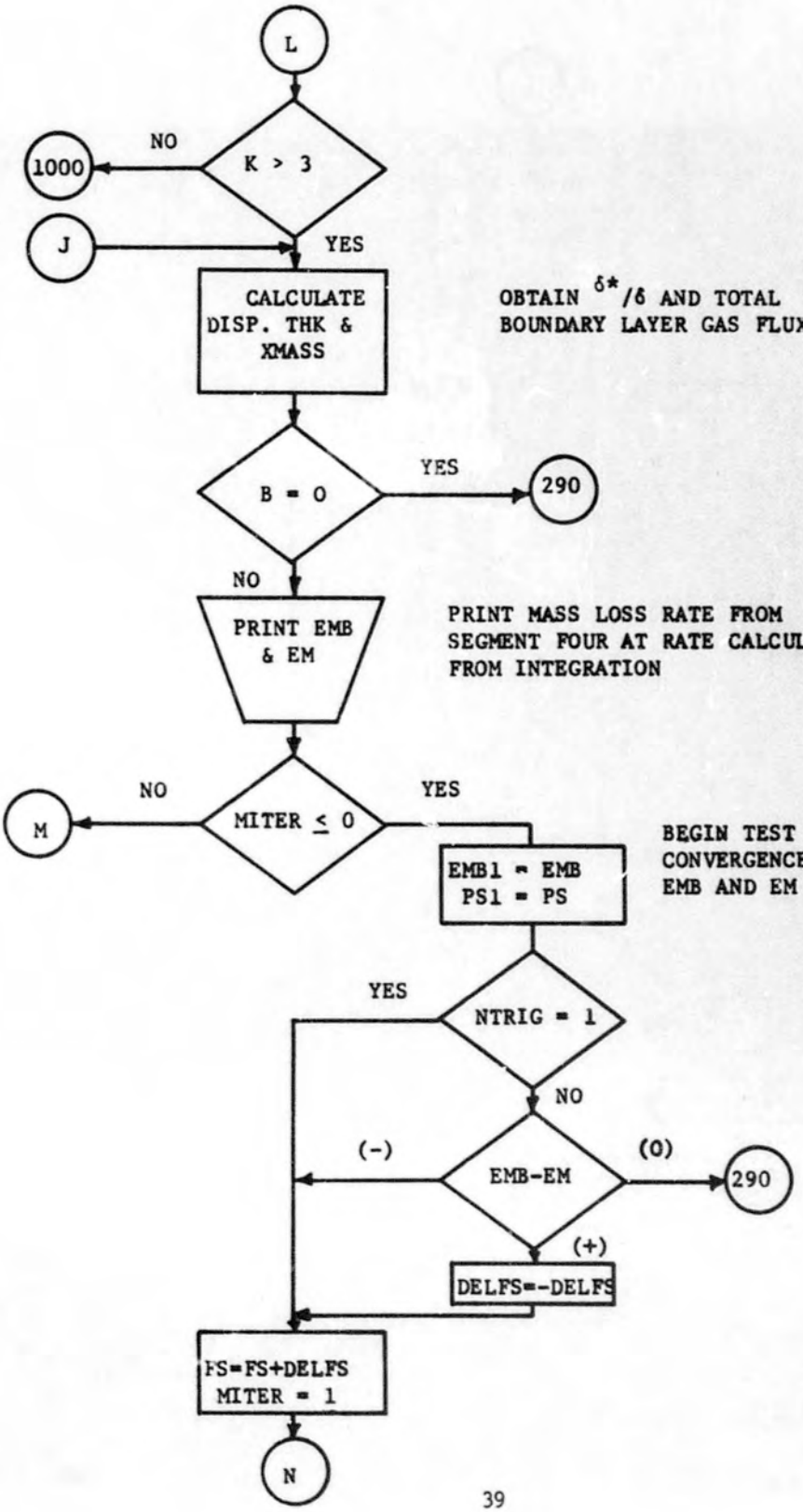

F
PRINT F,L,
NTRIG
RHOW= VRTO
TTEMP(1)=TO
IEND = 2
YES
LIB = 2
NO
CALL
CHEMS FOR
RHOW & HW
LIB = 5
YES
HW = .24*TW
PRINT L, F,
HW, RHOW
JFIVEL = JFIVEL+1
YES
JFIVEL>3
STOP
NO
G

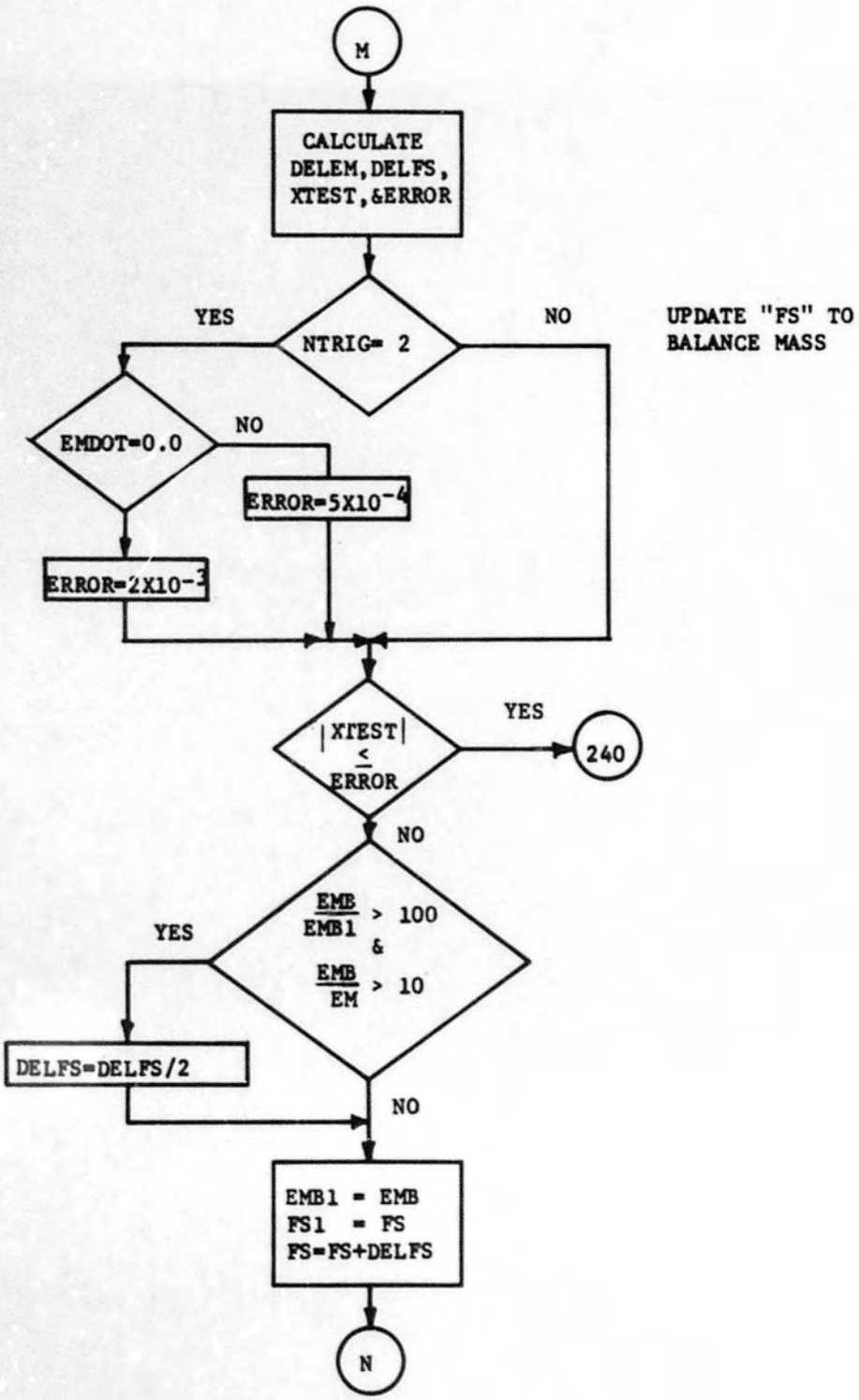

G
LIB = 2
EMDOT = 0.0
YES
NO
TAU = UE*
EMDOT/B
740
WMIX = RHOW*TW*1545./P
WB = FS/(1./WMIX - (1.-FS)/WA)
GENERATE 101
POINT ENTHAL-
PY PROFILE
CROCCO ENTHALPY PROFILE
CALCULATE C5,
C6, CPB
CONSTANTS FOR MIXING
LENGTH INTEGRAL
U = 0.0
AM = M
DU = .01*AM
INITIALIZE U AND ΔU
FOR INTEGRATION
K = 0
1000
K = K + 1
H

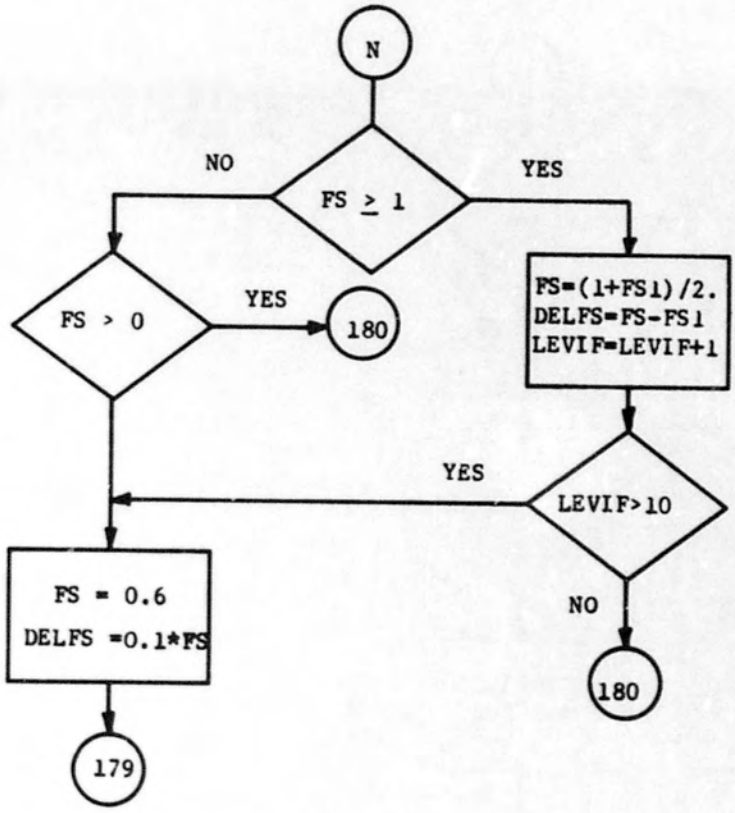

H
K>1 & NTRIG=1
YES
K > 1 & B=0.0
YES
251
B = 0
YES
NO
NTRIG = 1
NO
SET COUNTERS & INDEXES FOR REAL GAS PROFILE
YES
U = 0.0
INIT = 1
IFINAL= 101
SET COUNTERS & INDEXES FOR IDEAL GAS PROFILE
AM = M
DU=0.01* AM
I = INIT - M
250
I = I + M
J

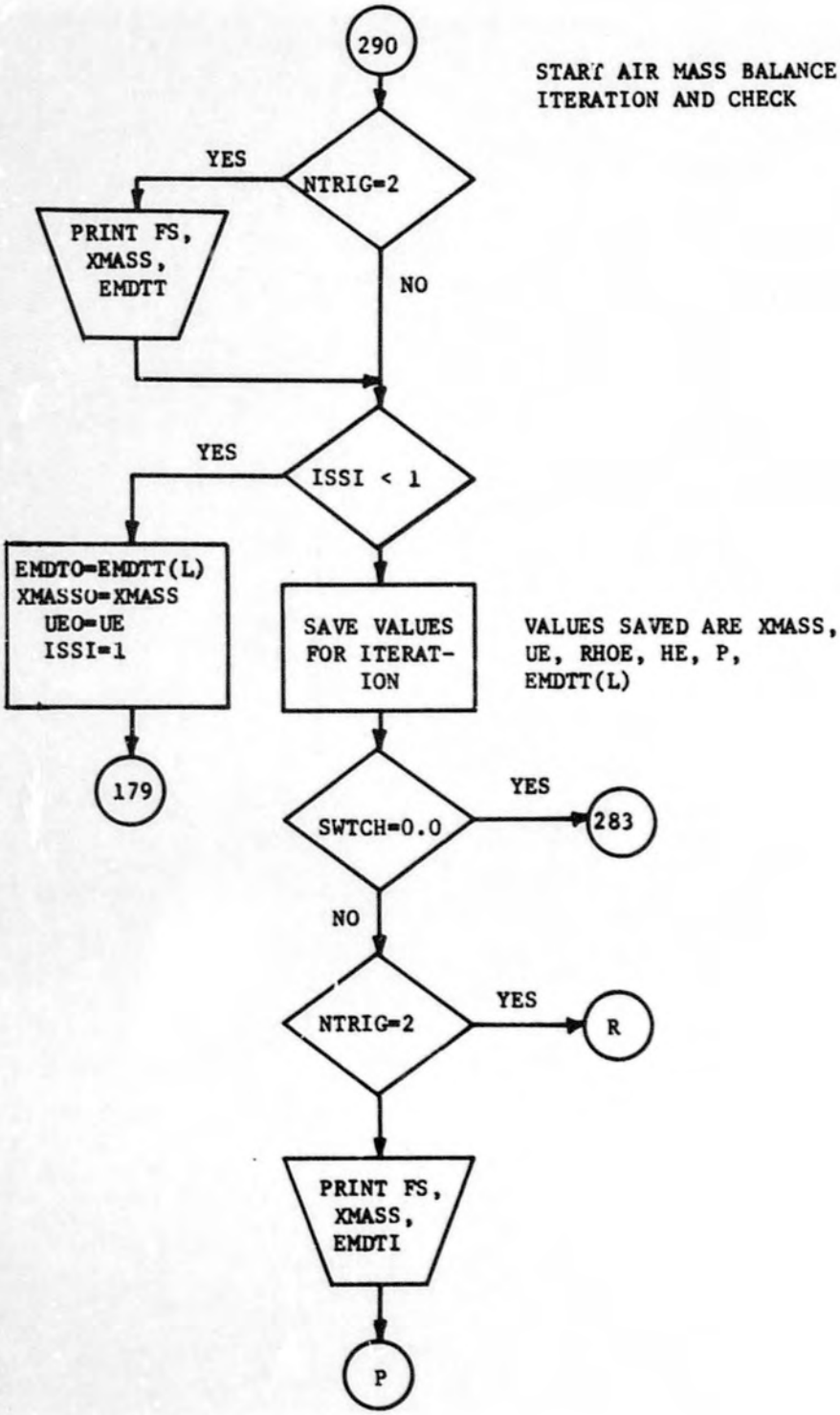

J
NO ABLATION
ABLATION
NO
B > 0.0
YES
I = 1
YES
NO
SET TEMP & DENSITY TO WALL VALUE
CALL HANSEN FOR TEMP
CALL HANSEN FOR ρ
CALCULATE DENSITY RATIO
F=FS*(1.-U)
YES
NTRIG = 1
NO
CALCULATE IDEAL GAS TEMP & DENS.
ICALC = 2
CALL INPUT FOR CHEM DATA
COMPOSITION OF REAL GAS
PRINT F,L, NTRIG,RHO TTEMP
LIB = 2
I = 1
SET WALL TEMP & DENSITY
NO
I = 101
SET EDGE TEMP AND DENSITY
NO
CALL CHEMS FOR REAL TEMP & RHO
K

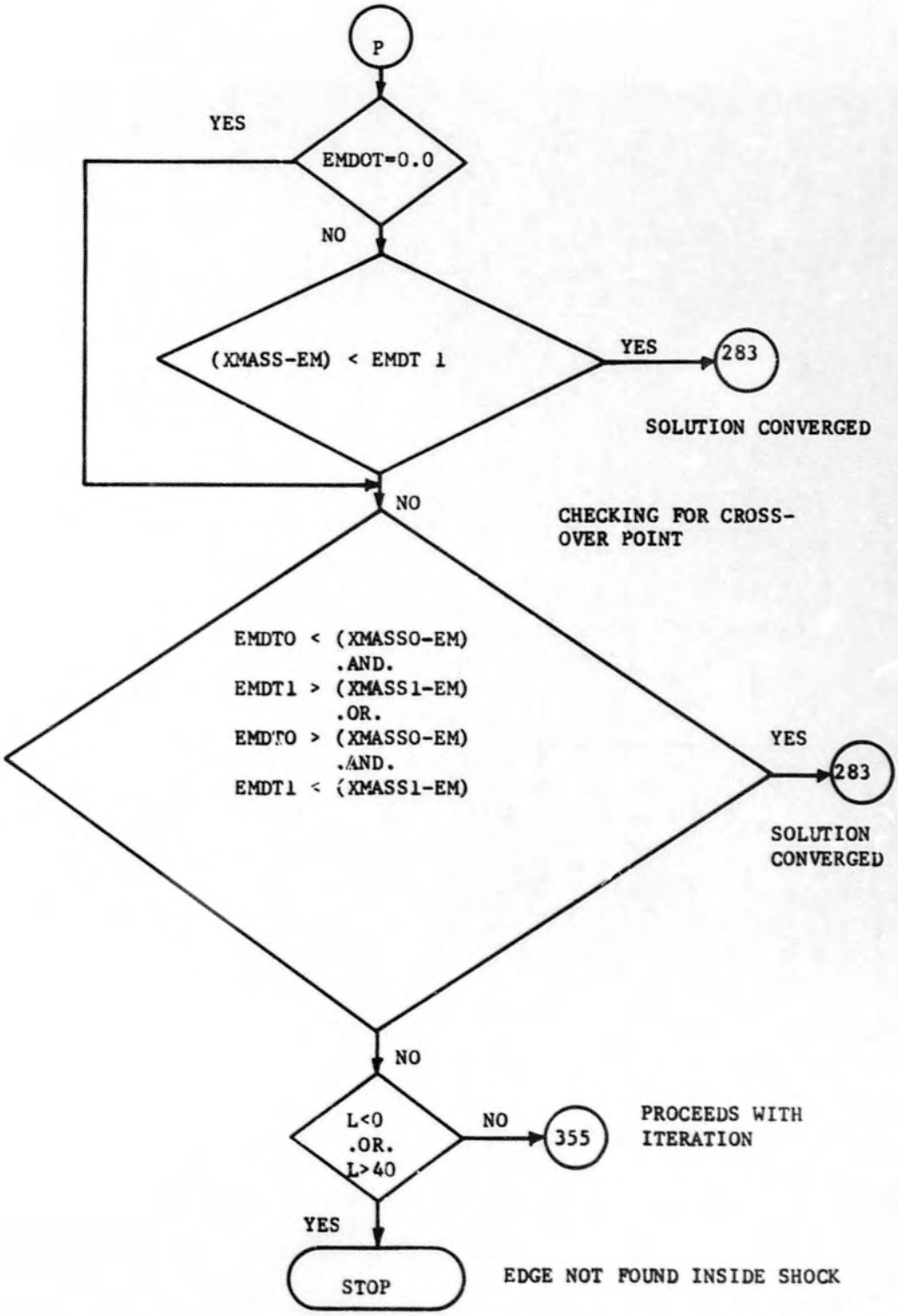

K
YES
NO
EMDOT=0.0
DENOM = 1.0
DENOM = 1.0+B*U
CALCULATE INTERMEDIATE VARIABLES
YES
NO
I = 1
XINT = 0.0
XINT=XINT+DU*(X+XO)/2
CALCULATE XO, Y(I)
EVALUATE MIXING LENGTH INTEGRAL AND MASS FLUX INTEGRALS
YES
NO
I = 1
ZINT = 0.0
ZXINT = 0.0
CALCULATE ZINT,ZXINT
ZO = Z
ZXO = ZX
U = U+DU
250
I > IFINAL
YES
L

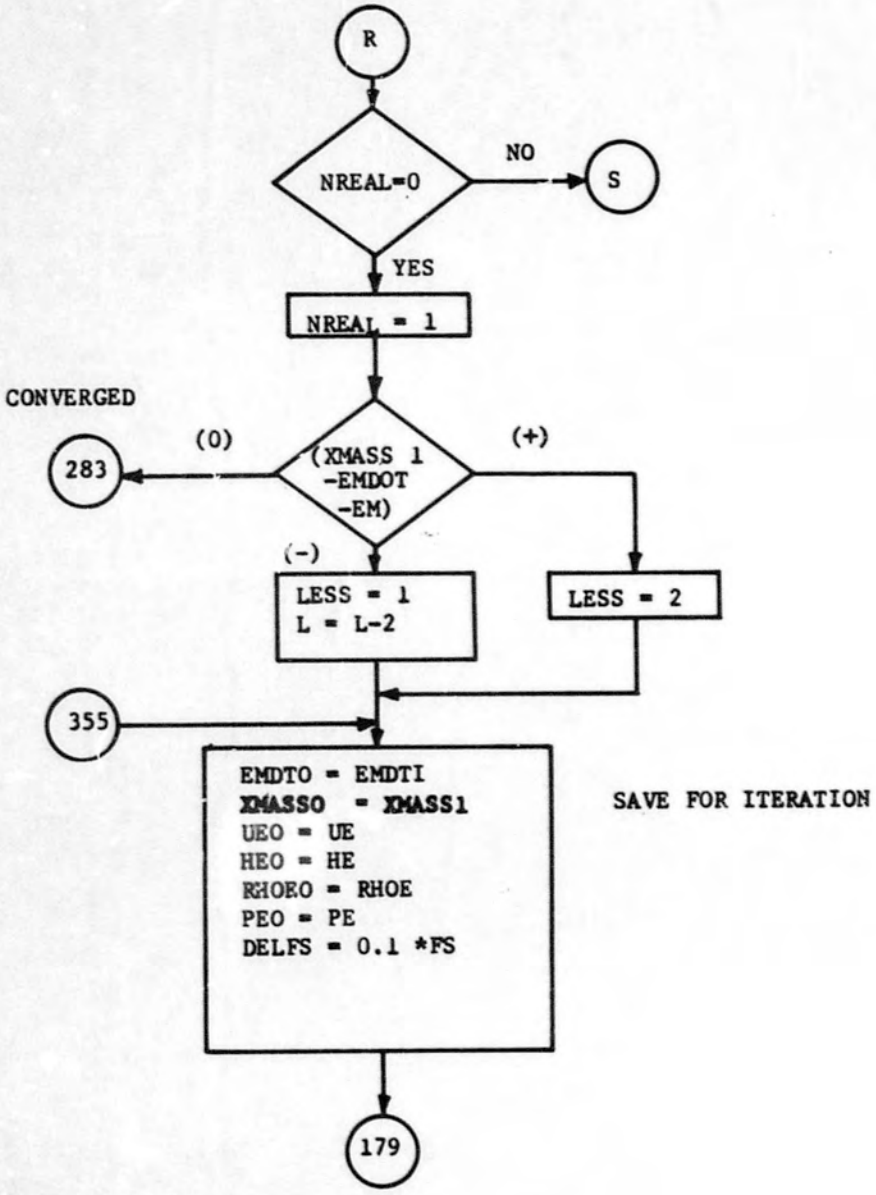

L
K > 3
NO
1000
J
YES
CALCULATE DISP. THK & XMASS
OBTAIN δ*/δ AND TOTAL BOUNDARY LAYER GAS FLUX
B = 0
YES
290
NO
PRINT EMB & EM
PRINT MASS LOSS RATE FROM SEGMENT FOUR AT RATE CALCULATED FROM INTEGRATION
MITER ≤ 0
NO
M
YES
EMB1 = EMB
PS1 = PS
BEGIN TEST FOR CONVERGENCE OF EMB AND EM
NTRIG = 1
YES
NO
EMB-EM
(-)
(0)
290
(+)
DELFS=-DELFS
FS=FS+DELFS
MITER = 1
N

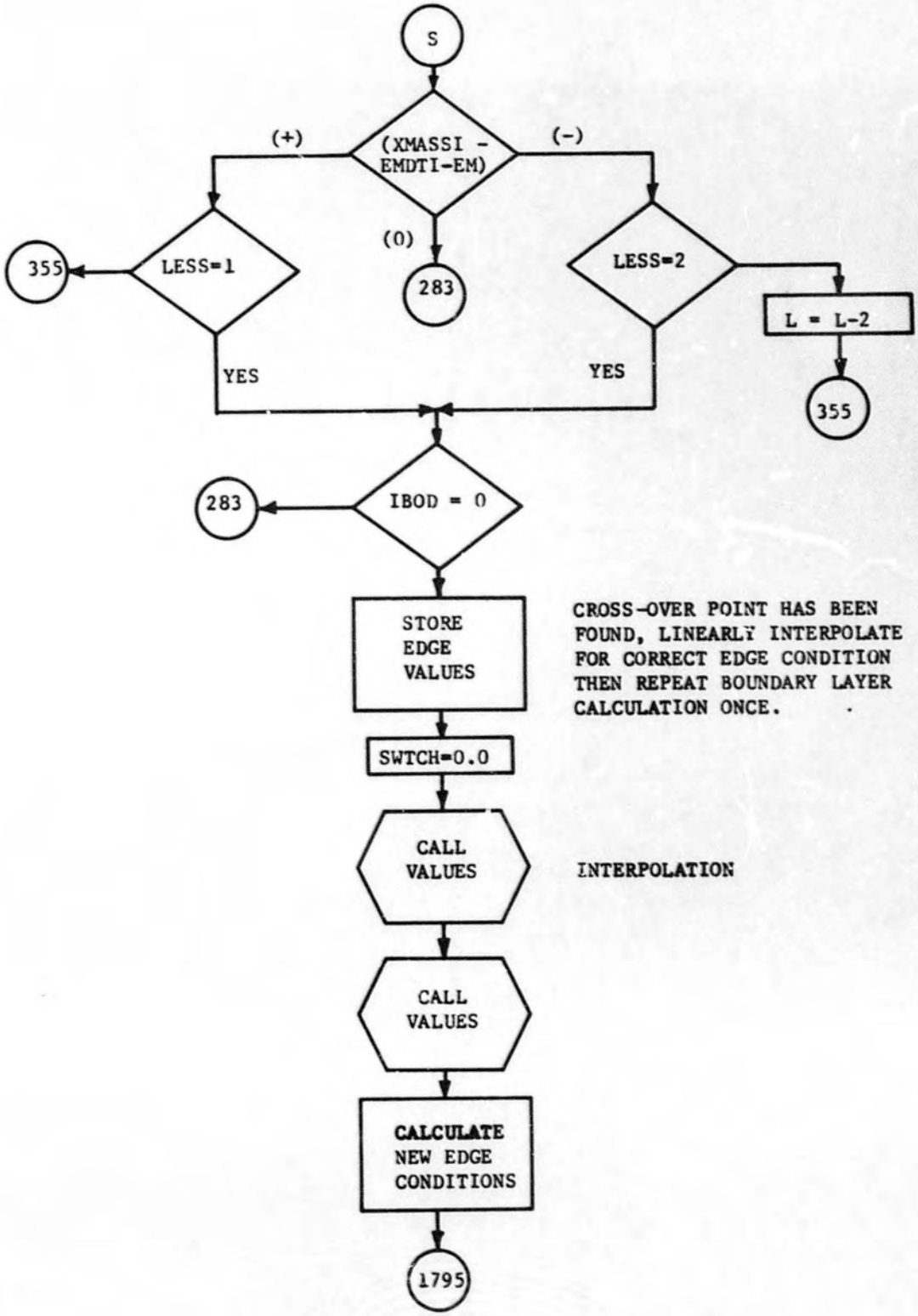

M
CALCULATE DELEM, DELFS, XTEST, &ERROR
YES
NTRIG= 2
NO
UPDATE "FS" TO BALANCE MASS
EMDOT=0.0
NO
ERROR=5X10^-4
ERROR=2X10^-3
|XTEST| ≤ ERROR
YES
240
NO
EMB/EMB1 > 100 & EMB/EM > 10
YES
DELFS=DELFS/2
NO
EMB1 = EMB
FS1 = FS
FS=FS+DELFS
N

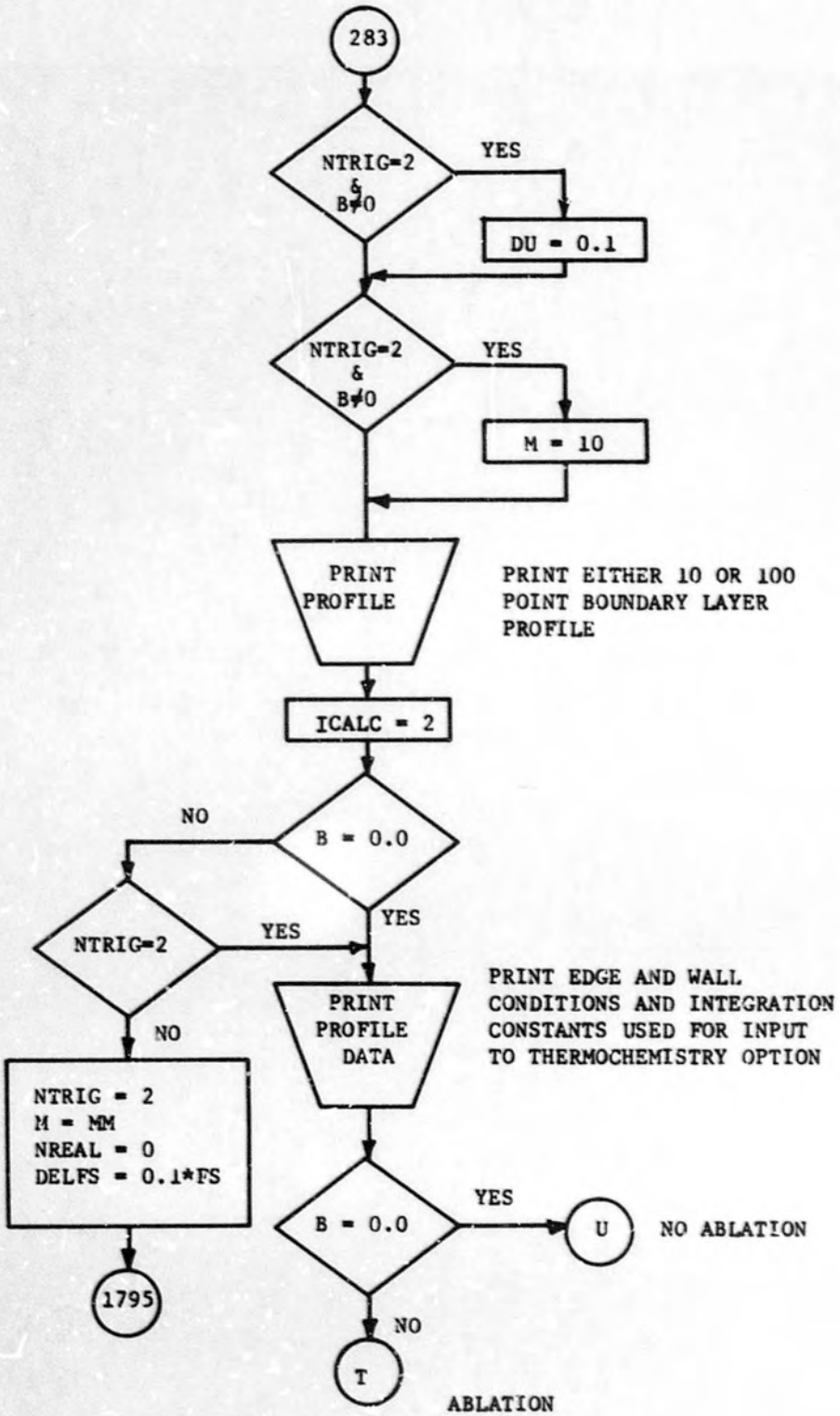

N
NO
FS ≥ 1
YES
FS > 0
YES
180
FS=(1+FS1)/2.
DELFS=FS-FS1
LEVIF=LEVIF+1
YES
LEVIF>10
FS = 0.6
DELFS =0.1*FS
NO
180
179

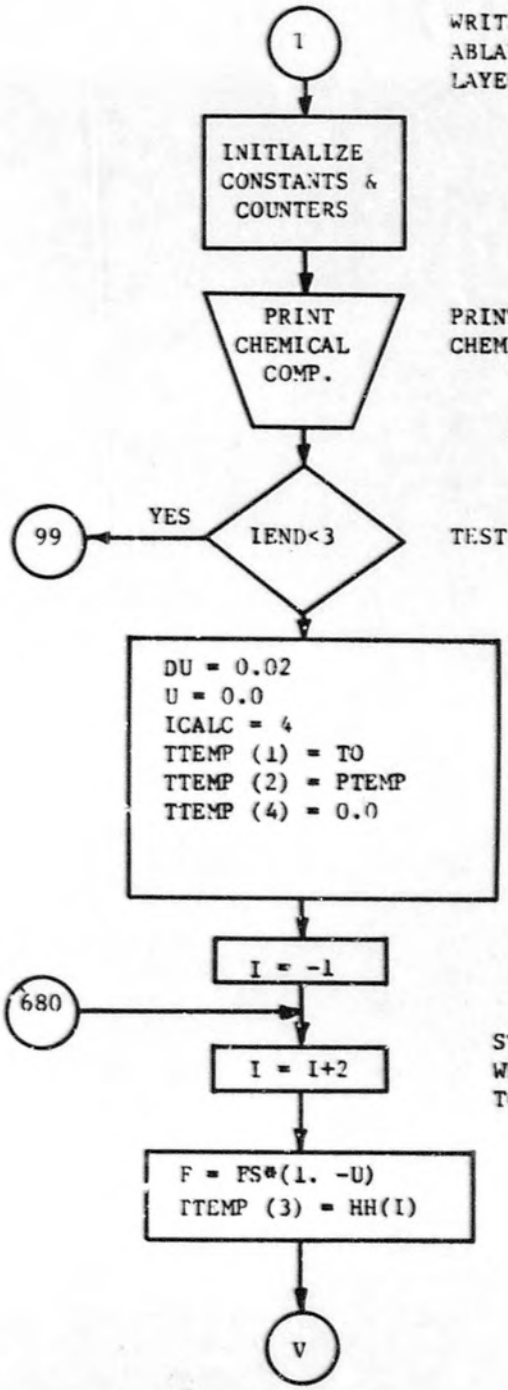

290
START AIR MASS BALANCE ITERATION AND CHECK
YES
NTRIG=2
PRINT FS, XMASS, EMDTT
NO
YES
ISSI < 1
EMDTO=EMDTT(L)
XMASSO=XMASS
UEO=UE
ISSI=1
SAVE VALUES FOR ITERAT-ION
VALUES SAVED ARE XMASS, UE, RHOE, HE, P, EMDTT(L)
179
YES
SWTCH=0.0
283
NO
YES
NTRIG=2
R
PRINT FS, XMASS, EMDTI
P

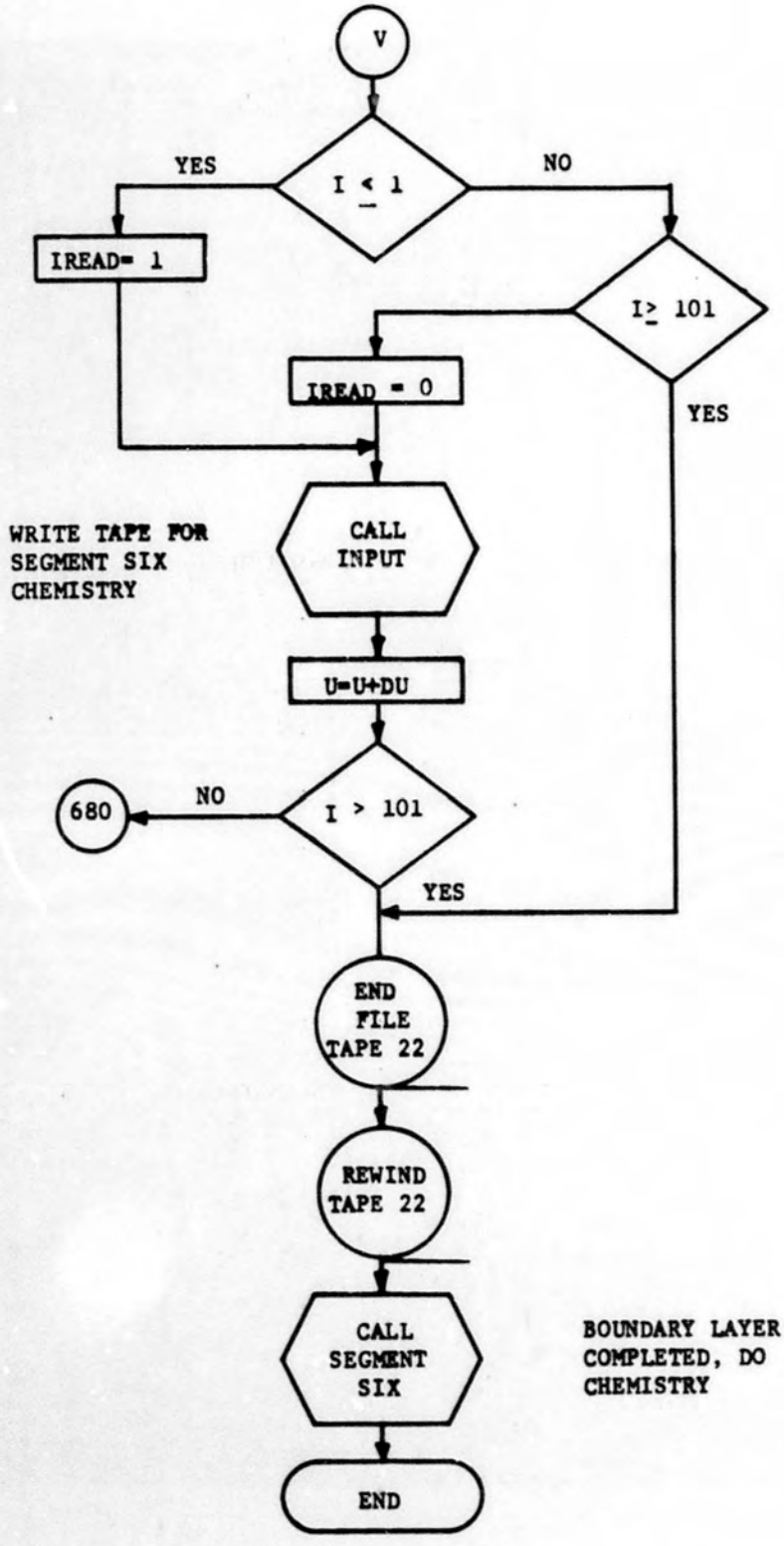

P
YES
EMDOT=0.0
NO
(XMASS-EM) < EMDT 1
YES
283
SOLUTION CONVERGED
NO
CHECKING FOR CROSS-OVER POINT
EMDTO < (XMASSO-EM)
.AND.
EMDT1 > (XMASS1-EM)
.OR.
EMDTO > (XMASSO-EM)
.AND.
EMDT1 < (XMASS1-EM)
YES
283
SOLUTION CONVERGED
NO
L<0
.OR.
L>40
NO
355
PROCEEDS WITH ITERATION
YES
STOP
EDGE NOT FOUND INSIDE SHOCK

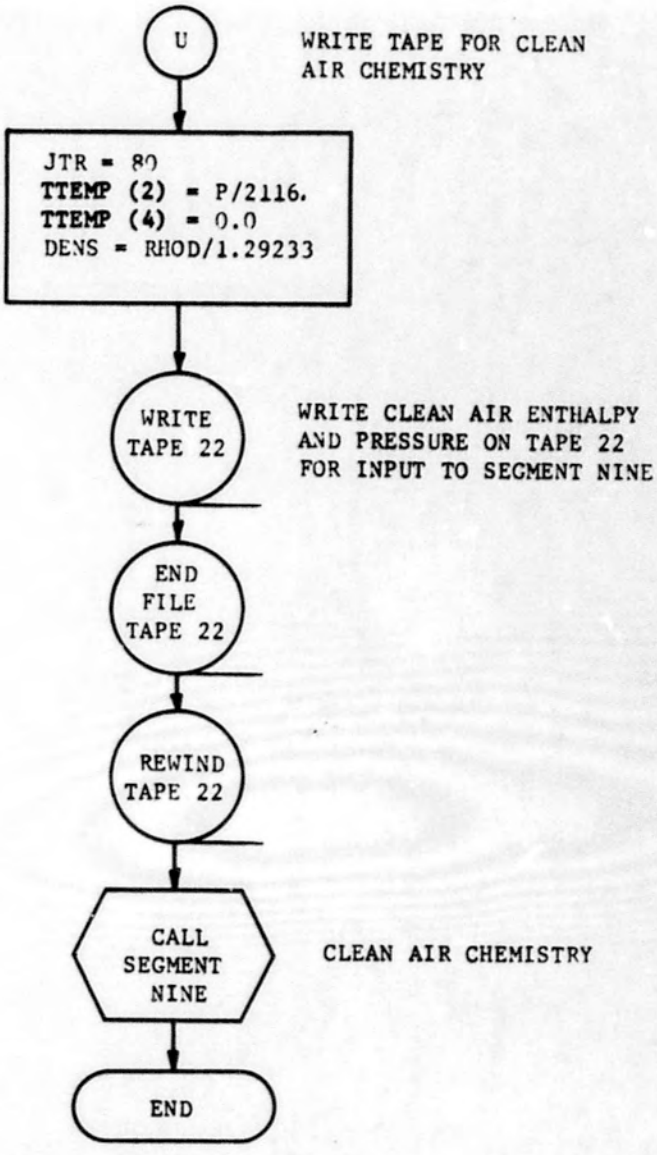

R
NREAL=0
NO
S
YES
NREAL = 1
CONVERGED
(0)
283
(XMASS 1
-EMDOT
-EM)
(+)
(-)
LESS = 1
L = L-2
LESS = 2
355
EMDTO = EMDTI
XMASSO = XMASS1
UEO = UE
HEO = HE
RHOEO = RHOE
PEO = PE
DELFS = 0.1 *FS
SAVE FOR ITERATION
179


S
(XMASSI - EMDTI-EM)
(+)
(0)
(-)
355
LESS=1
283
LESS=2
L = L-2
355
YES
YES
283
IBOD = 0
STORE EDGE VALUES
SWTCH=0.0
CALL VALUES
CALL VALUES
CALCULATE NEW EDGE CONDITIONS
1795
CROSS-OVER POINT HAS BEEN FOUND, LINEARLY INTERPOLATE FOR CORRECT EDGE CONDITION THEN REPEAT BOUNDARY LAYER CALCULATION ONCE.
INTERPOLATION


283
NTRIG=2 & B≠0
YES
DU = 0.1
NTRIG=2 & B≠0
YES
M = 10
PRINT PROFILE
PRINT EITHER 10 OR 100 POINT BOUNDARY LAYER PROFILE
ICALC = 2
B = 0.0
NO
YES
NTRIG=2
YES
NO
PRINT PROFILE DATA
PRINT EDGE AND WALL CONDITIONS AND INTEGRATION CONSTANTS USED FOR INPUT TO THERMOCHEMISTRY OPTION
NTRIG = 2
M = MM
NREAL = 0
DELFS = 0.1*FS
B = 0.0
YES
U
NO ABLATION
1795
NO
T
ABLATION


1
WRITE TAPE FOR AIR-ABLATION PRODUCT BOUNDARY LAYER CHEMISTRY
INITIALIZE CONSTANTS & COUNTERS
PRINT CHEMICAL COMP.
PRINT 10 POINT REAL GAS CHEMISTRY COMPOSITION
YES
99
IEND<3
TEST FOR OPTION
DU = 0.02
U = 0.0
ICALC = 4
TTEMP (1) = TO
TTEMP (2) = PTEMP
TTEMP (4) = 0.0
I = -1
680
I = I+2
START OF LOOP TO WRITE TAPE FOR INPUT TO SEGMENT SIX
F = FS*(1. -U)
TTEMP (3) = HH(I)
V


V
YES
I ≤ 1
NO
IREAD= 1
I≥ 101
IREAD = 0
YES
WRITE TAPE FOR SEGMENT SIX CHEMISTRY
CALL INPUT
U=U+DU
680
NO
I > 101
YES
END FILE TAPE 22
REWIND TAPE 22
CALL SEGMENT SIX
BOUNDARY LAYER COMPLETED, DO CHEMISTRY
END


U
WRITE TAPE FOR CLEAN AIR CHEMISTRY
JTR = 80
TTEMP (2) = P/2116.
TTEMP (4) = 0.0
DENS = RHOD/1.29233
WRITE TAPE 22
WRITE CLEAN AIR ENTHALPY AND PRESSURE ON TAPE 22 FOR INPUT TO SEGMENT NINE
END FILE TAPE 22
REWIND TAPE 22
CALL SEGMENT NINE
CLEAN AIR CHEMISTRY
END

```
C.  LISTING

      SEGMENT ONE(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
      COMMON /234/ SSSS(3960)
      COMMON /333/ JFS(150)
      COMMON /367/ AM367(150)
      COMMON / 456/ INTAP,IOTAP,INT,NMN,M,NGR
      COMMON /666/ SSS(292)
      COMMON /716/   TITLE,  IOPT,  IBOD,  IEND, NABL1, NABL2, IREAD,
     1  KODE,IHEMIS, KODEW,
     2  ANTL,    RN,   ALT,   VEL, XINCH,    Y1,  SMXB,  XMXB, DYDX1,
     3  CP10,    TW,    EM, EMDOT,    WA, WABL1, WABL2,   YBB,
     4    K1,     N,    P1,  PHO1,   ZMD,    ZS,     P,    YB,    VM,
     5   UEV,   HEH, RHOER, EMDTT,    TO,  RHOW,    HW,XMASSO, EMDTO,
     6  VRTO,   II8,   II9,     R,   TAU,   CI ,NTOTAL,NWHICH,  IJJJ,
     7     L
      COMMON /567/ PS,THS,PSTRM(40)
      COMMON /1/ SOS,TATMOS,RHO2
      COMMON /222/ Y,FS,RHOOW,TRS,CFS,ELS,HZ,HAW,HE,RHW,TITL(5),
     * C5,C6,UE,B,CIP
      EQUIVALENCE (TITLE(1), AA(1))
      DIMENSION TITLE (5), AA(1)                                        T716
      DIMENSION TRS(101),CFS(101),ELS(101)
      DIMENSION HH(101)
      DIMENSION ZMD(40),ZS(40)
      DIMENSION RHOOW(101), Y(101)
      DIMENSION UEV(40), HEH(40), RHOER(40), EMDTT(40)
      DIMENSION TTEMP(4)
      DIMENSION INITX(3), IFINLX(3), UX(3), MX(3)
    2 FORMAT (2F12.3,2F12.6,3(4XE12.5))
    4 FORMAT(10X,E20.6)
      ITEST = K1
      IF(ITEST.EQ.2.AND.IBOD.EQ.0)K1=6
      IGO = K1
      GO TO (90,909,912,730,178,1171), IGO
   90 INTAP = 5
      IOTAP = 6
      REWIND 22
   99 XMASSO = 0.0
      EMDTO = 0.0
      K1=2
      CALL CHAIN (TWO)
C        IEND = 1 FOR THE FIRST OF MULTIPLE BOUNDARY LAYER CASES.
C               2 FOR BOUNDARY LAYER CASES FOLLOWING THE FIRST.
```

```
C
C         IEND = 3 FOR A SINGLE BOUNDARY LAYER CASE USING THE LARGE
C                CHEMICAL COMPOSITION PROGRAM.
C
C         IEND = 4 FOR THE FIRST OF MULTIPLE CHEMICAL COMPOSITION CASES.
C
  909 IF (IEND .EQ.4)  CALL CHAIN (EIGHT)
10101 CONTINUE
      K1=3
      CALL CHAIN (THREE)
  912 NWHICH = 1
      R=.896
      GO TO (730,710), IOPT
C       IOPT IS THE TRIGGER FOR THE EM AND EMDOT CALCULATIONS.
C         IF IOPT = 1, SINGLE VALUES OF EM,EMDOT ARE READ IN
C         IF IOPT = 2, EM AND EMDOT ARE OBTAINED FROM THE MASS LOSS
C            PROGRAM
C
C
  710 K1 =4
      DO 7101 II = 3,7
 7101 AM367(II) = TITLE(II-2)
      CALL CHAIN(FOUR)
  730 CIR = 2.* 3.1415926 * YB * RN /12.
 5000 K1=5
      CALL CHAIN (FIVE)
  178 CONTINUE
C
C     DETERMINE CLEAN AIR SHEAR STRESS FOR CASE OF ZERO LOCAL INJECTION
C     CALCULATIONS PERFORMED IN SEGMENT(FOUR)
C
C     FOR CASE WHERE SEGMENT(FOUR) HAS NOT BEEN PREVIOUSLY CALLED
C
C     TRIGGERED BY EMDOT = 0.0 AND IOPT = 1
C
      IF(EMDOT.NE.0.0.AND.IOPT.NE.1) GO TO 1171
      K1 = 6
      CALL CHAIN(FOUR)
 1171 CONTINUE
      NTRIG = 1
      JFIVE1=0
      IF(NABL1.EQ.0) GO TO 5
      CALL INPUT (KODE,TO,WABL1,WABL2,NABL1,NABL2,IREAD,1.0,1.0,1.
     *            RHOD,PTEMP,TTEMP,VRTO,NTOTAL,U)
    5 L=0
      JP = 0
```

```
      INITX(1) = 1
      INITX(2) = 86
      INITX(3) = 93
      IFINLX(1) = 81
      IFINLX(2) = 91
      IFINLX(3) =101
      MX(1) = 10
      MX(2) =  5
      MX(3) =  2
      UX(1) = 0.0
      UX(2) = 0.85
      UX(3) = 0.92
      ISSI = 0
      SWTCH = 1.0
      TWOGJ = 5.0132E4
      LIB=1
      MM = 10
      M=1
      P = PSTRM(1)
      UE = UEV(1)
      B = 1.5
C       INITIAL GUESS FOR FS
      IF (EMDOT .EQ. 0.0) B = 0.05
      IF (EM.EQ.0.0.AND.EMDOT.EQ.0.0) B=0.0
      FS=B/(1.0+B)
      DELFS=0.1*FS
      RHOW =(39. * P/2116.18) / TW
      IF(IBOD.EQ.0)
     *READ(5,101)UEP,HEP,RHOEP,PP,DELY,CIR
  101 FORMAT(6E12.0)
  179 L=L+1
      LEVIF = 0
      IF(L.LE.40) GO TO 7
      PRINT 6
    6 FORMAT(30X//*  BOUNDARY LAYER EDGE OUTSIDE SHOCK  *//)
      STOP
    7 CONTINUE
  181 IF(IBOD.EQ.0) GO TO 183
      UE = UEV(L)
      HE=HEH(L)
      RHOE=RHOEB(L)
      P = PSTRM(L)
      GO TO 1795
  183 UE = UEP
      HE = HEP
      RHOE = RHOEP
```

```
      P = PP
      EMDTT(L) = RHOE*UE*(DELY/12.)*(L-1)
 1795 DSTDEL = 0.0
      HAW = HE + 0.9 * (UE**2 / TWOGJ)
      VISC =  (1.18E-5*716.0)/(TW+216.0) * (TW/500.)**1.5
      PL = ALOG10(P/2116.18)
      CALL HANSEN (4,HE,PL,TE)
      CPA = HE / TE
      WA = 28.962
      XMASS=0.0
      MITER=0
  180 IF(B) 1180,1180,1181
 1180 CALL HANSEN (5,TW,PL,COMPR)
      RHOW = P /(53.35 * COMPR * TW)
      CALL HANSEN (3,TW,PL,HW)
      FS = 0.0
      GO TO 740
 1181 B = FS /(1.0-FS)
      PTEMP = P/2116.18
      RHOD = RHOW * 16.018432
      F=FS
      IF (L.NE.1 .AND. L.NE.20 .AND.NTRIG.EQ.1) GO TO 8025
      ICALC = 3
      U = 0.0
      CALL INPUT (KODE,TO,WABL1,WABL2,NABL1,NABL2,IREAD,1.0,F,ICALC,
     * RHOD,PTEMP,TTEMP,VRTO,NTOTAL,U)
      IREAD = 0
      TTEMP(1) = TW /1.8
      TTEMP(2) = PTEMP
      WRITE(6,9917) F,L,NTRIG
 9917 FORMAT (1H07X3HF =1PE15.7,5H  L =I3,9H  NTRIG =I2)
      RHOW = VRTO
      TTEMP(1) = TO
      IF (IEND.EQ.2) LIB = 2
      CALL CHEMS(TTEMP,80,RHOW,NTOTAL,HW,LIB,TREAL,JP)
      IF (LIB-5) 8025,8020,8025
 8020 HW= .24 * TW
      WRITE (6,8099) L,F,HW,RHOW
      JEIVEL=JEIVEL+1
      IF(JEIVEL.GT.3) STOP
 8099 FORMAT (1H05X48H** EQUILIBRIUM STATE WAS NOT REACHED IN CHEMS **/
     *1H 20X3HL =I4,5H  F =1PE13.5,5H  HW=E13.5,8H  RHOW =E13.5)
 8025 LIB=2
      IF (EMDOT .EQ. 0.0) GO TO 740
      TAU = UE*EMDOT/B
  740 WMIX = RHOW * TW * 1545. /P
```

```
      WB = FS/(1.0/WMIX- (1.-FS)/WA)
C      DENSITY RATIO
  200 CONTINUE
C       LOOP TO GENERATE ENTHALPY PROFILE
      U = 0.0
      DU = 0.01
      DO 770 I = 1,101
      HH(I) = HW + (HAW - HW) * U  - (HAW -HE) * U**2
  770 U = U + DU
C
      C5=VISC/12./SQRT(TAU*RHOW)
      C6=.4*UE/SQRT(TAU/RHOW)
C
      CPB = (HW /TW - (1.-FS)* CPA) /FS
      U=0.
      AM = M
      DU = .01 * AM
      DO 1000 K=1,3
      IF(K.GT.1 .AND. NTRIG.EQ.1) GO TO 251
      IF (K.GT.1 .AND. B.EQ.0.0) GO TO 251
      IF (B.EQ. 0.0) GO TO 1202
      IF (NTRIG.NE.1) GO TO 202
 1202 U = 0.0
      INIT = 1
      IFINAL = 101
      GO TO 203
  202 INIT = INITX(K)
      IFINAL = IFINLX(K)
      M = MX(K)
      U = UX(K)
  203 AM = M
      DU = .01* AM
      DO 250 I= INIT,IFINAL,M
      IF (B.EQ. 0.0) GO TO 1203
      F=FS*(1.0-U)
      IF (NTRIG.NE.1) GO TO 205
      RHOOW(I) = TW*WA*CPA/WMIX * (1.0+F *(CPB/CPA-1.0))
      RHOOW(I) = RHOOW(I) / (HH(I) * (1.0 + F * (WA/WB -1.0)))
      TRS(I) = TREAL
      GO TO 208
 1203 IF(I-1)1205,1204,1205
 1204 TRS(1)=TO
      RHOOW(1)=1.0
      GO TO 208
 1205 CALL HANSEN(4,HH(I),PL,TMP)
      TRS(I)=TMP/1.8
```

```
      CALL HANSEN (5,TMP,PL,COMPR)
      RHOOW(I) = P / (53.35 * COMPR * TMP) / RHOW
      GO TO 208
  205 RHOD = RHOW * 16.018432
      ICALC =2
      CALL INPUT(KODE,TO,WABL1,WABL2,NABL1,NABL2,IREAD,RHOOW(I),F,ICALC,
     * RHOD,PTEMP,TTEMP,VRTO,NTOTAL,U)
      TTEMP(3) = HH(I)
      WRITE (6,9918) F,L,NTRIG ,RHO,TTEMP(1) ,TTEMP(2), TTEMP(3)
 9918 FORMAT (1H 7X3HF =1PE15.7,5H  L =I3,9H  NTRIG =I2,4E15.7)
      LIB=2
      RHO = RHO * 16.018432 / 1.29233
      IF ( I.EQ.1) GO TO 1207
      IF(I-101) 206,207,207
  207 RHOOW(I) = RHOE/ RHOW
      TRS(I) = TE / 1.P
      GO TO 208
 1207 RHOOW(1) = 1.0
      TTEMP(I) = TO
      RHO = RHOW
      GO TO 208
  206 CALL CHEMS(TTEMP,90,RHO,NTOTAL,HJ,LIB,TREAL,JP)
      TRS(I) = TREAL
      TTEMP(I) = TREAL
      RHOOW(I) = RHO/RHOW
  208 IF(EMDOT) 1308,1308,1310
 1308 DENOM = 1.0
      GO TO 1312
 1310 DENOM = 1.0 + B*U
 1312 X = SQRT(RHOOW(I) / DENOM)
      ZX = U* RHOOW(I)
      Z = ZX * (1.0 - U)
      IF(I-1)220,210,220
  210  XINT=0.0
       GO TO 230
  220  XINT=XINT+DU*(X+XO)/2.0
  230   XO=X
       Y(I)=C5*EXP(C6*XINT)
       IF(I-1)240,235,240
  235  ZINT=0.0
       ZXINT = 0.0
       GO TO 245
  240  IM=I-M
       ZINT=ZINT+(Z+ZO)*(Y(I)-Y(IM ))/2.0
       ZXINT = ZXINT + (ZX+ZXO) * (Y(I)-Y(IM)) /2.0
  245    ZO=Z
```

```
      ZX0 = ZX
      U=U+DU
  250 CONTINUE
 1000 CONTINUE
  251 CONTINUE
      DSTDEL = 1.0 -ZXINT* RHOW /RHOE /Y(101)
      XMASS=CIR*RHOW*UE*ZXINT
      IF (B.EQ. 0.0) GO TO 290
      EMB = CIR*RHOW*UE*FS*ZINT
      WRITE (6,3751) EMB,EM
 3751 FORMAT (1H 6H EMB =1PE15.7,6H  EM =E15.7)
      IF(MITER)800,800,805
  800 EMB1=EMB
      FS1=FS
      IF (NTRIG.EQ.1) GO TO 801
      IF (EMB-EM) 801,290,802
  802 DELFS = - DELFS
  801 FS = FS + DELFS
      MITER=1
      GO TO 814
  805 DELEM=(EMB-EMB1)/DELFS
      DELFS=(EM-EMB)/DELEM
      XTEST=1.0-(EMB/EM)
      ERROR = 0.000001
      IF (NTRIG .EQ.2) ERROR = 0.0005
      IF (NTRIG .EQ. 2  .AND. EMDOT .EQ.0.0) ERROR = 0.002
      IF(ABS(XTEST)-ERROR    )290,290,810
  810 IF ((EMB/EMB1) .GT.100. .AND. (EMB/EM).GT. 10.0) DELFS=DELFS /2.0
      EMB1 = EMB
      FS1=FS
      FS=FS+DELFS
  814 IF (FS - 1.0) 817,815,815
  815 FS=(1.0+FS1)/2.0
      DELFS=FS-FS1
      LEVIF = LEVIF + 1
      IF(LEVIF.GT.10) GO TO 818
      GO TO 180
  817 IF(FS) 818,818,180
  818 FS=0.6
      DELFS=0.1*FS
      GO TO 179
  290  CONTINUE
  280 CONTINUE
      IF (NTRIG.EQ.2)  WRITE (6,5525) XMASS,FS,EMDTT(L)
 5525 FORMAT (1H010X7HXMASS =1PE13.5,5X4HFS =E13.5,5X10HEMDTT(L) =E13.5)
      IF(ISS1-1)350,349,349
```

```
350   EMDTO=EMDTT(L)
      XMASSO=XMASS
      UEO=UE
      ISSI=1
      GO TO 179
349   XMASS1=XMASS
      UE1=UE
      RHOE1=RHOE
      HE1=HE
      PE1 = P
3515  EMDT1=EMDTT(L)
      IF(SWTCH.EQ.0.0) GO TO 283
      IF (NTRIG .EQ. 2) GO TO 500
      WRITE(6,5525) XMASS,FS,EMDT1
      IF(EMDOT.EQ.0.0) GO TO 3516
      IF((XMASS-EM).LT.EMDT1) GO TO 283
3516  CONTINUE
      IF(((EMDTO .LT.(XMASSO-EM)).AND.(EMDT1 .GT.(XMASS1-EM))).OR.
     1((EMDTO.GT.(XMASSO-EM)).AND.(EMDT1.LT.(XMASS1-EM))))GO TO 283
      IF (L.LT.0 .OR. L.GE.40) STOP
      GO TO 355
500   IF(NREAL.EQ.0) GO TO 600
      IF(XMASS1-EM-EMDT1) 505,283,550
505   IF(LESS.EQ.2) GO TO 354
      L=L-2
      GO TO 355
550   IF(LESS.EQ.1) GO TO 354
      GO TO 355
600   NREAL=1
      IF(XMASS1-EM-EMDT1) 605,283,610
605   LESS=1
      L=L-2
      GO TO 355
610   LESS=2
355   EMDTO = EMDT1
      XMASSO=XMASS1
      UEO=UE
      HEO=HE
      RHOEO=RHOE
      PEO = P
      DELFS = 0.1 * FS
      GO TO 179
354   RHOE2 = RHOE
      IF(IBOD.EQ.0) GO TO 283
      HE2=HE
      PE2 = P
```

```
      UE2=UF
      EMDT2=EMDT1
      XMASS2=XMASS
      HEOSV =HE0
      PEOSV = PE0
      RHOOSV=RHOE0
      UF2SV=UF
      PE2SV = P
      UEOSV=UE0
      EMDTOS =EMDTO
      EMDT2S = EMDT1
      DHE =     (HE2 -HEOSV)
      DRHOE =     (RHOE2 - RHOOSV)
      DEMDT1 =     (EMDT2S - EMDTOS)
      DPE = (PE2 - PEOSV)
      DUE20=UE2SV-UEOSV
 3525 SWTCH=0.0
  357 CALL VALUES (EMDTO,EMDT2,UE0,UE2,A,B1)
      CALL VALUES (XMASSO-EM,XMASS2-EM,UE0,UE2,C,D)
      UE1 = (A-C) / (D-B1)
      DUE01=    (UE1-UEOSV)
      PRCNT= DUE01/DUE20
      HE =HEOSV + PRCNT *DHE
      RHOE = RHOOSV + PRCNT * DRHOE
      P = PEOSV + PRCNT * DPE
      EMDT1 = EMDWOS + PRCNT * DEMDT1
  359 UE =UF1
      GO TO 1795
  283 CONTINUE
      IF (NTRIG.EQ.2 .AND. B.NE. 0.0) DU = 0.1
      IF(NTRIG.EQ. 2 .AND. B.NE. 0.0) M= 10
      WRITE (IOTAP,8) TITLE
    8 FORMAT (1H130X24HTURBULENT BOUNDARY LAYER/1H 30X5A10)
      WRITE (IOTAP,9) TAU,RHOW,RHOE,WMIX,HW,EM,EMDOT,DSTDEL,XMASS,UE
    9 FORMAT(1H07X3HTAU14X4HRHOW12X4HRHOE12X4HWMIX12X2HHW/
     *5(4X1PE12.5)//
     *8X2HEM14X5HEMDOT11X6HDSTDEL10X5HXMASS11X2HUE/
     *5(4X1PE12.5)//
     *6X1HU10X5H HH  7X5HRHOOW0X1HF12X1HY15X2HYD14X1HT)
      RHOD = RHOW * 16.018432
      U=0.0
      DO 300 I= 1,51,M
      YD=Y(I)/Y(101)
      F = FS * (1.-U)
      WRITE (IOTAP,2) U,HH(I),RHOOW(I),F,Y(I),YD,TRS(I)
      U = U + DU
```

```
  300  CONTINUE
       WRITE (IOTAP,8) TITLE
       KK = 51 + M
       DO 3300 I= KK,101,M
       YD = Y(I)/Y(101)
       F = FS * (1.-U)
       WRITE (IOTAP,2) U,HH(I),RHOOW(I),F,Y(I),YD,TRS(I)
       IF (I.EQ.101) GO TO 3300
       U = U + DU
 3300  CONTINUE
       ICALC = 2
       IF (B.EQ. 0.0) GO TO 284
       IF(NTRIG-2) 295,284,284
  295 NTRIG=2
       M=MM
       NREAL = 0
       DELFS = 0.1 * FS
       GO TO 1795
  284 CONTINUE
       WRITE (6,988) FS,HW,HAW,HE,RHOW,C5,C6,UE,B,CIR
       PATM = P/2116.
       IF(IEND.LT.3)
      *PUNCH 3000, TITLE
 3000 FORMAT(5A10)
       IF(IEND.LT.3)
      *PUNCH3001,FS,HW,HAW,HE,RHOW,C5,C6,UE,B,CIR,TO,PATM
 3001 FORMAT(10X,4E15.5)
       IF(IEND.LT.3)
      *PUNCH3002,KODE,NABL1,NABL2,KODFW,WABL1,WABL2
 3002 FORMAT(13X,I2,3I5/10X,2E15.5)
C
       IF ( B.EQ. 0.0) GO TO 2900
C
       RHW = RHOW
       JP = 1
       U = 0.0
       DO 2800 I=1,100,M
       F = FS * (1.0 - U)
       ICALC = 3
       CALL INPUT(KODE,TO,WABL1,WABL2,NABL1,NABL2,IREAD,RHOOW(I),F,ICALC,
      * RHOD,PTEMP,TTEMP,VRTO,NTOTAL,U)
       RHO = RHOOW(I) * RHOW * 16.018432 / 1.29233
       TTEMP(1) = TRS(I)
       TTEMP(3) = HH(I)
       CALL CHEMS(TTEMP,00,RHO,NTOTAL,HJ,LIB,TREAL,JP)
 2800 U = U + DU
```

```
C
 2810 CONTINUE
      IF (IEND.LT.3) GO TO 99
      DU=0.02
      U = 0.0
      ICALC =4
      TTEMP(1) = TO
      TTEMP(2) = PTEMP
      TTEMP(4) = 0.0
      DO 680 I=1,100,2
      F = FS * (1.0 -U)
      TTEMP(3) = HH(I)
      IF (I-1) 655,655,660
  655 IREAD = 1
      GO TO 670
  660 IF (I-101) 665,680,680
  665 IREAD =0
C        THIS CALL TO INPUT WILL WRITE THE INPUT TAPE FOR DECK T666.
  670 CALL INPUT (KODE,TO,WABL1,WABL2,NABL1,NABL2,IREAD,1.0,F,ICALC,
     * RHOD,PTEMP,TTEMP,VRTO,NTOTAL,U)
      U = U + DU
  680 CONTINUE
      ENDFILE 22
      REWIND 22
      RHW = RHOW
      HZ = HW
      CALL CHAIN (SIX)
  988 FORMAT (1H0/1H010X4HFS =E16.8,5X4HHW =E16.8,5X5HHAW =E16.8/
     *1H010X4HHE =E16.8,5X6HRHOW =E16.8,5X4HCP =E16.8/
     *1H010X4HC6 =E16.8,5X4HUE =E16.8,5X3HB =E16.8/
     *1H010X5HCIR =E16.8)
 2900 CONTINUE
      JTR = 80
      TTEMP(2) = P /2116.18
      TTEMP(4) = 0.0
      DENS = RHOD / 1.29233
      REWIND 22
      DO 2920 I=1,101,2
      TTEMP(3) = HH(I)
      TTEMP(1) = TRS(I)
      WRITE (22,4192) JTR,DENS,TTEMP
 4192 FORMAT (I3, 5E12.5)
 2920 CONTINUE
      RHW = RHOW
      HZ = HW
      END FILE 22
```

```
      REWIND 22
      CALL CHAIN (NINE)
      END
```

d. Program Variables

Symbolic representation	Meaning or use
A	Function value used in iteration
AA	First location of common block 716
AM	Floating point M (see M) used to calculate DU
AM367	Common array for Segment FOUR
B	Function value used in iteration
C	Function value used in iteration
CIR	Circumference
CPA	Specific heat of air
CPB	Specific heat of injected material
C5	Exponent used to calculate Y array
C6	Exponent used to calculate Y array
D	Function value used in iteration
DELEM	Delta EM
DELFS	Delta FS used in iteration
DELY	Delta Y in flat plate option (normal distance increment)
DEMDTI	Delta M used in iteration
DHE	Delta HE used in iteration
DPE	Delta PE used in iteration
DRHOE	Delta RHOE used in iteration
DSTDEL	Displacement thickness divided by boundary layer thickness
DU	Increment used for U in DO LOOPS
DUE01	Delta U used in iteration
DUE20	Delta U used in iteration
EM	Mass loss from heating routine

EMB	Calculated EM from conservation of mass
EMB1	Saved EM
EMDOT	Mass loss rate from Segment FOUR
EMDTO	Variable EMDOT, set to EMDTT(L)
EMDTOS	Saved EMDOT
EMDTT	Mass flow profile in the inviscid flow region
EMDT1	Mass flow used in iteration
EMDT2	Mass flow used in iteration
EMDT2S	Saved mass flow
F	Mass fraction
FS	Mass fraction at wall
FS1	Saved mass fraction
HAW	Adiabatic wall enthalpy
HE	Variable enthalpy set to HEH(L)
HEH	Enthalpy profile from the streamtube calculation
HEO	Enthalpy used in iteration
HEOSV	Enthalpy used in iteration
HE1	Enthalpy used in iteration
HE2	Enthalpy used in iteration
HEP	Edge enthalpy, flat plate option
HH	Calculated enthalpy array
HJ	Enthalpy returned from subroutine CHEMS
HW	Wall enthalpy
I	DO LOOP index used throughout the program
ICALC	Trigger for calculations in subroutine INPUT
IEND	Trigger used for putting an end of file on input tape for Segment SIX

IGO	Program segment control
INTAP	Symbolic tape number for system input tape
IOPT	Trigger for EM and EMDOT calculations
IM	Index used in integral calculation
IOTAP	Symbolic tape number for system output tape
IREAD	Trigger for reading cards in subroutine INPUT
ISSI	Iteration control trigger
KK	Index used in DO LOOPS
KODE	Number of compounds in air model (2 or 3)
K1	Segment indicator
L	Profile index
LESS	Indicator for real gas iteration
LIB	Indicator for reading library data in subroutine CHEMS; 1 if data is to be read, 2 if not
M	DO LOOP increment
MITER	Trigger used in iteration scheme
MM	DO LOOP increment for real gas solutions
NABL1	Number of compounds in first ablator
NABL2	Number of compounds in second ablator
NREAL	Indicator for direction of interpolation in real gas solution
NTOTAL	Total number of species for subroutine CHEMS
NTRIG	Trigger for ideal or real gas calculations; 1 for ideal, 2 for real
NWHICH	Iteration trigger
P	Variable pressure, set to PSTRM(L)
PP	Edge pressure, flat plate option

PEO	Pressure used in iteration
PEOSV	Pressure used in iteration
PE1	Pressure used in iteration
PE2	Pressure used in iteration
PE2SV	Pressure used in iteration
PRCNT	Percentage test value used in iteration
PSTRM	Pressure profile from streamtube calculation
PTEMP	Variable pressure
QQ	Not used
QS	Heating rate
R	Constant: 0.896
RHO	Density
RHOD	Density
RHOE	Variable density, set to RHOER(L)
RHOEO	Density used in iteration
RHOET	Edge density, flat plate option
RHOER	Density profile from streamtube calculator
RHOW1	Density used in iteration
RHOE2	Density used in iteration
RHOOSV	Saved density
RHOOW	Array of ratio of density at a point to density at the wall
RHOW	Density ratio at wall
RN	Nose radius
SCT	Common array for subroutine DISCOT
SS	Shock angle
SSS	Common array for subroutine CHEMS

SSSS	Common Array for subroutines CHEMS and INPUT
SWTCH	Trigger used for printing profiles after convergence of ideal gas properties
TAU	Shear stress
TITLE	Title for case
TO	Temperature at wall
TREAL	Temperature calculated in subroutine CHEMS
TTEMP	Array of input values for subroutine CHEMS
TW	Temperature at wall
TWOGJ	Constant: 50132
U	Velocity
UE	Variable velocity, set to UEV(L)
UEO	Velocity used in iteration
UEP	Edge velocity, flat plate option
UEOSV	Saved velocity
UEV	Velocity profile from the streamtube calculation
UE1	Velocity used in iteration
UE2	Velocity used in iteration
UE2SV	Saved velocity
VISC	Viscosity
VRTO	Density ratio used in subroutine CHEMS
WA	Molecular weight of air
EMAB1	Array of molecular weights of materials in first ablator
EMAB2	Array of molecular weights of materials in second ablator
WB	Molecular weight of the injected material

WMIX	Molecular weight of the gas mixture at wall
X	Point values for integral
XINT	Value of the integral
XMASS	Total gas flow through the boundary layer
XMASSO	Gas flow used in iteration
XMASS1	Gas flow used in iteration
XMASS2	Gas flow used in iteration
XO	Variable used in integral DO LOOPS
XTEST	Value for iteration test
Y	Distance through boundary layer, normal to wall
YB	Body radius (YBB/nose radius)
YD	Local value of Y divided by boundary layer thickness
Z	Point values for integral
ZINT	Value of integral
ZO	Variable used in integral DO LOOPS
ZS	Not used in main program

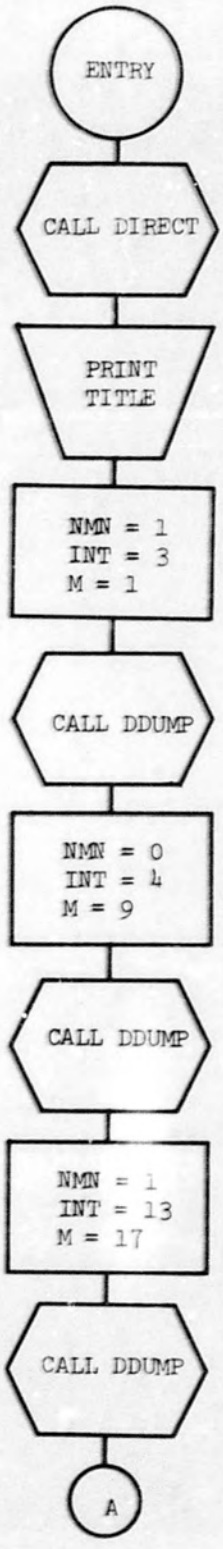
5. SEGMENT TWO

a. Function

Segment TWO is a short section of the program, and it performs two operations. The symbolic case data is read into the computer by the first operation, and the atmospheric properties are determined in the second operation. Subroutine DIRECT calls subroutine GINPUT which actually reads the data and places the data in the proper common location. The actual mechanics of subroutine GINPUT are discussed in Section IV. After the data is read into the computer, the program heading is printed and the input data is listed for ready reference. Three successive calls to subroutine DDUMP are made to list this data. These calls list respectively the BCD, fixed point, and floating point data.

Subroutine ARDC59 is used to calculate the local atmospheric properties at the altitude of interest. These properties include pressure, temperature, density, viscosity, and sound speed. The free stream Mach number and the vehicle wall temperature are also calculated in this segment before control is returned to Segment ONE.

b. Flow Chart

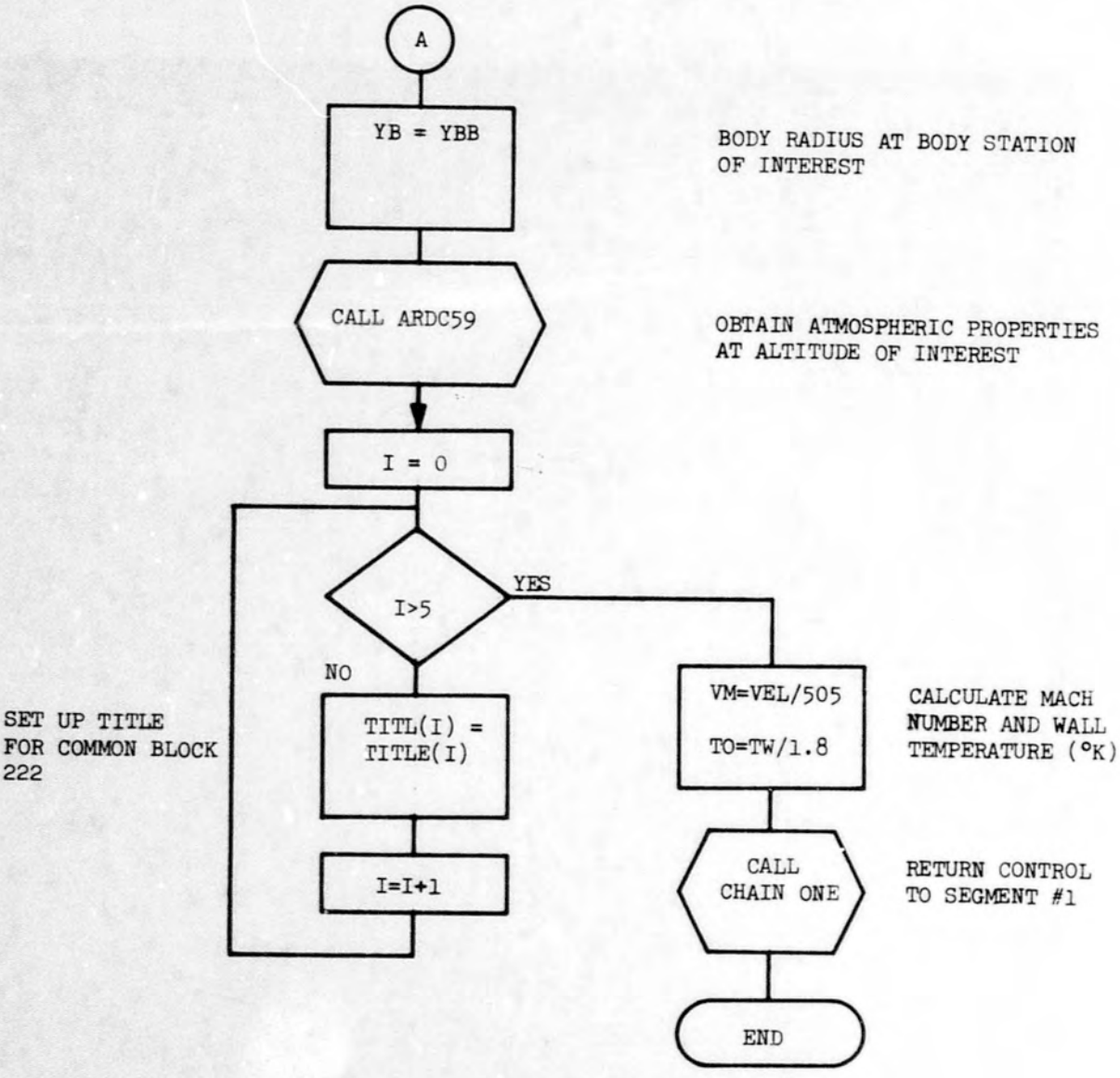


SYMBOLIC CASE DATA IS READ UTILIZING DIRECT, AND GINPUT

PRINTS BCD DATA FROM SYMBOLIC INPUT

PRINTS FIXED POINT DATA FROM SYMBOLIC INPUT

PRINTS FLOATING POINT DATA FROM SYMBOLIC INPUT


A
YB = YBB
BODY RADIUS AT BODY STATION OF INTEREST
CALL ARDC59
OBTAIN ATMOSPHERIC PROPERTIES AT ALTITUDE OF INTEREST
I = 0
I>5
YES
NO
SET UP TITLE FOR COMMON BLOCK 222
TITL(I) = TITLE(I)
I=I+1
VM=VEL/505
TO=TW/1.8
CALCULATE MACH NUMBER AND WALL TEMPERATURE (°K)
CALL CHAIN ONE
RETURN CONTROL TO SEGMENT #1
END

c. Program Variables

The list below gives the variables actually used in this segment. Other variables appear in the common statements but are not used; hence, they are defined elsewhere.

Symbolic representation	Meaning or use
AA	First location of common block 716
ALT	Altitude
BCD	Array of input variable names
D	Dummy
INT	Number of variables indicator for subroutine DDUMP
IR	Array of input variable dimensions
M	Location indicator for DDUMP
NGR	BCD name used for comparison
NGRA	Variable set up in DATA statement to WHCASE
NMN	Type of variable indicator for subroutine DDUMP
P1	Pressure
RHO1	Density
SOS	Sound speed
TATMOS	Temperature
TITLE	Title
TO	Temperature at wall (°K)
TW	Temperature at wall (°R)
VEL	Velocity
YB	Body radius
YBB	Read in body radius

D. LISTING

```
      SEGMENT TWO(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
      COMMON /234/ SSSS(3960)
      COMMON /333/ JFS(150)
      COMMON /367/ AM367(150)
      COMMON / 456/ INTAP,IOTAP,INT,NMN,M,NGR
      COMMON /666/ SSS(292)
      COMMON /716/   TITLE,  IOPT,  IBOD,  IEND, NABL1, NABL2, IREAD,
     1  KODE,IHEMIS, KODEW,
     2  ANTL,    RN,   ALT,   VEL, XINCH,    Y1,  SMXB,  XMXB, DYDX1,
     3  CP10,    TW,    EM, EMDOT,    WA, WABL1, WABL2,   YBB,
     4    K1,     N,    P1,  RHO1,   ZMD,    ZS,     P,    YB,    VM,
     5   UEV,   HEH, RHOER, EMDTT,    TO,  RHOW,    HW,XMASSO, EMDTO,
     6  VRTO,   II8,   II9,     R,   TAU,    CI ,NTOTAL,NWHICH,  IJJJ,
     7     L
      COMMON /567/ PS,THS,PSTRM(40)
      COMMON /1/ SOS,TATMOS,RHO2
      COMMON /222/ Y,FS,RHOOW,TRS,CFS,ELS,HZ,HAW,HE,RHW,TITL(5),
     * C5,C6,UE,B,CIR
      DIMENSION TITLE (5), AA(1)
      DIMENSION TRS(101),CFS(101),ELS(101)
      EQUIVALENCE (AA(1), TITLE(1))
      DIMENSION HH(101)
      DIMENSION ZMD(40),ZS(40)
      DIMENSION RHOOW(101), Y(101)
      DIMENSION UEV(40), HEH(40), RHOER(40), EMDTT(40)
      DIMENSION BCD(30), IR(28)
      DATA BCD /4HT716.27., 5HTITLE,  4HIOPT, 4HIBOD,  4HIEND,
     1     5HNABL1, 5HNABL2, 5HIREAD,  4HKODE,6HIHEMIS, 5HKODEW,
     2      4HANTL,    2HRN,   3HALT,   3HVEL, 5HXINCH,    2HY1,
     3      4HSMXB,  4HXMXB, 5HDYDX1,  4HCP10,    2HTW,    2HEM,
     4     5HEMDOT,    2HWA, 5HWABL1, 5HWABL2,   3HYBB/
      DATA IR /5,26*1/
      DATA NGRA /6HWHCASE/
      NGR = NGRA
C     VARIABLE          INDEX
C      TITLE             1
C      IOPT              6
C      IBOD              7
C      IEND              8
C      NABL1             9
C      NABL2            10
C      IREAD            11
```

```
C       KODE                12
C       IHEMIS              13
C       KODEW               14
C       ANTL                15
C       RN                  16
C       ALT                 17
C       VEL                 18
C       XINCH               19
C       Y1                  20
C       SMXB                21
C       XMXB                22
C       DYDX1               23
C       CP10                24
C       TW                  25
C       EM                  26
C       EMDOT               27
C       WA                  28
C       WABL1               29
C       WABL2               30
C       YBB                 31
      CALL DIRECT (AA,BCD)
      WRITE (6,1000) TITLE
 1000 FORMAT (1H1,30X,24HTURBULENT BOUNDARY LAYER/1H0,30X,5A10)
      NMN=-1
      INT = 3
      M=1
      CALL DDUMP(AA,BCD,IR)
      NMN=0
      INT=4
      M = 9
      CALL DDUMP(AA,BCD,IR)
      NMN=1
      INT = 13
      M = 17
      CALL DDUMP(AA,BCD,IR)
      YB=YBB
      CALL ARDC59 (ALT,RHO1,SOS,TATMOS,P1,D)
      DO 500 I=1,5
  500 TITL(I) = TITLE(I)
      VM=VEL/SOS
      TO = TW / 1.8
      CALL CHAIN(ONE)
      END
```

6. SEGMENT THREE

a. Function

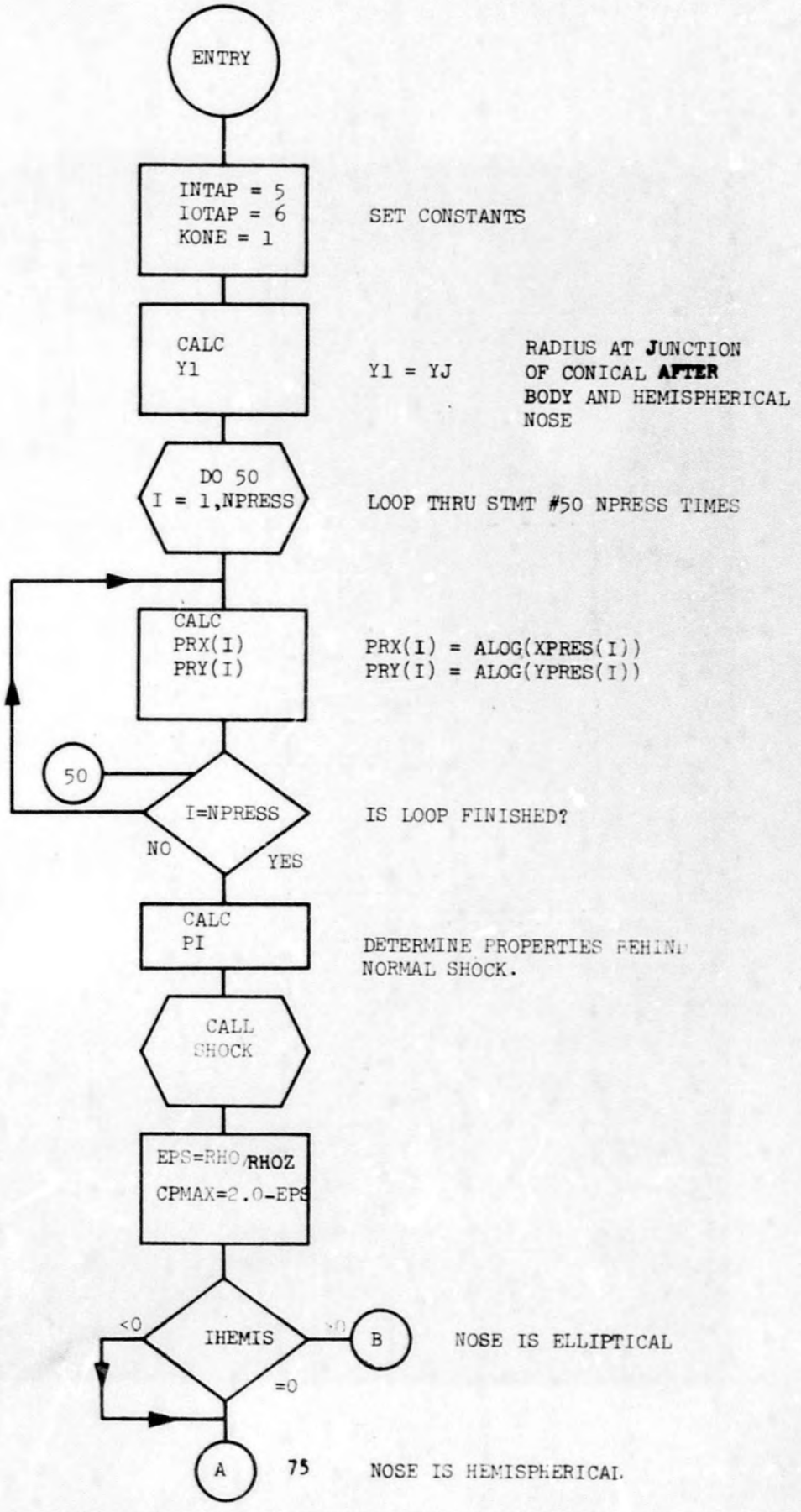
The purpose of this segment is to calculate the shape of the shock wave surrounding a blunt cone and to calculate the flow properties behind the oblique shock at the body station of interest. As outlined in the theory, three correlations are used in this calculation. Each theory is used over a certain portion of the body and the solutions are matched at the end points of each portion.

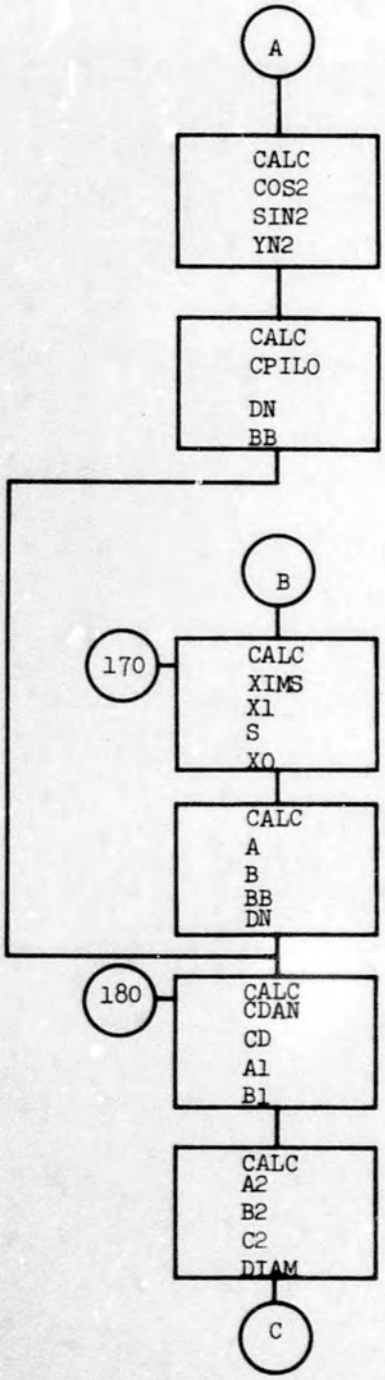
The first step in the shock shape calculation is the calculation of the flow properties behind a normal shock at the given flight Mach number. These properties are then used to calculate the drag coefficient-area product for either a hemispherical or elliptical nose. The drag coefficient to be used in the shock shape correlation calculations is determined next. The body pressure is also required. This parameter is calculated from data tabulated by Wagner and Watson (Ref. 10) and by Hayes and Probstein (Ref. 11). The normal shock properties are also used to calculate the parameter in the Mirels-Thornton shock shape correlation which is a function of the ratio of specific heats behind a normal shock.

To determine the shock angle of the oblique shock, it is necessary to obtain the proper free stream ratio of specific heats as a function of altitude and velocity. The coefficients for this calculation are determined in subroutine GAMCOF. This value of ratio of specific heats is then used to calculate the proper shock angle for the conical portion of the shock. This shock angle is used to determine when the shock, as predicted by Cheng's correlation, has reached conical form.

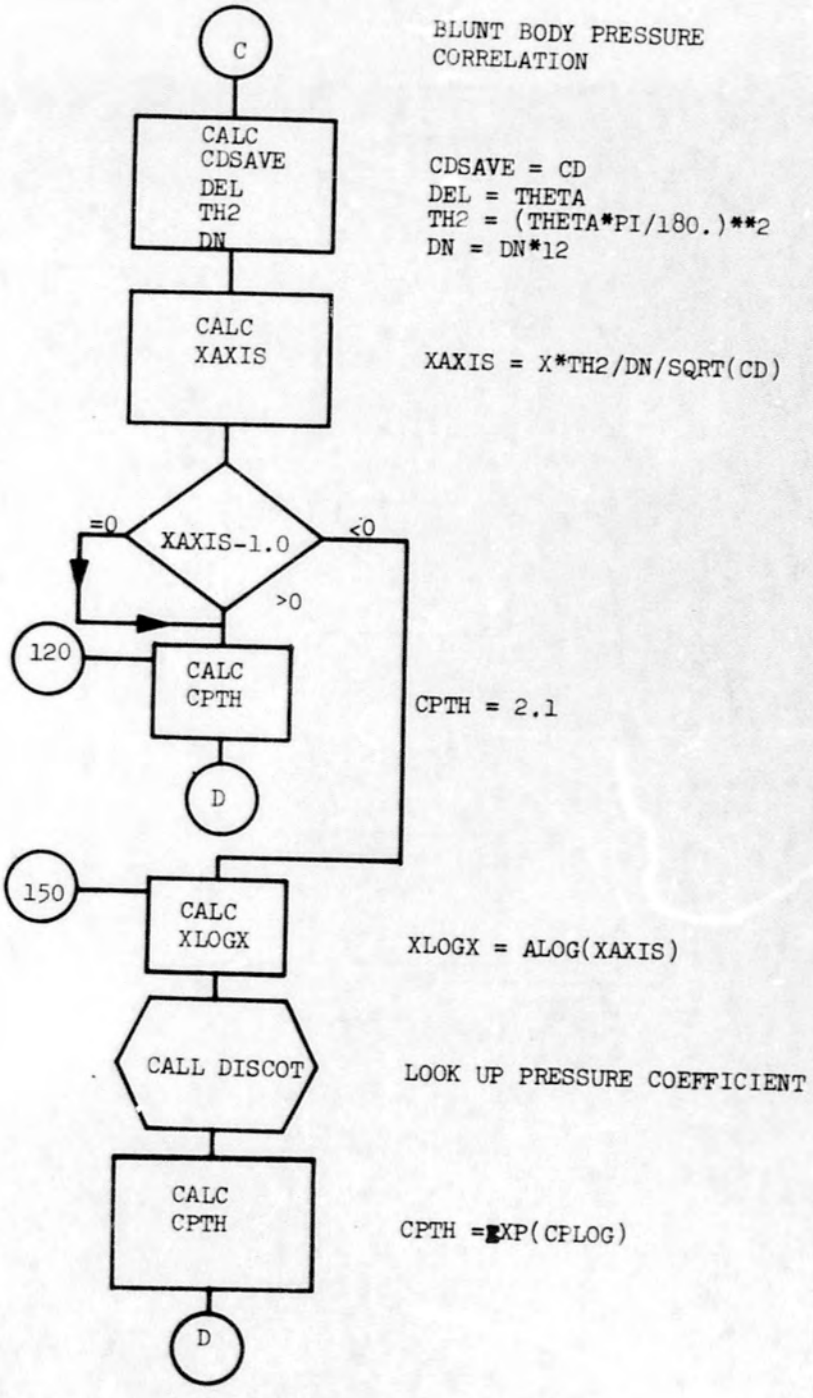
The next series of computations is to determine the shock shape around the nose. As stated in the theory section, the Mirels-Thornton method is used between $0 \leq \bar{X} \leq 0.1$. The Cheng method is used to join the Mirels-Thornton results to the conical shock. The shock shape and other parameters of interest are printed near the end of the segment; then control is returned to Segment ONE. The values of shock radius and angle are saved for use in Segment FIVE, which determines the inviscid flow properties of the air surrounding the body.

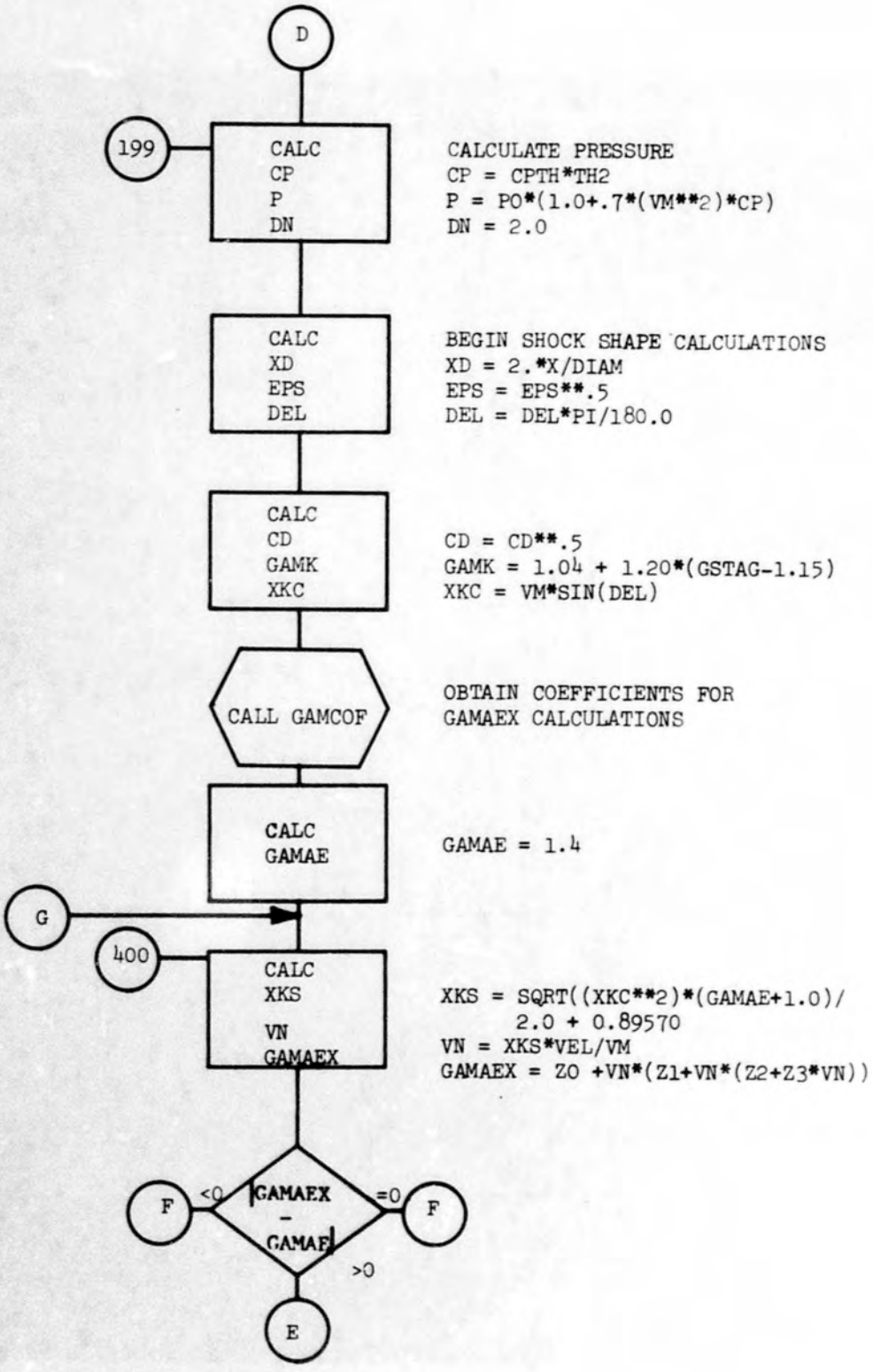
b. Flow Chart

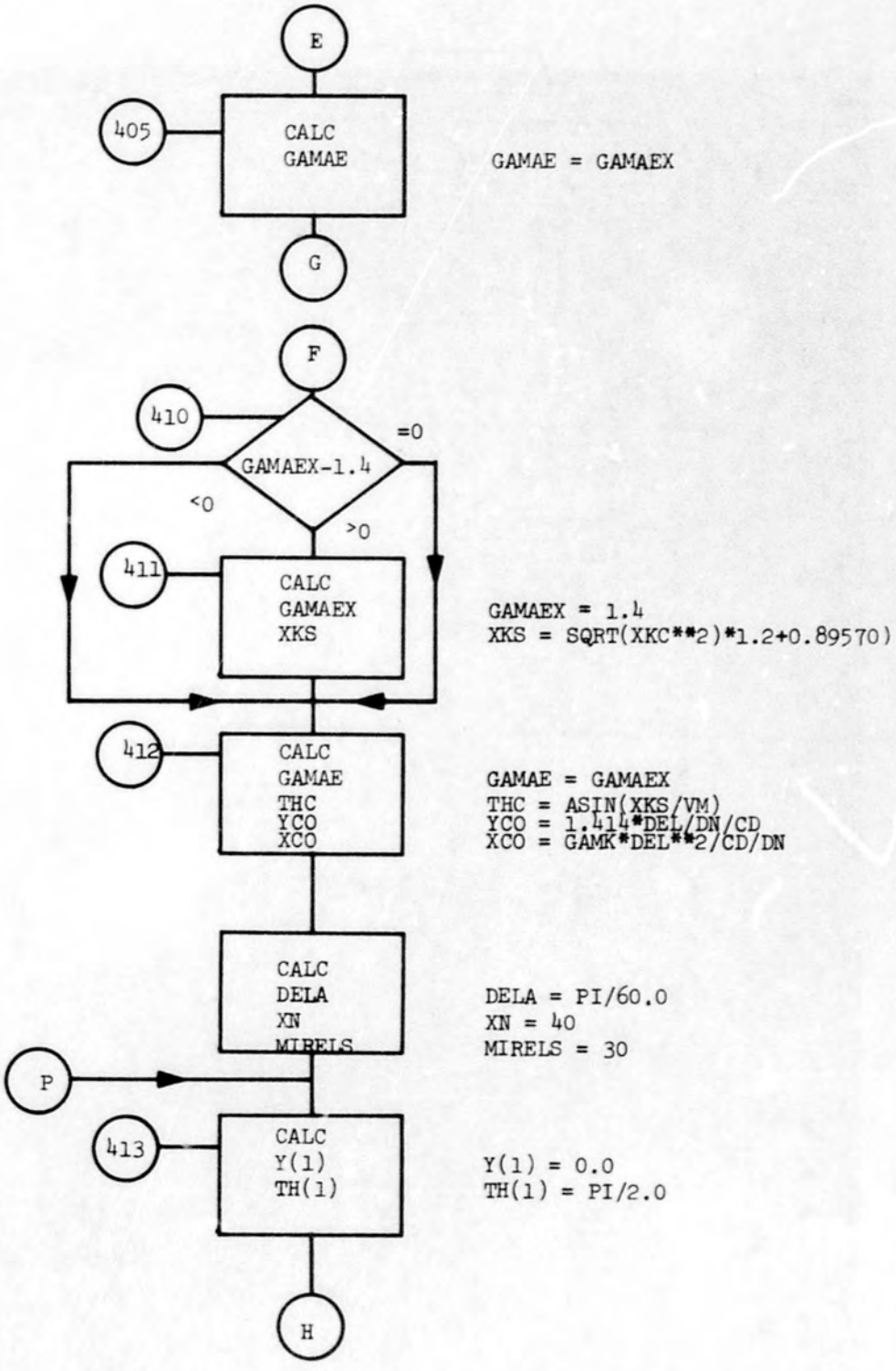

ENTRY
INTAP = 5
IOTAP = 6
KONE = 1
SET CONSTANTS
CALC
Y1
Y1 = YJ
RADIUS AT JUNCTION OF CONICAL AFTER BODY AND HEMISPHERICAL NOSE
DO 50
I = 1,NPRESS
LOOP THRU STMT #50 NPRESS TIMES
CALC
PRX(I)
PRY(I)
PRX(I) = ALOG(XPRES(I))
PRY(I) = ALOG(YPRES(I))
50
I=NPRESS
NO
YES
IS LOOP FINISHED?
CALC
PI
DETERMINE PROPERTIES BEHIND NORMAL SHOCK.
CALL
SHOCK
EPS=RHO/RHOZ
CPMAX=2.0-EPS
IHEMIS
<0
>0
=0
B
NOSE IS ELLIPTICAL
A
NOSE IS HEMISPHERICAL

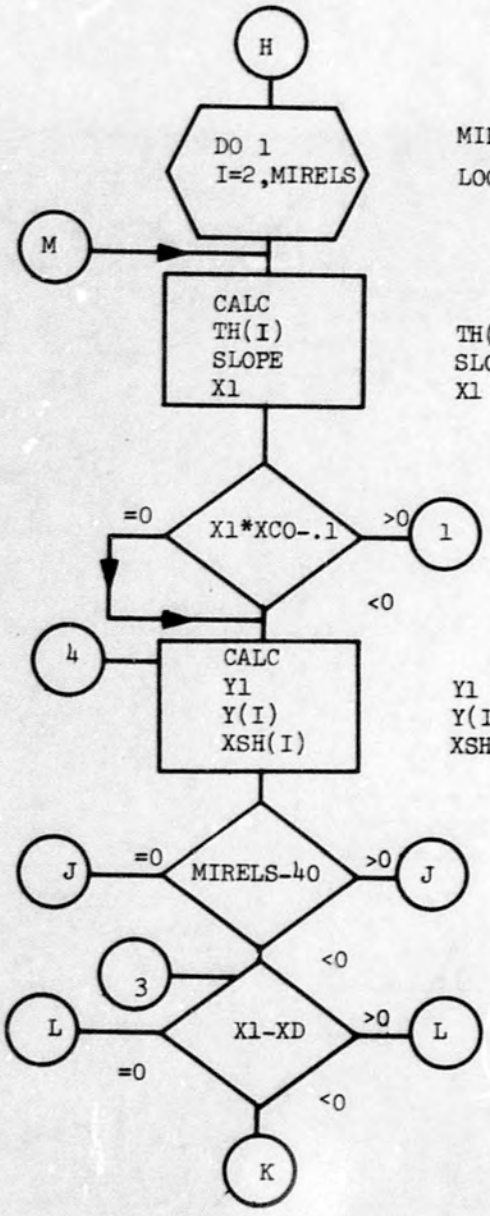



```
NOSE DRAG FOR HEMISPHERICAL
NOSE
COS2 = (COS(THETA*PI/180.0))**2
SIN2 = (SIN(THETA*PI/180.0))**2
YN2 = (RN*COS(THETA*PI/180.0)112.0)**2


CPIO = PI*YN2*
       (1.0-COS2/2.0-2.0*SIN2/0.940/
DN = RN/6.0                    CPMAX)
BB = 1.0


NOSE IS ELLIPTICAL, CURVE
FIT SHAPE


XIMS = XMXB-SMXB
X1 = Y1/DYDX1
S = X1-XIMS
XO = S**2/(2.0*S-X1)


A = XO-S
B = A*Y1/(A**2-(XO-X1)**2)**.5
BB = (B/A)**2
DN = B/6.

NOSE DRAG FOR CORRELATION
CALCULATIONS
CDAN = .940*CPMAX*CP10
CD = .2732395*CDAN/DN**2
A1 = .475 + .385/DD          CONSTANTS
B1 = .235 + .340/(BB+1.)     FOR M-T.
                             CORRELATION


A2 = .24389
B2 = .659056          CONSTANTS FOR
C2 = .20412           CHENG
DIAM = DN*12          CORRELATION
```


C
BLUNT BODY PRESSURE CORRELATION
CALC CDSAVE DEL TH2 DN
CDSAVE = CD
DEL = THETA
TH2 = (THETA*PI/180.)**2
DN = DN*12
CALC XAXIS
XAXIS = X*TH2/DN/SQRT(CD)
XAXIS-1.0
=0
<0
>0
120
CALC CPTH
CPTH = 2.1
D
150
CALC XLOGX
XLOGX = ALOG(XAXIS)
CALL DISCOT
LOOK UP PRESSURE COEFFICIENT
CALC CPTH
CPTH = EXP(CPLOG)
D


D
199
CALC
CP
P
DN
CALCULATE PRESSURE
CP = CPTH*TH2
P = PO*(1.0+.7*(VM**2)*CP)
DN = 2.0
CALC
XD
EPS
DEL
BEGIN SHOCK SHAPE CALCULATIONS
XD = 2.*X/DIAM
EPS = EPS**.5
DEL = DEL*PI/180.0
CALC
CD
GAMK
XKC
CD = CD**.5
GAMK = 1.04 + 1.20*(GSTAG-1.15)
XKC = VM*SIN(DEL)
CALL GAMCOF
OBTAIN COEFFICIENTS FOR
GAMAEX CALCULATIONS
CALC
GAMAE
GAMAE = 1.4
G
400
CALC
XKS
VN
GAMAEX
XKS = SQRT((XKC**2)*(GAMAE+1.0)/
2.0 + 0.89570
VN = XKS*VEL/VM
GAMAEX = ZO +VN*(Z1+VN*(Z2+Z3*VN))
F
<0
GAMAEX
-
GAMAE
=0
F
>0
E


E
405
CALC
GAMAE
GAMAE = GAMAEX
G
F
410
GAMAEX-1.4
=0
<0
>0
411
CALC
GAMAEX
XKS
GAMAEX = 1.4
XKS = SQRT(XKC**2)*1.2+0.89570)
412
CALC
GAMAE
THC
YCO
XCO
GAMAE = GAMAEX
THC = ASIN(XKS/VM)
YCO = 1.414*DEL/DN/CD
XCO = GAMK*DEL**2/CD/DN
CALC
DELA
XN
MIRELS
DELA = PI/60.0
XN = 40
MIRELS = 30
P
413
CALC
Y(1)
TH(1)
Y(1) = 0.0
TH(1) = PI/2.0
H

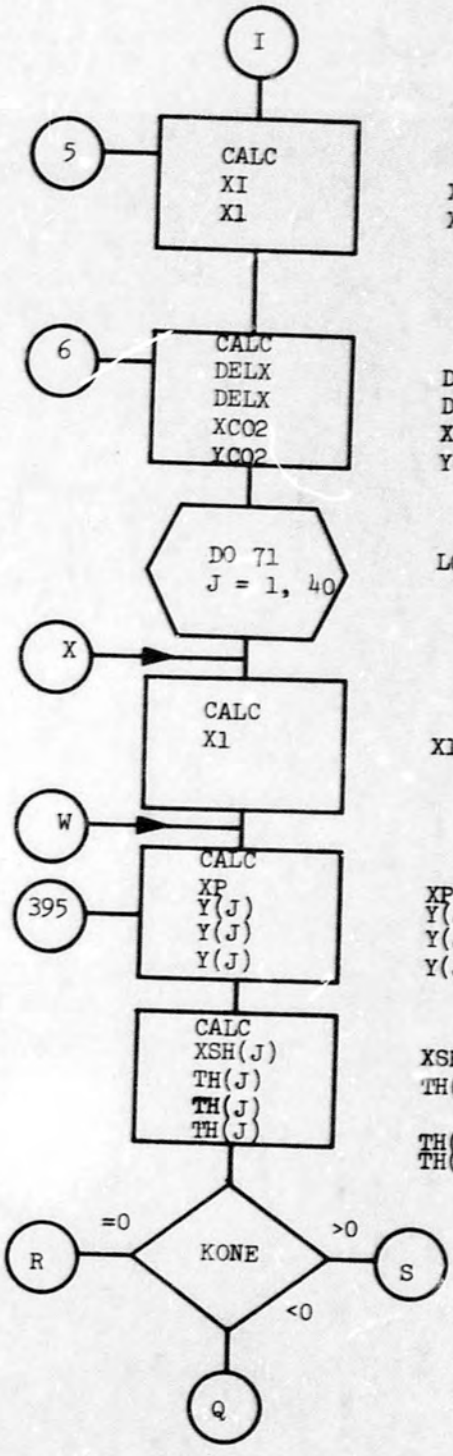


MIRELS-THORTON SHOCK SHAPE

LOOP THRU STMT #1 MIRELS-1 TIMES

```
TH(I) = TH(I-1)-DELA
SLOPE = SIN(TH(I))/COS(TH(I))
X1 = (YCO*SLOPE/A1/B1/XCO**B1)**
     (1./(B1-1))
```

```
Y1 = A1*(XCO*X1)**B1/YCO
Y(I) = Y1
XSH(I) = X1
```



CHENG SHOCK SHAPE

XI = I-1
X1 = XSH(I-1)

DELX = (XD-X1)/(XN-XI)
DELX = ABS(DELX)
XC02 = XC0/EPS/GAMK
YC02 = YC0/EPS/1.414

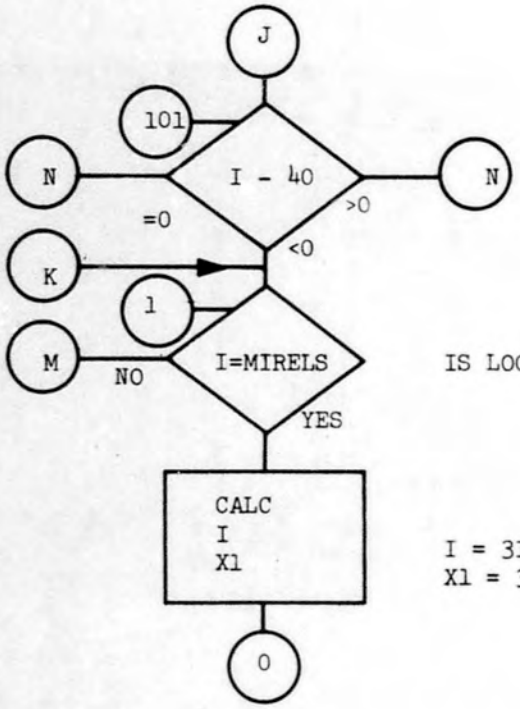
LOOP THRU STMT #71 40 TIMES

X1 = X1 + DELX

XP = ALOG10(X1*XC02)
Y(J) = A2*XP**2 + B2*XP + C2
Y(J) = 10.**Y(J)
Y(J) = Y(J)/YC02

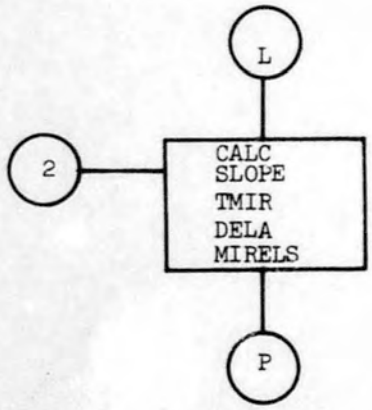
XSH(J) = X1
TH(J) = Y(J)/X1*(2.0*A2*ALOG10
(X1*XC02)
TH(J) = TH(J) + Y(J)*B2/X1
TH(J) = ATAN(TH(J))

HAVE CONICAL PROPERTIES
BEEN REACHED?

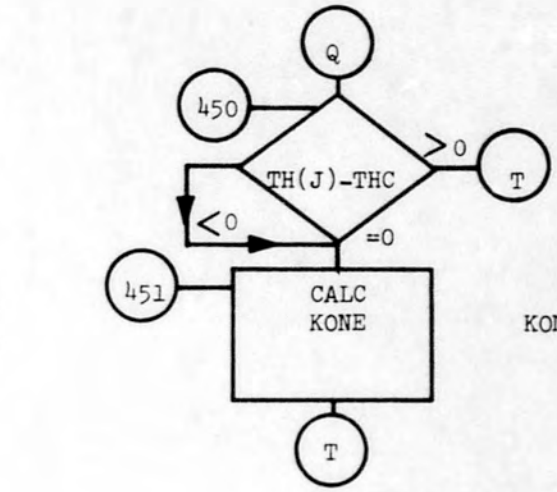


IS LOOP FINISHED?

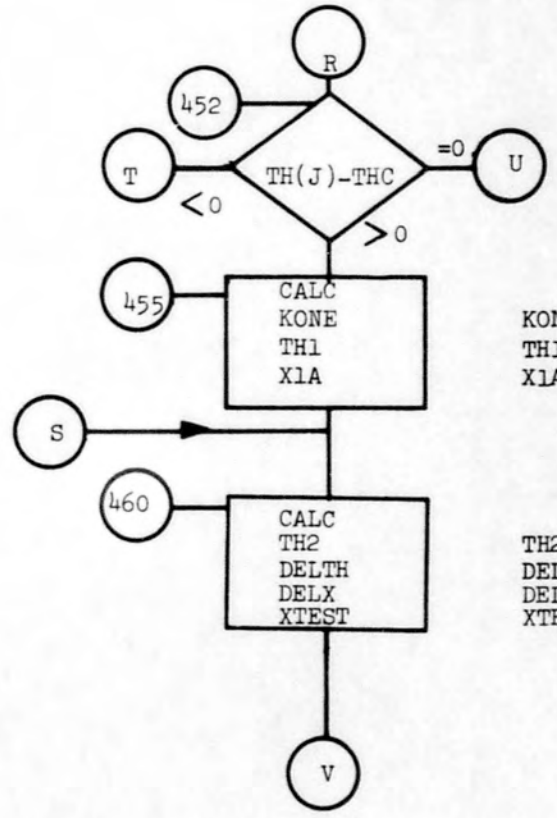
I = 31
X1 = 30.0



SLOPE = A1*R1*(XCO**B1)*(XD**(B1-1))/YCO
TMIR = ATAN(SLOPE)
DELA = (PI/2.0-TMIR)/39.0
MIRELS = 40



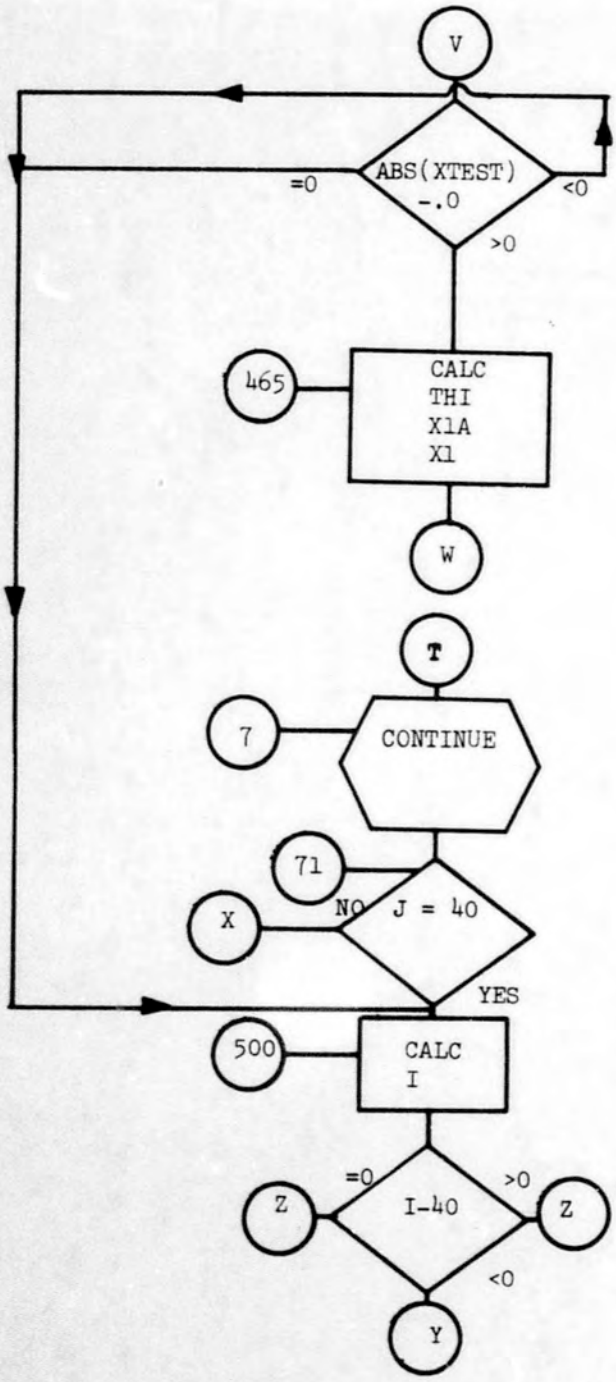
KONE = 0

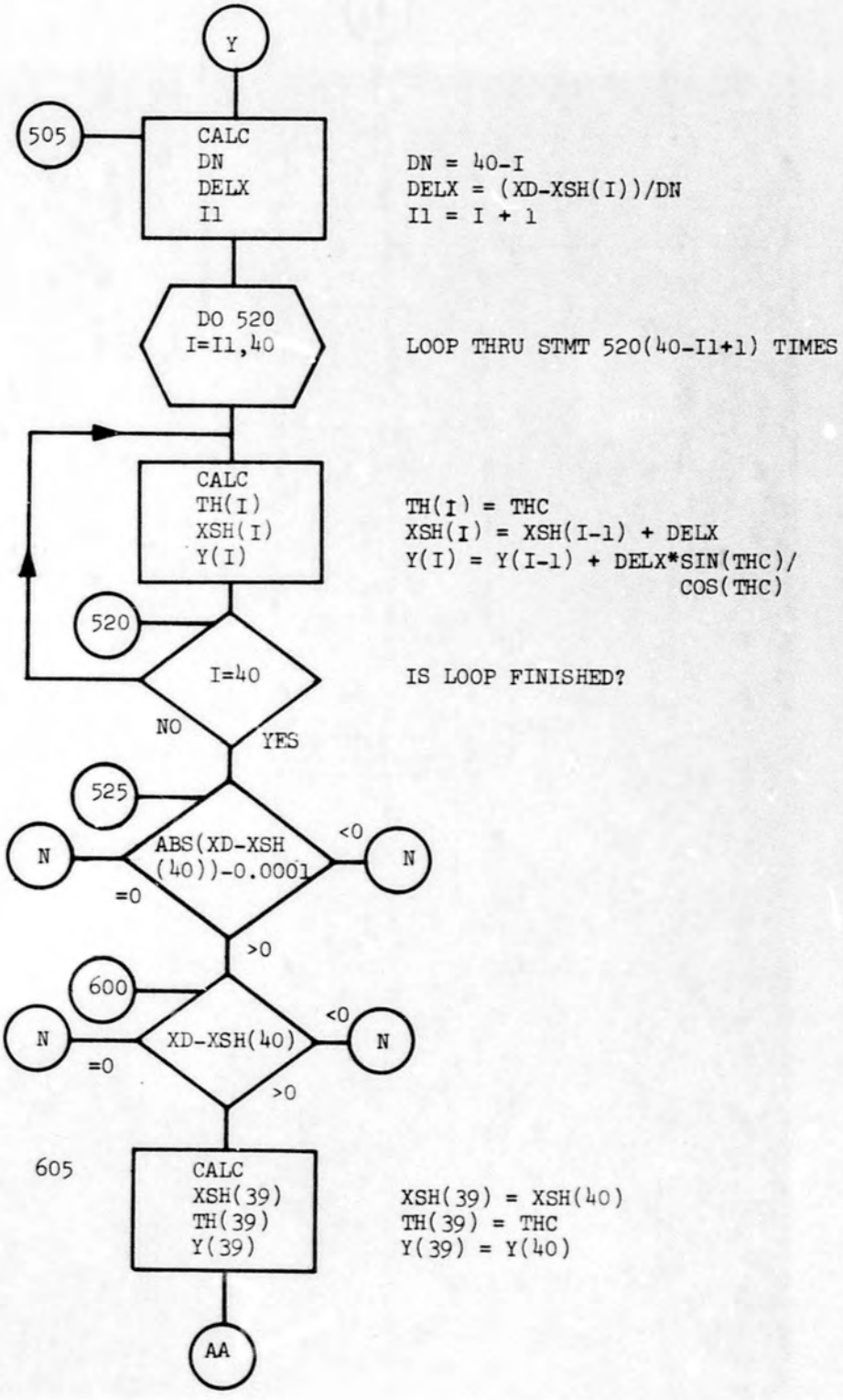


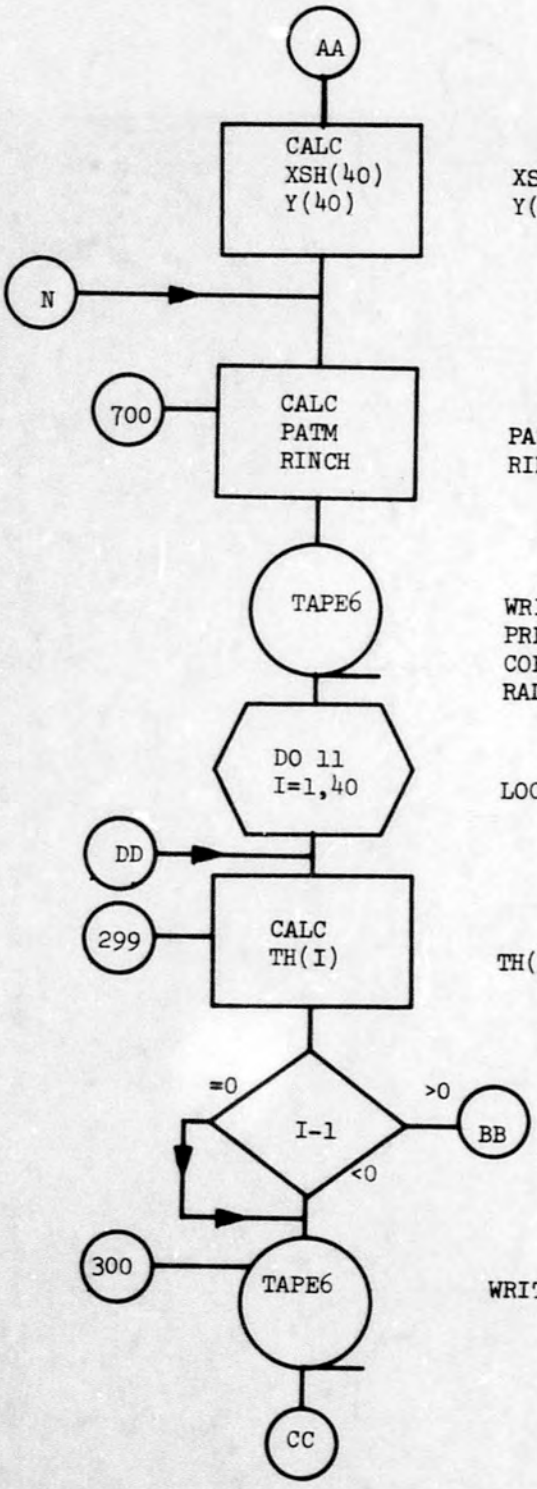
ITERATE TO MATCH
CONICAL FLOW.

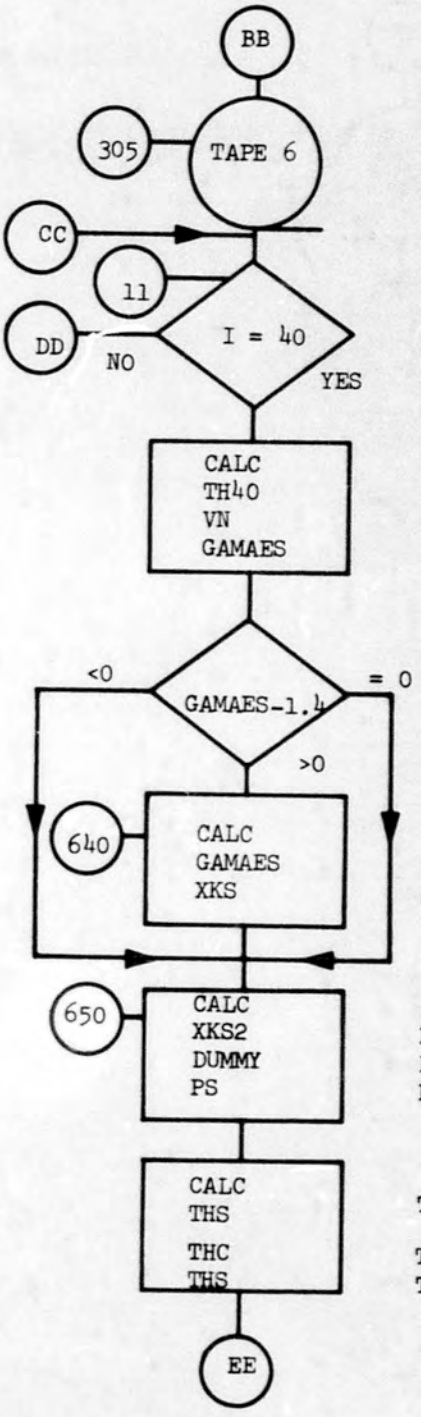
KONE = 1
TH1 = TH(J-1)
X1A = XSH(J-1)

TH2 = TH(J)
DELTH = (TH2-TH1)/DELX
DELX = (THC-TH2)/DELTH
XTEST = DELX/X1A


V
ABS(XTEST)
-.0
=0
<0
>0
465
CALC
THI
X1A
X1
THI = TH2
X1A = X1
X1 = X1 + DELX
W
T
7
CONTINUE
71
X
NO
J = 40
IS LOOP FINISHED?
YES
500
CALC
I
I = J
=0
>0
Z
I-40
Z
<0
Y


Y
505
CALC
DN
DELX
I1
DN = 40-I
DELX = (XD-XSH(I))/DN
I1 = I + 1
DO 520
I=I1,40
LOOP THRU STMT 520(40-I1+1) TIMES
CALC
TH(I)
XSH(I)
Y(I)
TH(I) = THC
XSH(I) = XSH(I-1) + DELX
Y(I) = Y(I-1) + DELX*SIN(THC)/
COS(THC)
520
I=40
IS LOOP FINISHED?
NO
YES
525
ABS(XD-XSH
(40))-0.0001
<0
N
=0
>0
600
XD-XSH(40)
<0
N
=0
>0
605
CALC
XSH(39)
TH(39)
Y(39)
XSH(39) = XSH(40)
TH(39) = THC
Y(39) = Y(40)
AA


AA
CALC
XSH(40)
Y(40)
XSH(40) = XD
Y(40) = Y(39) +
(XSH(40)-XSH(39)**SIN(THC)/
COS(THC))
N
700
CALC
PATM
RINCH
PATM = P/2116.18
RINCH = DIAM/2.0
TAPE6
WRITE TITLE, ALTITUDE, MACH NO., PRESSURE, CHNG CONSTANTS, DRAG COEFFICIENT, STATION AND NOSE RADIUS.
DO 11
I=1,40
LOOP THRU STMT #11 40 TIMES
DD
299
CALC
TH(I)
TH(I) = TH(I)*180.0/PI
=0
I-1
>0
BB
<0
300
TAPE6
WRITE SHOCK SHAPE
CC



WRITE SHOCK SHAPE

IS LOOP FINISHED?

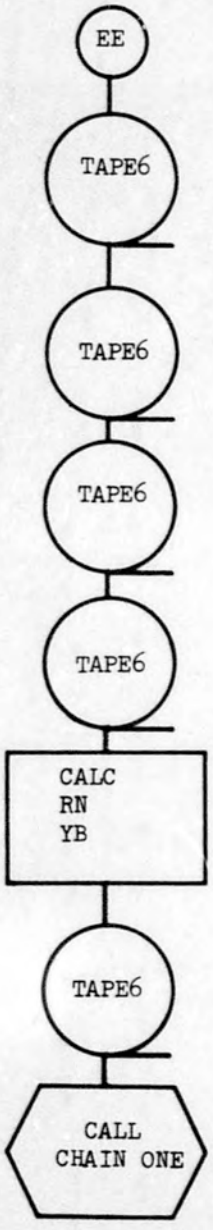
EDGE PROPERTIES BEHIND OBLIQUE SHOCK AT DESIRED BODY STATION

```
TH(40)*PI/180.0
VN = XKS*VEL/VM
GAMAES = Z0+VN*(Z1+VN*(Z2+Z3*VN))
```

```
GAMAES = 1.4
XKS = SQRT((XKC**2)*1.2+0.89570)
```

```
XKS2 = XKS**2
DUMMY = (XKS2-1.0)/GAMAES+1.0)
PS = P0*(1.0+2.0*GAMAES*DUMMY)
```

```
THS = ATAN(SQRT(VM**2-XKS2)/XKS/
      ((VM**2)/2.0 /DUMMY-1.0))
THC = TCH*180.0/PI
THS = THS*180/PI
```



WRITE EDGE PROPERTIES

WRITE (IOTAP,205)

WRITE (IOTAP,206)

WRITE (IOTAP,207)

WIRTE (IOTAP,208)

RN = DIAM/24.
YB = YB*2.0/DIAM

WRITE (IOTAP,209)

TRANSFER CONTROL TO
SEGMENT #1

```
C.  LISTING

      SEGMENT THREE(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
C       COMBINED SHOCK SHAPE,ELLIPTICAL NOSE, AND PRESSURE
C     REVISED BY JACK FIVEL - 18 OCTOBER 1965
      COMMON /234/ SSSS(3960)
      COMMON /333/ JFS(150)
      COMMON /367/ AM367(150)
      COMMON / 456/ INTAP,IOTAP,INT,NMN,M,NGR
      COMMON /666/ SSS(292)
      COMMON /716/   TITLE,  IOPT,  IBOD, IDUM(5), IHEMIS, KODEW,
     1 THETA,    RN,  ALT1,   VEL, X, YJ,  SMXB,  XMXB, DYDX1,  CP10,
     2D1(3),RNFEET,D2(4),
     3     N,    PO,   RHO,     Y,    TH,     P,    YB,    VM, D3(175)
      COMMON /567/ PS,THS,PSTRM(40)
      COMMON /1/ SOS,TATMOS,RHO2
      COMMON /222/ DDT(509),RHW,TITL(5),DDD(5)
      DIMENSION Y(40), TH(40), XSH(40)
      DIMENSION ZS(40)
      DIMENSION TITLE (5)                                                T716
      DIMENSION XPRES(66),YPRES(66),PRX(66),PRY(66)
200    FORMAT(4E15.6)
201   FORMAT(I5)
 202 FORMAT (1HC10X14HSHOCK SHAPE   5A10/
     *9X5HALT =F10.2,3HFT
     1 ,4X9HMACH NO =,F9.5,4X7HP-LOC =,F10.6,4H ATM//10X4HA1 =,F8.5,13X4
     2HB1 =,F8.5,8X4HCD =,F9.6//20X3HX =,F9.4,3H IN,10X4HRN =,F9.6,3H IN
     3//18X2HX/R,17X3HY/R,16X5HANGLE/)
203   FORMAT(F24.6,2F20.6)
204   FORMAT(F44.6,F20.6)
206   FORMAT(2H   4X5HEPS =F10.6,10X7HCPMAX =F10.6,9X6HCP10 =F10.6)
207   FORMAT(2H   4X4HKC =F10.6,11X4HKS =F10.6,12X9HGAMMA-E =F10.6)
208   FORMAT(2H   4X9HTHETA-S =F10.6,6X7HETA-C =F10.6,9X9HP-SHOCK =F10.3)
209   FORMAT(2H   4X7HTHETA =F10.6,8X4HRN =F10.6,12X8HP-BODY =F10.3)
210   FORMAT(2H   4X7HGSTAG =F10.6,8X6HRHO0 =E11.4,9X8HP-ZERO =F10.3)
      DATA  NPRESS,XPRES      / 66, .020,     .0225,     .024,     .026,
     *          .028,     .030,     .032,     .034,     .036,     .038,
     *          .040,     .042,     .044,     .047,     .050,     .054,
     *          .057,     .060,     .065,     .070,     .075,     .080,
     *          .095,     .090,     .095,     .100,     .110,     .12,
     *          .13,      .14,      .15,      .16,      .17,      .18,
     *          .19,      .20,      .215,     .23,      .245,     .26,
     *          .28,      .30,      .32,      .34,      .36,      .38,
     *          .40,      .42,      .44,      .46,      .48,      .50,
```

```
     *          .53,      .56,      .59,      .60,      .64,      .68,
     *          .72,      .76,      .80,      .84,      .88,      .92,
     *          .96,      1.0/
      DATA          YPRES     /     3.7,      3.45,     3.3,      3.15,
     *          3.0,      2.875,    2.77,     2.65,     2.57,     2.475,
     *          2.40,     2.325,    2.27,     2.175,    2.12,     2.02,
     *          1.97,     1.92,     1.84,     1.775,    1.725,    1.675,
     *          1.625,    1.60,     1.57,     1.54,     1.48,     1.44,
     *          1.41,     1.38,     1.36,     1.35,     1.34,     1.335,
     *          1.34,     1.345,    1.355,    1.375,    1.4,      1.45,
     *          1.50,     1.55,     1.59,     1.65,     1.71,     1.775,
     *          1.84,     1.90,     1.94,     2.0,      2.04,     2.09,
     *          2.125,    2.175,    2.19,     2.19,     2.185,    2.175,
     *          2.16,     2.14,     2.13,     2.12,     2.115,    2.110,
     *          2.05,     2.1/
      INTAP = 5
      IOTAP = 6
      KONE=-1
      DO 50 I=1,NPRESS
      PRX(I) = ALOG(XPRES(I))
      PRY(I) = ALOG(YPRES(I))
   50 CONTINUE
      PI=3.1415926
C
C     PROPERTIES BEHIND NORMAL SHOCK
C
      CALL SHOCKR (PO,RHO,VEL,PI/2.0,XDUMY,GSTAG,XDUMZ,MACH)
      EPS=RHO/RHO2
      CPMAX=2.0-EPS
      IF(IHEMIS)160,160,170
C
C     NOSE PRESSURE DRAG TERM FOR HEMISPHERICAL NOSE
C
  160 COS2=(COS(THETA*PI/180.0))**2
      SIN2=(SIN(THETA*PI/180.0))**2
      YN2=(RN*COS(THETA *PI/180.0)/12.0)**2
      CP10 = PI*YN2*(1.0-COS2/2.0)
      DN=RN/6.0
      BB=1.0
      GO TO 180
  170 CONTINUE
C
C     CURVE FITTING ELLIPSE TO ABLATED NOSE
C
      Y1 = YJ
      X1MS=XMXB-SMXB
```

```
      X1=Y1/DYDX1
      S=X1-X1MS
      X0=S**2/(2.0*S-X1)
      A=X0-S
      B=A*Y1/(A**2-(X0-X1)**2)**.5
      BB=(B/A)**2
      DN=B/6.
  180 CONTINUE
      CDAN=.940*CPMAX*CP10
      CD=1.2732395*CDAN/DN**2
      A1=.475+.385/BB
      B1=.235+.340/(BB+1.)
      A2=.24389
      B2=.659656
      C2=.20412
      DIAM=DN*12.
       CDSAVE=CD
      DEL = THETA
C      BLUNT BODY PRESSURE CORRELATION
C      FROM FRANK MAGIERA
C      PROGRAMMED BY JACK FIVEL 5 OCTOBER 1965
      TH2=(THETA*PI/180.)**2
      DN=DN*12.
      XAXIS = X * TH2/DN/SQRT(CD)
C      TABLE LOOK-UP FOR PRESSURE COEFFICIENT
      IF(XAXIS-1.0)150,120,120
  120 CPTH=2.1
      GO TO 199
  150 XLOGX = ALOG(XAXIS)
      CALL DISCOT (XLOGX,DUMY,PRX,PRY,DUMZ,10,NPRESS,0,CPLOG,0)
      CPTH = EXP(CPLOG)
C      CALCULATE PRESSURE
  199 CP=CPTH*TH2
      P=PO*(1.0+.7*(VM**2)*CP)
      DN=2.0
      XD=2.*X/DIAM
C     SHOCK SHAPE
C      PROGRAMMED BY FRANK MAGIERA
C      REVISED BY JACK FIVEL 22 SEPTEMBER 1965
      EPS=EPS**.5
      DEL=DEL*PI/180.0
      CD=CD**.5
      GAMK=1.04+1.20*(GSTAG-1.15)
C
C     CONICAL SHOCK PROPERTIES
C
```

```
      XKC=VM*SIN(DEL)
C
C     OBTAIN COEFFICIENTS FOR GAMMA-E AT ALTITUDE
C
      CALL GAMCOF(Z0,Z1,Z2,Z3,ALT1)
      GAMAE=1.4
  400 XKS=SQRT((XKC**2)*(GAMAE+1.0)/2.0+0.89570)
      VN=XKS*VEL/VM
      GAMAEX=Z0+VN*(Z1+VN*(Z2+Z3*VN))
      IF(ABS(GAMAEX-GAMAE)-0.0001)410,410,405
  405 GAMAE=GAMAEX
      GO TO 400
  410 CONTINUE
      IF(GAMAEX-1.4)412,412,411
  411 GAMAEX=1.4
      XKS=SQRT((XKC**2)*1.2+0.89570)
  412 GAMAE=GAMAEX
      THC=ASIN(XKS/VM)
      YCO=1.414*DEL/DN/CD
      XCO=GAMK*DEL**2/CD/DN
      DELA=PI/60.0
      XN=40
      MIRELS=30
  413 Y(1)=0.0
      TH(1)=PI/2.0
C
C     SHOCK SHAPE BY MIRELS-THORNTON
C
      DO 1 I=2,MIRELS
      TH(I)=TH(I-1)-DELA
      SLOPE=SIN(TH(I))/COS(TH(I))
      X1=(YCO*SLOPE/A1/B1/XCO**B1)**(1./(B1-1.))
      IF(X1*XCO-.1)4,4,5
    4 Y1=A1*(XCO*X1)**B1/YCO
      Y(I)=Y1
      XSH(I)=X1
      IF(MIRELS-40)3,101,101
    3 IF(X1-XD)1,2,2
  101 IF(I-40)1,700,700
    1 CONTINUE
      I=31
      XI=30.0
      GO TO 6
C
C     X1 EXCEEDED XD IN MIRELS-THORNTON
C
```

```
  2     SLOPE = A1*B1*(XCO**B1)*(XD**(B1-1.0))/YCO
        TMIR=ATAN(SLOPE)
        DELA=(PI/2.0-TMIR)/39.0
        MIRFLS=40
        GO TO 413
C
C       SHOCK SHAPE ACCORDING TO CHENG
C
    5   XI=I-1
        X1=XSH(I-1)
    6   CONTINUE
        DELX=(XD-X1)/(XN-XI)
        DELX=ABS(DELX)
        XCO2=XCO/EPS/GAMK
        YCO2=YCO/EPS/1.414
        DO 71 J=1,40
        X1=X1+DELX
  395   XP=ALOG10(X1*XCO2)
        Y(J)=A2*XP**2+B2*XP+C2
        Y(J)=10.**Y(J)
        Y(J)=Y(J)/YCO2
        XSH(J)=X1
        TH(J)=Y(J)/X1*(2.0*A2*ALOG10(X1*XCO2))
        TH(J)=TH(J)+Y(J)*B2/X1
        TH(J) = ATAN(TH(J))
C
C       CHECK TO SEE IF CONICAL SHOCK IS REACHED
C
        IF(KONE)450,452,460
  450   IF(TH(J)-THC)451,451,7
C
C       WE ARE PAST THE FIRST DIP IN THE CURVE
C
  451   KONE=0
        GO TO 7
  452   IF(TH(J)-THC)7,500,455
C
C       ITERATE ON X1 TO HIT CONE SHOCK ANGLE
C
  455   KONE=1
        TH1=TH(J-1)
        X1A=XSH(J-1)
  460   TH2=TH(J)
        DELTH=(TH2-TH1)/DELX
        DELX=(THC-TH2)/DELTH
        XTEST=DELX/X1A
```

```
      IF(ABS(XTEST)-0.0001)500,500,465
  465 TH1=TH2
      X1A=X1
      X1=X1+DELX
      GO TO 395
    7 CONTINUE
   71 CONTINUE
C
C     COORDINATES FOR CONICAL SHOCK
C
  500 I=J
      IF(I-40)505,525,525
  505 DN=40-I
      DELX=(XD-XSH(I))/DN
      I1=I+1
      DO 520 I=I1,40
      TH(I)=THC
      XSH(I)=XSH(I-1)+DELX
      Y(I)=Y(I-1)+DELX*SIN(THC)/COS(THC)
  520 CONTINUE
  525 CONTINUE
      IF(ABS(XD-XSH(40))-0.0001)700,700,600
  600 IF(XD-XSH(40))700,700,605
C
C     LAST X FALLS SHORT OF XD
C
  605 XSH(39)=XSH(40)
      TH(39)=THC
      Y(39)=Y(40)
      XSH(40)=XD
      Y(40)=Y(39)+(XSH(40)-XSH(39))*SIN(THC)/COS(THC)
  700 CONTINUE
      PATM=P/2116.18
      RINCH=DIAM/2.0
      WRITE (6,1000) TITLE
 1000 FORMAT (1H1,30X,24HTURBULENT BOUNDARY LAYER/,1H0,3CX,5A10)
      WRITE (IOTAP,202) TITLE,ALT1,VM,PATM,A1,B1,CDSAVE,X,RINCH
      DO 11 I=1,40
  299 TH(I)=TH(I)*180.0/PI
      IF(I-1)300,300,305
  300 WRITE(IOTAP,204)Y(1),TH(1)
      GO TO 11
  305 WRITE(IOTAP,203)XSH(I),Y(I),TH(I)
   11 CONTINUE
C
C     EDGE PROPERTIES BEHIND OBLIQUE SHOCK AT DESIRED STATION
```

```
C
      TH40=TH(40)*PI/180.0
      CALL SHOCKR(PO,RHO,VEL,TH40,THS,XDUMY,PS,MACH)
      IF(MACH.EQ.1) PS=P
      THC=THC*180.0/PI
      THS=THS*180.0/PI
      WRITE(IOTAP,206)EPS,CPMAX,CP10
      WRITE(IOTAP,207)XKC,YKS,GAMAE
      WRITE(IOTAP,208)THS,THC,PS
      RNFEET = DIAM/24.
      RN = DIAM/2.
      YB=YB*2.0/DIAM
      WRITE(IOTAP,209)THETA,RN,P
      WRITE(IOTAP,210)GSTAG,RHO,PO
      CALL CHAIN (ONE)
      END
```

d. List of Variables

Symbolic representation	Meaning or use
Y1	Body radius at nosecone junction point (used if nose is not spherical)
PO	Atmospheric pressure
RHO	Atmospheric density
VEL	Velocity
SOS	Speed of sound
THTMOS	Atmospheric temperature
RHO2	Density behind normal shock
EPS	Density ratio across normal shock (or $\sqrt{\rho/\rho_2}$)
CPMAX	Maximum pressure coefficient for normal shock
BB	Bluntness ratio
CP10	Product of nose drag coefficient and projected nose area divided by maximum pressure coefficient
CDAN	Corrected Newtonian drag coefficient-area product for nose
CD	Nose drag coefficient for shock shape calculation
XKC	Hypersonic similarity parameter $M_\infty \sin \theta_c$
XKS	Hypersonic similarity parameter $M_\infty \sin \eta_s$
GAMAE	Effective ratio of specific heats
THS	Flow angle at body station of interest
THC	Shock angle of oblique shock
PS	Pressure behind oblique shock
THETA	Cone angle
RN	Nose radius
P	Body pressure

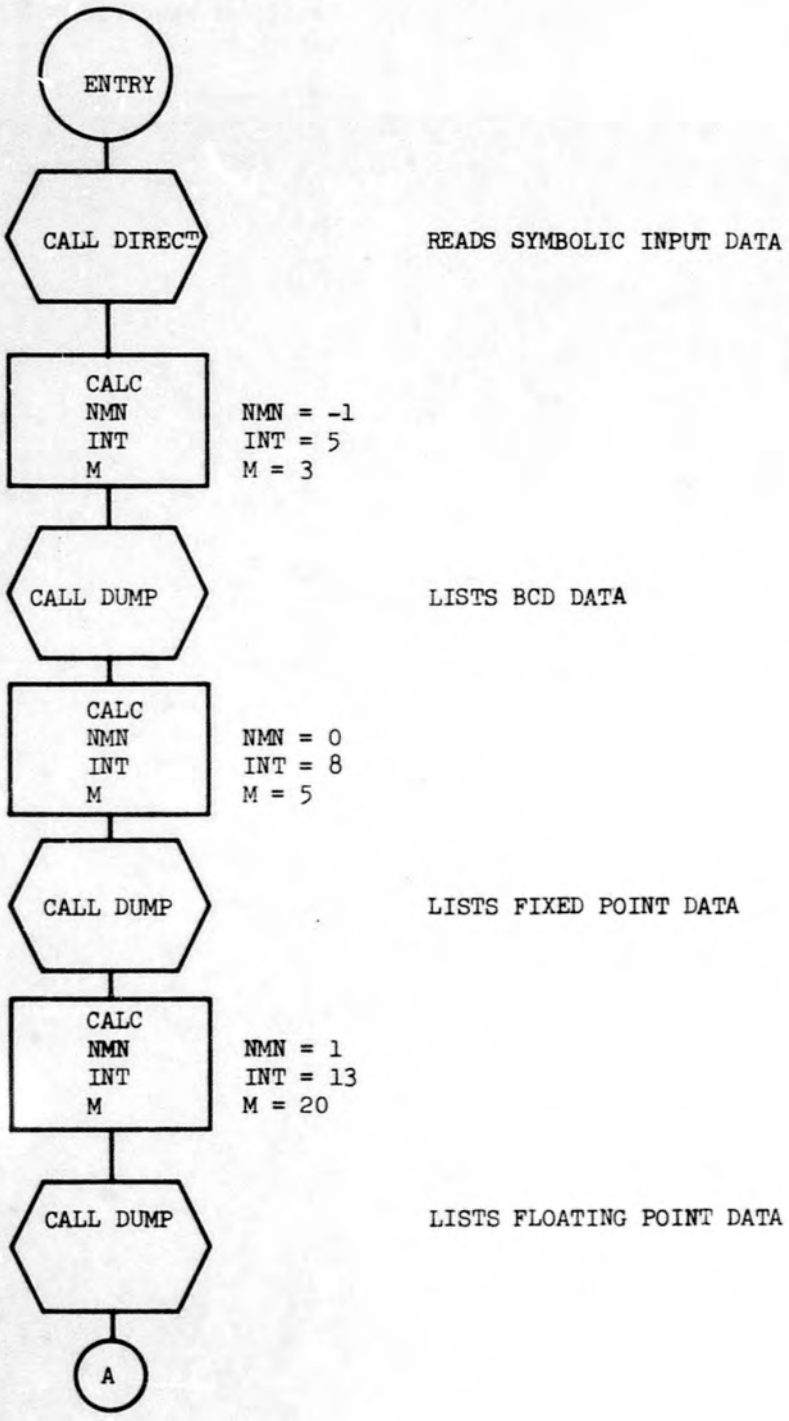
GSTAG	Ratio of specific heats behind normal shock
XD	x/r, nondimensional axial distance
VN	Velocity normal to oblique shock
GAMK	Parameter K in M - T shock shape

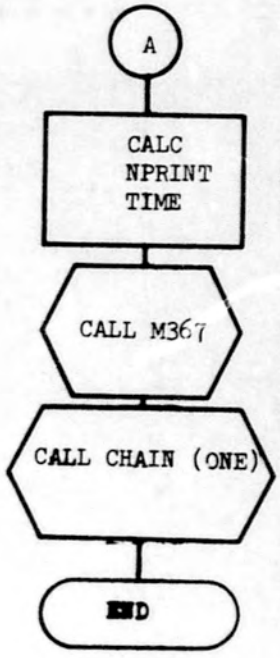
7. SEGMENT FOUR

a. Function

Segment FOUR performs the ablation weight loss calculation for a conical vehicle. The main portions of the calculations are done in subroutines that are described in detail in reference 1. These routines are also listed in Section IV of this report. The mainline portion of Segment FOUR is quite similar to Segment TWO in that it serves to schedule the reading of input data, and lists the input data for ready reference. The ablation mass loss calculation is initiated by calling Subroutine M367. Once the mass loss rates have been calculated, control is returned to Segment ONE.

b. Flow Chart


ENTRY
CALL DIRECT
READS SYMBOLIC INPUT DATA
CALC
NMN
INT
M
NMN = -1
INT = 5
M = 3
CALL DUMP
LISTS BCD DATA
CALC
NMN
INT
M
NMN = 0
INT = 8
M = 5
CALL DUMP
LISTS FIXED POINT DATA
CALC
NMN
INT
M
NMN = 1
INT = 13
M = 20
CALL DUMP
LISTS FLOATING POINT DATA
A


A
CALC
NPRINT
TIME
NPRINT = 0
TIME = 0.
CALL M367
ABLATIVE MASS LOSS
CALCULATION
CALL CHAIN (ONE)
RETURN CONTROL TO
SEGMENT #1
END

```
C.  LISTING

      SEGMENT FOUR(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
CM367E  ABLATION MASS LOSS
      COMMON /234/ SSSS(3960)
      COMMON /333/ JFS(150)
      COMMON /367/ AM367(150)
      COMMON /456/ INTAP,IOTAP,INT,NMN,M,NGR
      COMMON /666/ SSS(292)
      COMMON /716/ TITLE(5),DUMMY(11),ALT,VO,DDD(7),ALO,EMD1,D5,
     1 WABL1,WABL2,D3(257),TSS1,D4(5)
      COMMON /567/ PS,THS,PSTRM(40)
      COMMON /1/ SOS,TATMOS,PHO2
      COMMON /222/ DDT(509),RHW,TITL(5),DDDD(5)
      DIMENSION BCDA(41), IR(39)
      DIMENSION      XL   (  5),     CO   (  5),     CI   (  5),                15
     *               RO   (  5),     EMO  (  5),     EMI  (  5),                30
     *               TI   (  5),     AB   (  5),     ETL  (  5),                45
     *               ETT  (  5),     ZTL  (  5),     ZTT  (  5),                60
     *CPO  (  3),    RHOS (  4),
     *               ABLAT(  5),     TITLA(  9),     TITLB(  9),               490
     *               TITLC(  9),     KASE (  5),     MVSV (  6),               510
     *               RE   (  3)
      EQUIVALENCE(AA(1),PAGE)
      EQUIVALENCE   (AM367(  1), PAGE    ),(AM367(  2), ICODE     )
     *,             (AM367(  3),TITLA(1)),(AM367( 12), TITLB(1) )
     *,             (AM367( 21),TITLC(1)),(AM367( 30), IRETML   )                34
     *,             (AM367( 35),NOK     ),(AM367( 41), I20      )                 4
     *,             (AM367( 42),I22     ),(AM367( 43), NMAT     )                43
     *,             (AM367( 44),RAD     ),(AM367( 45), RETMLC   )                45
     *,             (AM367( 46),DTAU    ),(AM367( 47), TR       )                47
     *,             (AM367( 48),RE(1)   ),(AM367( 51), DELS     )                51
     *,             (AM367( 52),XL(1)   ),(AM367( 57),CO(1)     )
     *,             (AM367( 62),CI(1)   ),(AM367( 67),RO(1)     )
     *,             (AM367( 72),EMO(1)  ),(AM367( 77),EMI(1)    )
     *,             (AM367( 82),TI(1)   ),(AM367( 87),AB(1)     )
     *,             (AM367( 92),ETL(1)  ),(AM367( 97),ETT(1)    )
     *,             (AM367(102),ZTL(1)  ),(AM367(107),ZTT(1)    )
     *,             (AM367(112),CPO(1)  ),(AM367(115),RHOS(1)   )
     *,             (AM367(119),ABLAT(1)),(AM367(124),NPRNT(1)  )
     *,             (AM367(125),TIME    )
      EQUIVALENCE (AM367(126),TRAD),(AM367(131),AMODEL),
     1            (AM367(132),TRCUT)
      EQUIVALENCE (AM367(31),K1)
```

```
      EQUIVALENCE (D3(2),IGO)
      K1 = IGO
      IF(K1 - 6) 200,100,200
      DIMENSION TRAD(5)
      DATA BCDA /4HM367,39,,  4HPAGE, 5HICODE, 5HTITLA, 5HTITLB,
     1 5HTITLC,6HIRETML,        3HNOK,   3HI20,   3HI22,  4HNMAT,
     2   3HRAD,6HRETMLC,       4HDTAU,    2HTR,    2HRE,  4HDELS,
     3                                             2HXL,    2HCO,
     4    2HCI,    2HRO,        3HEMO,   3HEMI,    2HTI,    2HAB,
     5   3HETL,   3HETT,        3HZTL,   3HZTT,   3HCPO,  4HRHOS,
     6 5HABLAT, 5HNPRNT,       4HTIME,  4HTRAD,6HAMODEL/
      DATA IR /1,1,3*9,5,6,7*1,3,1,12*5,3,4,5,1,5,1/
C
C     CALCULATE SHEAR STRESS FOR CLEAN AIR ONLY
C     TRIGGERED WHEN D3(2) = 6
C
C     (FROM COMMON/T716/)
  100 RAD = DUMMY(11)
      TI(1) = DDD(7)
      RE(3) = DUMMY(10)
      XL(1) = DDD(1)
      NMAT = 1
      GO TO 300
  200 CONTINUE
      CALL DIRECT (AM367,BCDA)
      WRITE (6,1000) TITLE
 1000 FORMAT (1H130X24HTURBULENT BOUNDARY LAYER/1H030X5A10)
      NMN = -1
      INT = 5
      M = 3
      CALL DDUMP(AM367,BCDA,IR)
C
      NMN = 0
      INT = 8
      M =  5
      CALL DDUMP(AM367,BCDA,IR)
C
      NMN = 1
      INT = 13
      M = 20
      CALL DDUMP(AM367,BCDA,IR)
  300 CONTINUE
      NPRNT=0
      TIME=0.
      CALL M367 (ALT,V0,ALO,TSHSTR,EMD1)
   10 CONTINUE
```

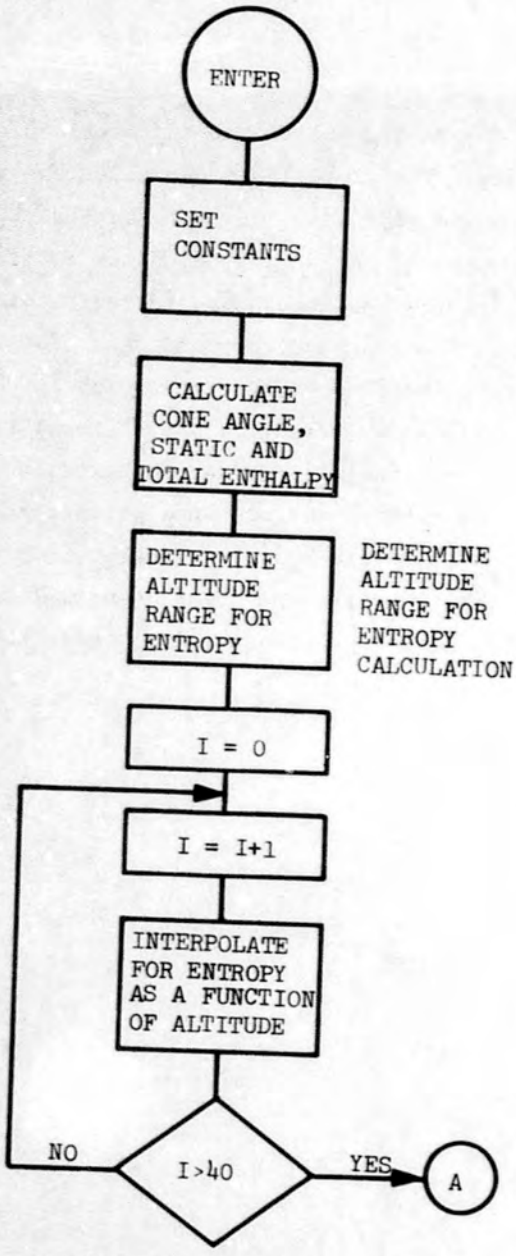
```
CALL CHAIN (ONE)
END
```

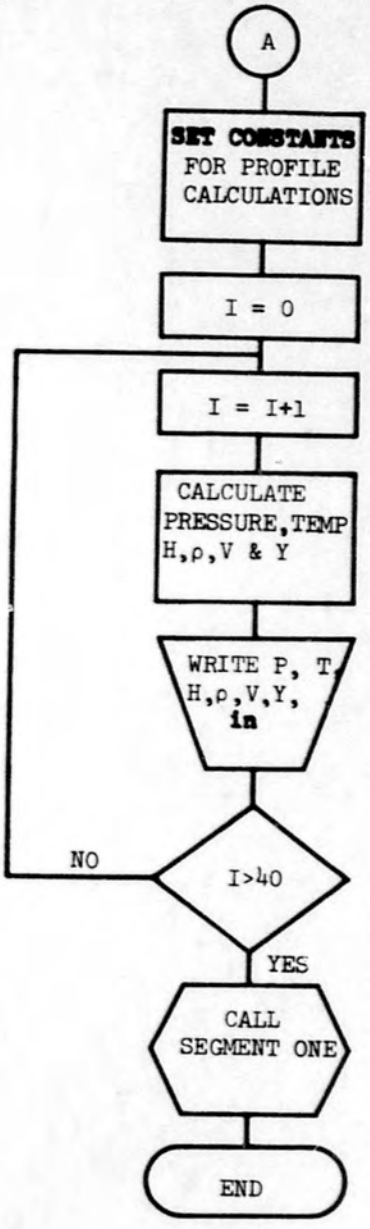
8. SEGMENT FIVE

a. Function

Segment FIVE calculates the real gas streamtube properties of the inviscid flow surrounding the body. The properties calculated are entropy, density, enthalpy, velocity, pressure, and mass flow rate versus the normal distance between the body and the shock. As described in the section on theory, the shock shape and shock angle are the variables used to begin this calculation. These values are used to calculate the shock entropy and mass flow rate as a function of position. The pressure and flow angle are assumed to vary linearly with mass flow between the body and shock. With the use of these variations, pressure and entropy are used to find the temperature, enthalpy, and density at each streamtube point, or mass flow rate. Some of these properties are in turn used to calculate the location normal to the body that corresponds to a given mass flow rate. This array of mass flow rate and location normal to the body surface is used in Segment ONE to locate the boundary layer edge through the mass balance procedure.

b. Flow Chart


ENTER
SET CONSTANTS
CALCULATE CONE ANGLE, STATIC AND TOTAL ENTHALPY
DETERMINE ALTITUDE RANGE FOR ENTROPY
DETERMINE ALTITUDE RANGE FOR ENTROPY CALCULATION
I = 0
I = I+1
INTERPOLATE FOR ENTROPY AS A FUNCTION OF ALTITUDE
I>40
NO
YES
A


A
SET CONSTANTS FOR PROFILE CALCULATIONS
I = 0
I = I+1
CALCULATE PRESSURE, TEMP H, ρ, V & Y
WRITE P, T, H, ρ, V, Y, in
NO
I>40
YES
CALL SEGMENT ONE
END

```
      C.  LISTING

      SEGMENT FIVE(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
C     REAL GAS STREAMTUBE METHOD
C     FEB 18,1965
C     F.T.MAGIERA  PROGRAMMER
C REVISED AND INCORPORATED AS A SUBROUTINE  AUG.19,1965
C           BY JANE SCHAEFER
      COMMON /SHOCK/ DSL1(31,2,8)
      COMMON /234/ SSSS(3960)
      COMMON /333/ JFS(150)
      COMMON /367/ AM367(150)
      COMMON / 456/ INTAP,IOTAP,INT,NMN,M,NGR
      COMMON /566/ SSS(292)
      COMMON /716/   TITLE,  IOPT,  IBOD,  IEND, NABL1, NABL2, IREAD,
     1  KODE,IHEMIS, KODEW,
     2ANTL,RNINCH,ALT1,VEL1,XINCH,Y1,SMXB,XMXB,DYDX1,
     3CP10,TW,EM,EMDOT,RN,WAPL1,WABL2,YBB,
     4    K1,     N,     P1,  PH01,    XMD,     SS,     P,     YB,     VM,
     5    VX,    HX,     RX,    XMX,     T0,  RHOW,     HW,XMASSO, EMDTO,
     6  VPTO, SWTCH, TWOGJ,     PP,    TAU,    CI ,NTOTAL,NWHICH,  ISSI,
     7     L
      COMMON /567/ PSHOCK,THS,PSTPM
      COMMON /1/ SOS,TATMOS,RHO2
      COMMON /222/ YY,ES,PHOOW,TRS,CES,ELS,HZ,HAW,HE,PHW,TITL(5),
     * C5,C6,UF,B,CIP
      DIMENSION  VX(40),  HX(40),     RX(40),    XMX(40)
      DIMENSION TITLE (5)                                              T716
      DIMENSION TRS(101),CES(101),ELS(101) ,RHOOW(101)
      DIMENSION YY(101), VX(40)
      DIMENSION XMD(40),SS(40)
      DIMENSION PSTPM(40)
  898 FORMAT(F12.4,F10.1,2XE11.4,2E11.2,2XE11.4,F12.3)
      NIN=40
      INTAP = 5
      IOTAP = 6
      GR = 32.174
      CONJ = 778.
      ANGL = ANTL
      WRITE (IOTAP,878)TITLE
  878 FORMAT (1H130X24HTURBULENT BOUNDARY LAYER/ 1H030X5A10/1H0
     *25X28HINVISCID FLOW FIELD PROFILES/1H0
     *7X3HY/R,7X2HSS,10X3HRH0,9X1HH,10X1HV,11X3HXMD,8X5HPSTPM/
     *14X8HFT2/SEC2,4X8HLB/CU.FT,5X6HBTU/LB,5X6HFT/SEC,6X6HLB/SEC,
```

```
     *8X3HPSF/)
 3000 FORMAT(5E15.7)
      ANGL=ANGL/57.295779
      H=3.4885423*P1/RH01
      HT=H+(VEL1**2)/2.
C     CALCULATE MASS FLOW
      JALT1=(ALT1-.01)/10000.
      ALT2=JALT1
      ALT2=ALT2*10000.
C
C     DETERMINE ALTITUDE RANGE FOR ENTROPY CURVES
C
      JALT1=JALT1+1
      JALT2=JALT1+1
 4006 DO 2 I=1,N1N
  714 BETA=SS(I)/57.295779
  715 IF(IBOD)700,700,701
  700 XMD(I)=RH01*VEL1*XMD(I)
      GO TO 716
  701 XMD(I)=RH01*VEL1*3.1415926*(XMD(I))**2
  716 VN=VEL1*SIN(BETA)
C     INTERPOLATE FOR ENTROPY
      IVN=VN/4000.
      IVN=IVN+1
      BSD1=DSL1(JALT1,1,IVN)+DSL1(JALT1,2,IVN)*VN
      BSD2=DSL1(JALT2,1,IVN)+DSL1(JALT2,2,IVN)*VN
    2 SS(I)=BSD1+(BSD2-BSD1)*(ALT1-ALT2)/10000.
    5 CONTINUE
      AMF1=0.
      Y=YB
      YB2=YB**2
C
C     DETERMINE ENTROPY BEHIND NORMAL SHOCK
C     DETERMINE TEMPERATURE FOR ENTROPY AND PRESSURE
C
      PI=3.1415926
      THC=ANTL*PI/180.0
      THS=THS*PI/180.0
      TANTHC=SIN(THC)/COS(THC)
      TANTHS=SIN(THS)/COS(THS)
      C1=(PSHOCK-P)/XMD(N1N)
      C2=(TANTHS-TANTHC)/XMD(N1N)
      DO 56 JK=1,N1N
      S=SS(JK)
      PSTRM(JK)=P+C1*XMD(JK)
      PST=PSTRM(JK)
```

```
      CALL ISNTRP(PST,S,T)
      PO=ALOG10(PST/2116.18)
C
C     OBTAIN ENTHALPY AND DENSITY FOR TEMPERATURE AND PRESSURE
C
      CALL HANSEN(3,T,PO,H)
      CALL HANSEN(5,T,PO,Z)
      RX(JK)=39.65*PST/(Z*T*2116.18)
      HX(JK)=H
      H=H*GP*CONJ
      R=RX(JK)/GR
   55 V=(2.*(HT-H))**.5
      THETA=ATAN(TANTHC+C2*XMD(JK))
      RXV=R*V*COS(THETA)
      IF(JK-1)2,2,8
    8 AMFI=AMFI+2.*(XMD(JK)-XMD(JK-1))/(RXVX+RXV)
      IF(IBOD)11,11,12
   11 Y=AMFI+YB
      GO TO 9
   12 Y=SQRT(AMFI/3.1415927+YB2)
    9 YX(JK) = Y
      VX(JK) = V
      XMX(JK) = XMD(JK)*GP*RN**2
      WRITE (IOTAP,898) YX(JK),SS(JK), RX(JK), HX(JK), VX(JK), XMX(JK),
     *PSTRM(JK)
   56 RXVX=RXV
   57 CONTINUE
      CALL CHAIN (ONE)
      END
```

d. Symbols

Symbolic representation	Meaning or use
YX	Normal distance/nose radius at body station of interest
SS	Entropy (ft^2/sec^2)
RX	Density (lb/ft^3)
HX	Enthalpy (B/lb)
VX	Velocity (ft/sec)
XMX	Mass flow rate (lb/sec)
PSTRM	Pressure (psf) $\left(P_{LOCAL}\right)$
GR	Gravity constant, 32.178
CONJ	778
ANTL	θ_c
ANGL	θ_c
P1	Pressure (free stream)
RHO1	Density (free stream)
H	Free stream enthalpy
HT	Total enthalpy
N1N	40 (DO LOOP index)
SS	Shock angle from shock shape (degrees)
XMD	Y/R from shock shape (also $\dot{n}(y)$) (slug/sec)
IBOD	0, flat plate (no provisions for flat plate shock shape)
BETA	Shock angle from shock shape (rad)
VN	Velocity normal to shock
YB	Body radius at local station
YB2	$(YB)^2$
AMFI	Dummy for intermediate calculation

THC	θ_c (rad)
THS	η_s (rad)
TANTHC	Tan θ_c
TANTHS	Tan η_s
C_1	$\left(P_s - P_b\right)/\dot{m}_s$
C_2	$\left(\tan \theta_c - \tan \eta_s\right) \dot{m}_s$
S	Entropy
PST	Pressure (flow field pressure) $\left(P_{LOCAL}\right)$
T	Temperature
P	Body pressure
PO	Log_{10} PST
H	Local enthalpy
Z	Local compressibility
V	Local velocity
THETA	Local flow angle
R	Density
RXV	$\rho u = \rho V_{LOCAL} \cos \theta$(at JK)
RXVX	RXV at (JK-1)
AMFI	$\frac{2\Delta\dot{m}}{(\rho u)_{JK-1} + (\rho u)_{JK}}$ Intermediate calculation in determining y location for given $\dot{m}$

e. Block Data Shock

This block data subroutine tabulates the coefficients from which the entropy behind a normal shock may be determined by the expression

$$S_2 = a + bV_N$$

Entropy values were calculated at the pressure and density behind a normal shock wave given in references 12 and 13. These entropy values for the velocity range of 0 to 32,000 feet per second were then curve-fitted by straight line segments for eight equal velocity regions, at altitudes from 0 to 300,000 feet

in increments of 10,000 feet. The coefficient a is given by

DSL1 (ALT, 1, VEL)

and the coefficient b by

DSL1 (ALT, 2, VEL)

```
F.  BLOCK DATA

 BLOCK DATA
 COMMON /SHOCK/ DSL1(31,2,8)
 DATA (DSL1(I),I=   1,  62) /
1  38629.81000,  38764.48000,  38920.34000,  39096.98000,
2  39540.74000,  40361.22000,  41180.75000,  41999.44000,
3  42817.34000,  43832.88000,  44872.42000,  45883.99000,
4  46855.29000,  47796.83000,  48699.50000,  49577.19000,
5  50322.04000,  50943.16000,  51410.77000,  51421.68000,
6  51892.34000,  52363.94000,  52861.66000,  53411.10000,
7  54014.23000,  54683.33000,  55351.23000,  56308.81000,
8  57371.46000,  58340.65000,  59358.34000,
1       1.23520,       1.30298,       1.37453,       1.45266,
2       1.50176,       1.50175,       1.50175,       1.50175,
3       1.50174,       1.47360,       1.43660,       1.40168,
4       1.36624,       1.33235,       1.30127,       1.27027,
5       1.25482,       1.25484,       1.27498,       1.39371,
6       1.40459,       1.42293,       1.44387,       1.46188,
7       1.47716,       1.48920,       1.51675,       1.52474,
8       1.52023,       1.53916,       1.54845/
 DATA (DSL1(I),I=  63, 124) /
1  38181.02000,  38526.25000,  38884.79000,  39302.70000,
2  39892.79000,  40709.14000,  41528.07000,  42349.60000,
3  43156.43000,  44080.16000,  45008.58000,  45911.52000,
4  46756.78000,  47585.84000,  48396.86000,  49195.48000,
5  49866.56000,  50467.43000,  50983.33000,  51421.68000,
6  51892.34000,  52363.94000,  52861.66000,  53411.10000,
7  54014.23000,  54683.33000,  55351.23000,  56308.81000,
8  57371.46000,  58340.65000,  59358.34000,
1       1.34740,       1.36254,       1.38342,       1.40123,
2       1.41375,       1.41477,       1.41492,       1.41421,
3       1.41697,       1.41178,       1.40257,       1.39480,
4       1.39087,       1.38510,       1.37693,       1.36570,
5       1.36869,       1.37378,       1.38184,       1.39371,
6       1.40459,       1.42293,       1.44387,       1.46188,
7       1.47716,       1.48920,       1.51675,       1.52474,
8       1.52023,       1.53916,       1.54845/
 DATA (DSL1(I),I= 125, 186) /
1  39603.19000,  39957.07000,  40405.59000,  40976.41000,
2  41473.99000,  42144.99000,  42829.27000,  43534.40000,
3  44241.00000,  44949.86000,  45650.37000,  46322.11000,
4  47013.36000,  47636.80000,  48190.31000,  48676.79000,
5  49254.10000,  49786.16000,  50213.99000,  50539.22000,
6  50888.34000,  51353.08000,  51905.12000,  52436.40000,
7  52933.86000,  53402.47000,  54092.04000,  54762.97000,
8  55322.51000,  56062.65000,  56795.81000,
1       1.16963,       1.18369,       1.19332,       1.19202,
2       1.21610,       1.23529,       1.25227,       1.26611,
3       1.28140,       1.30307,       1.32235,       1.34238,
4       1.35880,       1.37873,       1.40275,       1.43054,
5       1.44525,       1.45894,       1.47801,       1.50402,
6       1.53009,       1.54929,       1.56344,       1.58372,
7       1.61221,       1.64931,       1.67415,       1.71797,
8       1.77635,       1.82391,       1.86877/
 DATA (DSL1(I),I= 187, 248) /
1  40322.37000,  40636.01000,  40975.25 00,  41149.83000,
2  41816.13000,  42400.36000,  42987.58000,  43533.57000,
3  44160.29000,  44738.68000,  45387.46000,  46049.27000,
```

```
4  46689.70000,  47176.87000,  47821.80000,  48420.58000,
5  49110.35000,  49519.31000,  49814.29000,  50219.06000,
6  50680.41000,  51197.32000,  51655.72000,  51966.25000,
7  52454.03000,  53187.56000,  54001.93000,  54784.69000,
8  55768.81000,  57138.00000,  57924.54000,
1      1.10970,      1.12717,      1.14585,      1.17757,
2      1.18759,      1.21401,      1.23908,      1.26618,
3      1.28821,      1.32067,      1.34426,      1.36585,
4      1.38594,      1.41706,      1.43346,      1.44606,
5      1.45723,      1.48193,      1.51132,      1.53070,
6      1.54742,      1.56227,      1.58485,      1.62290,
7      1.65268,      1.66722,      1.68166,      1.71641,
8      1.73916,      1.73430,      1.77471/
 DATA (DSL1(I),I= 249, 310) /
1  41273.28000,  41548.43000,  41903.11000,  42334.01000,
2  43023.17000,  43406.44000,  43832.09000,  44359.67000,
3  44990.19000,  45558.22000,  46046.05000,  46528.00000,
4  47288.86000,  47748.13000,  48081.04000,  48435.24000,
5  48914.20000,  49436.57000,  49811.10000,  50176.03000,
6  50487.93000,  50798.96000,  51274.14000,  51816.01000,
7  52181.38000,  52564.07000,  52909.30000,  53646.82000,
8  54170.74000,  54606.49000,  55474.46000,
1      1.05027,      1.07010,      1.08786,      1.10356,
2      1.11240,      1.15113,      1.18680,      1.21455,
3      1.23628,      1.26945,      1.30310,      1.33593,
4      1.34893,      1.38192,      1.41769,      1.44952,
5      1.46949,      1.48654,      1.51152,      1.53339,
6      1.55945,      1.58723,      1.60823,      1.63229,
7      1.66936,      1.70619,      1.74995,      1.78736,
8      1.83904,      1.89252,      1.92784/
 DATA (DSL1(I),I= 311, 372) /
1  40647.10000,  40860.64000,  41017.94000,  41220.61000,
2  41359.77000,  42030.06000,  42453.51000,  42872.08000,
3  43275.20000,  43792.22000,  44039.68000,  44394.22000,
4  44565.28000,  45100.76000,  45281.17000,  45558.84000,
5  45560.02000,  45947.59000,  46236.53000,  46324.43000,
6  46564.93000,  46754.78000,  51947.74000,  47340.84000,
7  47626.79000,  47687.70000,  47897.92000,  48272.75000,
8  48596.14000,  48536.89000,  49041.06000,
1      1.08158,      1.10449,      1.13212,      1.15923,
2      1.19577,      1.21995,      1.25533,      1.28893,
3      1.32283,      1.35775,      1.40342,      1.44262,
4      1.48466,      1.51384,      1.55734,      1.59334,
5      1.63720,      1.66099,      1.69025,      1.72597,
6      1.75560,      1.78939,      1.82455,      1.85675,
7      1.89709,      1.95091,      2.00052,      2.05605,
8      2.11777,      2.19687,      2.24951/
 DATA (DSL1(I),I= 373, 434) /
1  40152.36000,  40317.31000,  40500.50000,  40453.59000,
2  40623.54000,  41069.34000,  41375.69000,  41401.41000,
3  41324.80000,  41746.22000,  42149.34000,  36036.29000,
4  42060.19000,  42512.21000,  43044.14000,  43322.28000,
5  43422.59000,  43234.41000,  43725.66000,  43990.70000,
6  44232.39000,  44099.66000,  43693.03000,  43739.09000,
7  44098.57000,  44283.32000,  44313.07000,  45112.38000,
8  45864.46000,  46576.00000,  46686.90000,
1      1.10282,      1.12713,      1.15368,      1.19119,
2      1.22618,      1.25998,      1.30024,      1.35025,
3      1.40343,      1.44300,      1.48231,      1.53503,
4      1.58904,      1.62178,      1.65055,      1.68653,
5      1.72626,      1.77404,      1.79487,      1.82196,
6      1.85279,      1.90002,      1.96098,      2.00624,
```

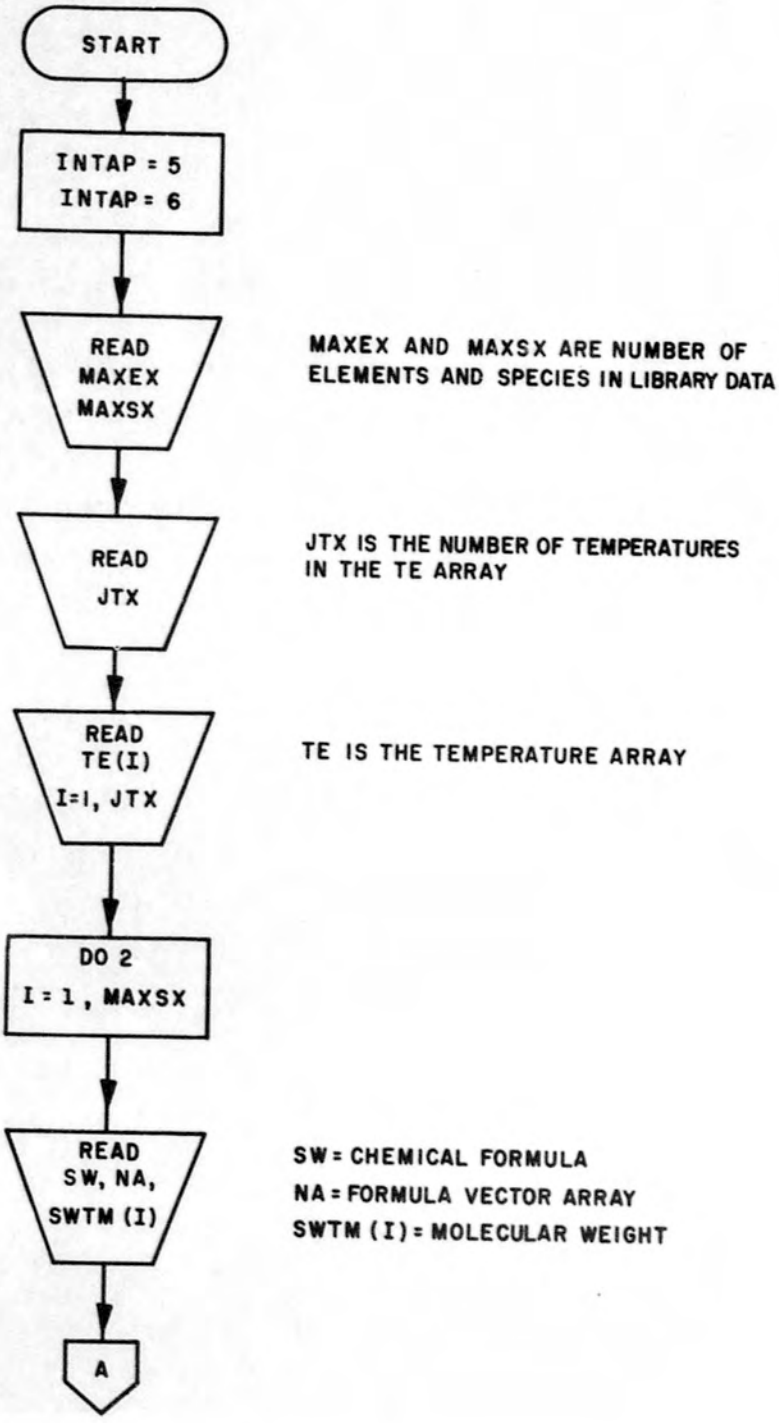
```
7       2.04410,       2.09206,       2.14989,       2.18773,
8       2.23159,       2.27785,       2.34760/
 DATA (DSL1(I),I= 435, 496) /
1  39893.51000,  40093.05000,  40238.43000,  40534.51000,
2  40454.50000,  40606.52000,  41006.40000,  41286.20000,
3  41250.34000,  41465.94000,  41776.62000,  42171.19000,
4  42354.21000,  42523.26000,  42818.48000,  43078.12000,
5  43690.83000,  44121.73000,  43916.90000,  43890.18000,
6  44229.59000,  44815.90000,  45792.50000,  46197.89000,
7  46551.40000,  47412.60000,  48516.44000,  48799.40000,
8  49234.84000,  50274.24000,  51408.54000,
1       1.11153,       1.13514,       1.16304,       1.18830,
2       1.23216,       1.27651,       1.31343,       1.35483,
3       1.40609,       1.45301,       1.49555,       1.53526,
4       1.57854,       1.62135,       1.65861,       1.69525,
5       1.71668,       1.74235,       1.78804,       1.82555,
6       1.85289,       1.87444,       1.88530,       1.91849,
7       1.95650,       1.98090,       1.99977,       2.05605,
8       2.11172,       2.14577,       2.17897/
 END
```

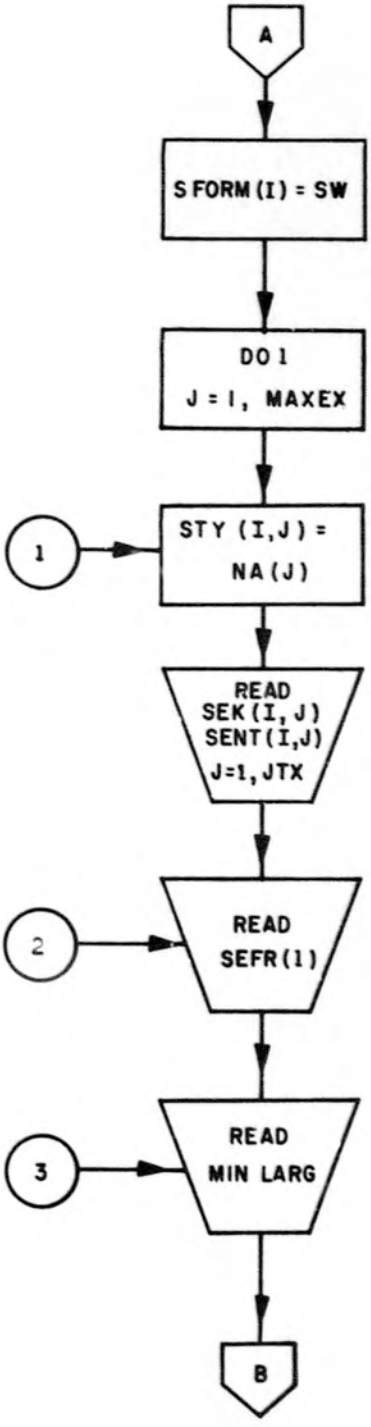
9. SEGMENT SIX

a. Function

This segment performs the chemical equilibrium calculations necessary to establish the chemical species present in the boundary layer at a given pressure, enthalpy, and mass fraction. The basic theory and calculation procedure has been adapted from the chemical equilibrium and normal shock program developed by JPL (Ref. 9). This basic program was modified extensively and a detailed description of the current version is contained in reference 2. Only minor changes were made to the program of reference 2 to obtain the version used in this calculation. In this case, the "initial condition group," or the elemental composition of the gaseous mixture, is generated in Segment ONE and transmitted to Segment SIX by a file named TAPE 22. The calculation option (which is always 90 for this program), pressure, and enthalpy are also transmitted via this file from Segment ONE. When the composition for all 50 boundary layer points has been calculated, control is given to Segment SEVEN.

b. Flow Chart

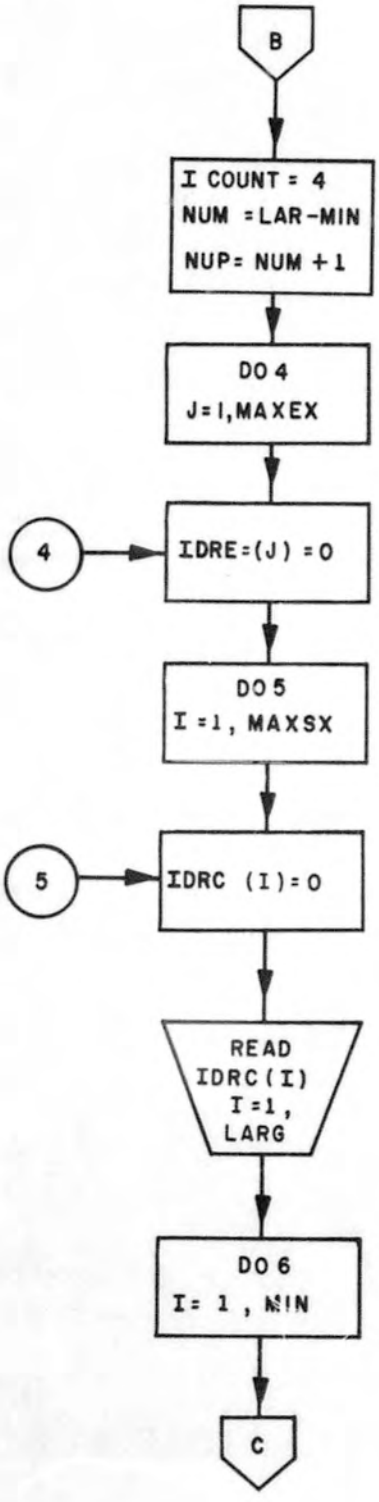

START
INTAP = 5
INTAP = 6
READ
MAXEX
MAXSX
MAXEX AND MAXSX ARE NUMBER OF ELEMENTS AND SPECIES IN LIBRARY DATA
READ
JTX
JTX IS THE NUMBER OF TEMPERATURES IN THE TE ARRAY
READ
TE(I)
I=1, JTX
TE IS THE TEMPERATURE ARRAY
DO 2
I = 1 , MAXSX
READ
SW, NA,
SWTM (I)
SW = CHEMICAL FORMULA
NA = FORMULA VECTOR ARRAY
SWTM (I) = MOLECULAR WEIGHT
A



SEK IS THE FREE ENERGY ARRAY
SENT IS THE ENTHALPY ARRAY

SEFR IS THE ENERGY OF FORMATION
ALL LIBRARY DATA HAS BEEN READ IN

MIN IS THE NUMBER OF ELEMENTS IN MODEL
LARG IS THE NUMBER OF SPECIES IN MODEL



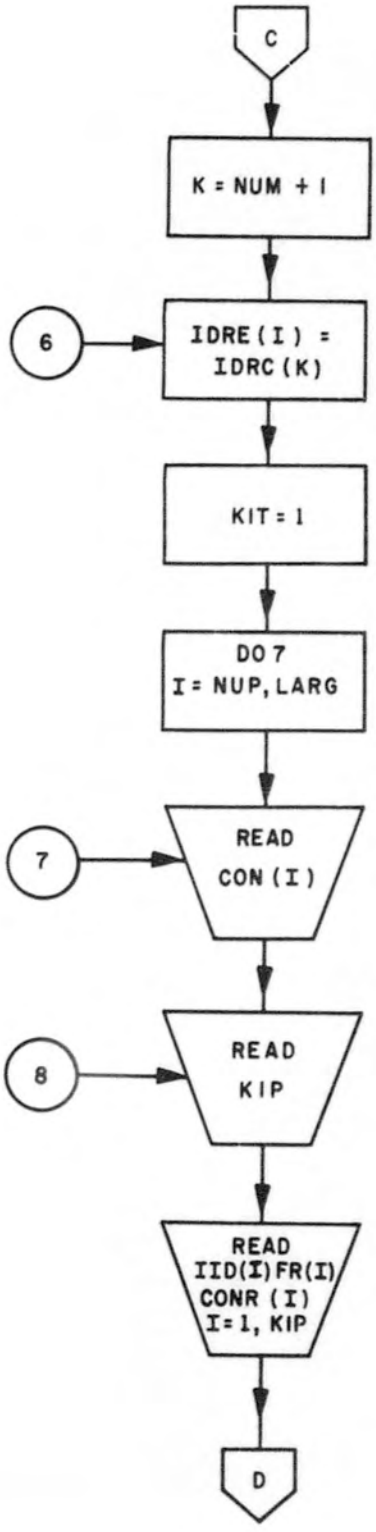
INITIALIZATION OF I COUNT TRIGGER NUM IS NUMBER OF SPECIES WHICH ARE NOT ELEMENTS

ZERO OUT I DRE ARRAY

ZERO OUT IDRC ARRAY

THIS ARRAY CONTAINS THE IDENTIFICATION NUMBERS FOR THE SPECIES IN THE MIXTURE

THIS DO LOOP PICKS UP THE IDENTIFICATION NUMBERS FOR THE SPECIES FROM THE IDRC ARRAY AND PLACES THEM IN THE IDRE ARRAY



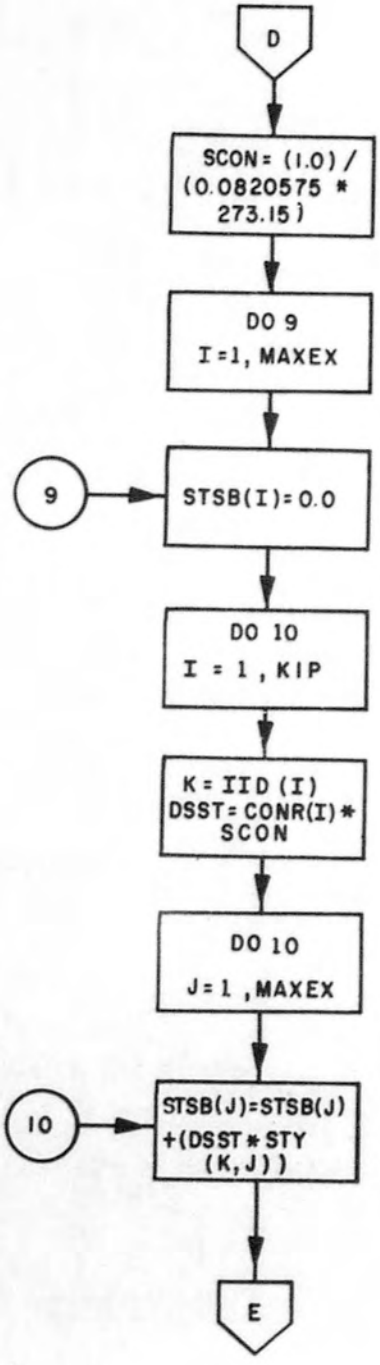
READ INITIAL CONCENTRATIONS OF ELEMENTAL SPECIES

KIP IS NUMBER OF COMPOUNDS PRESENT IN MIXTURE AT STANDARD CONDITIONS

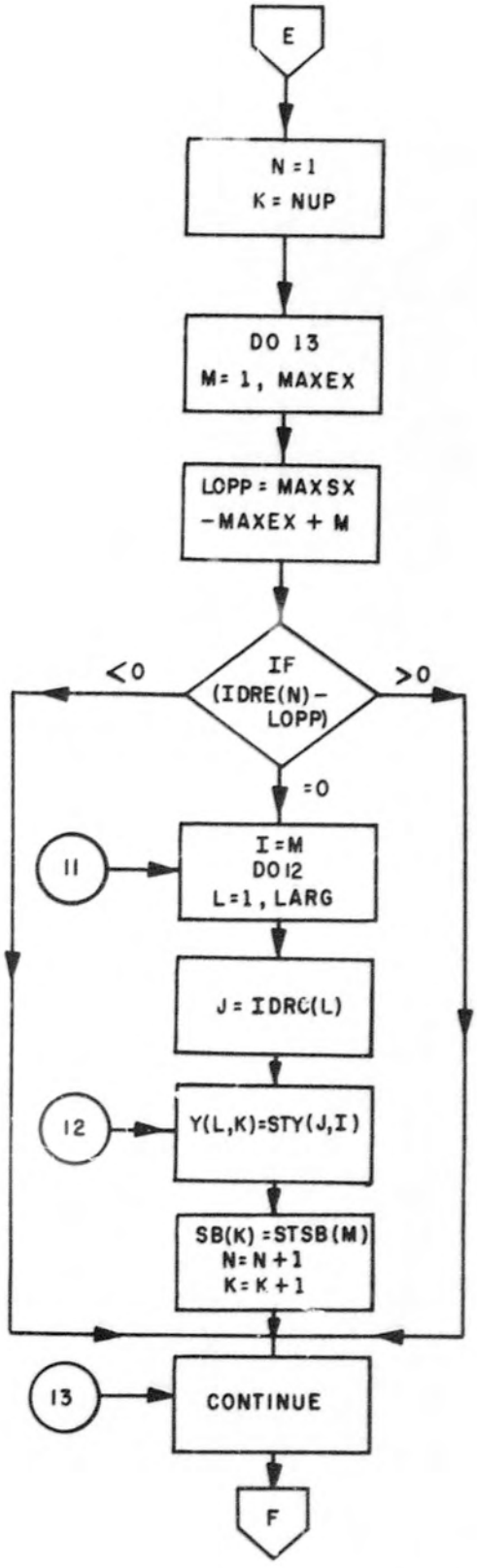
IID ARRAY IS SPECIES IDENTIFICATION NUMBERS

FR ARRAY IS SPECIES CHEMICAL SYMBOLS

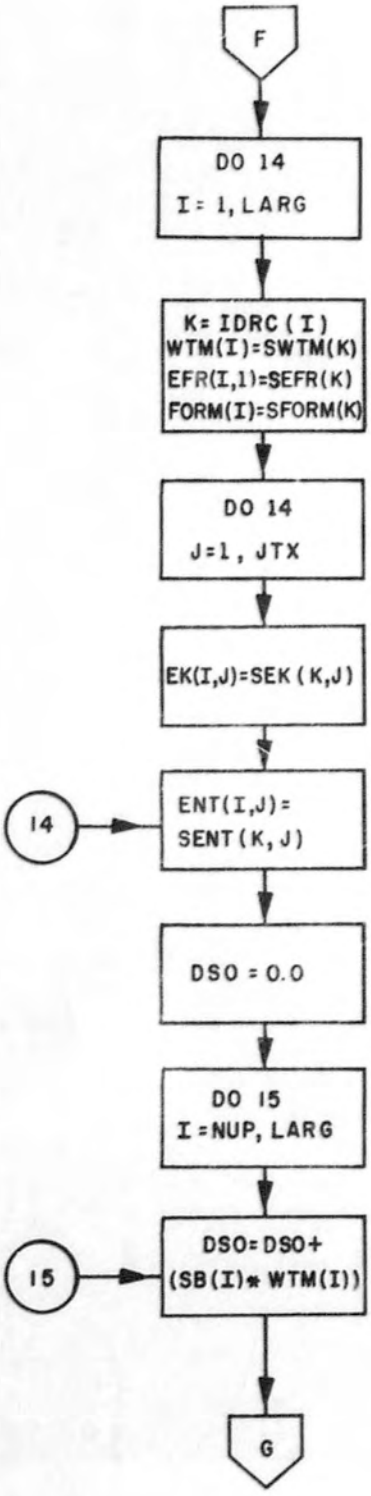
CONR ARRAY IS MOLEFRACTIONS AT STP



ZERO OUT STSB ARRAY



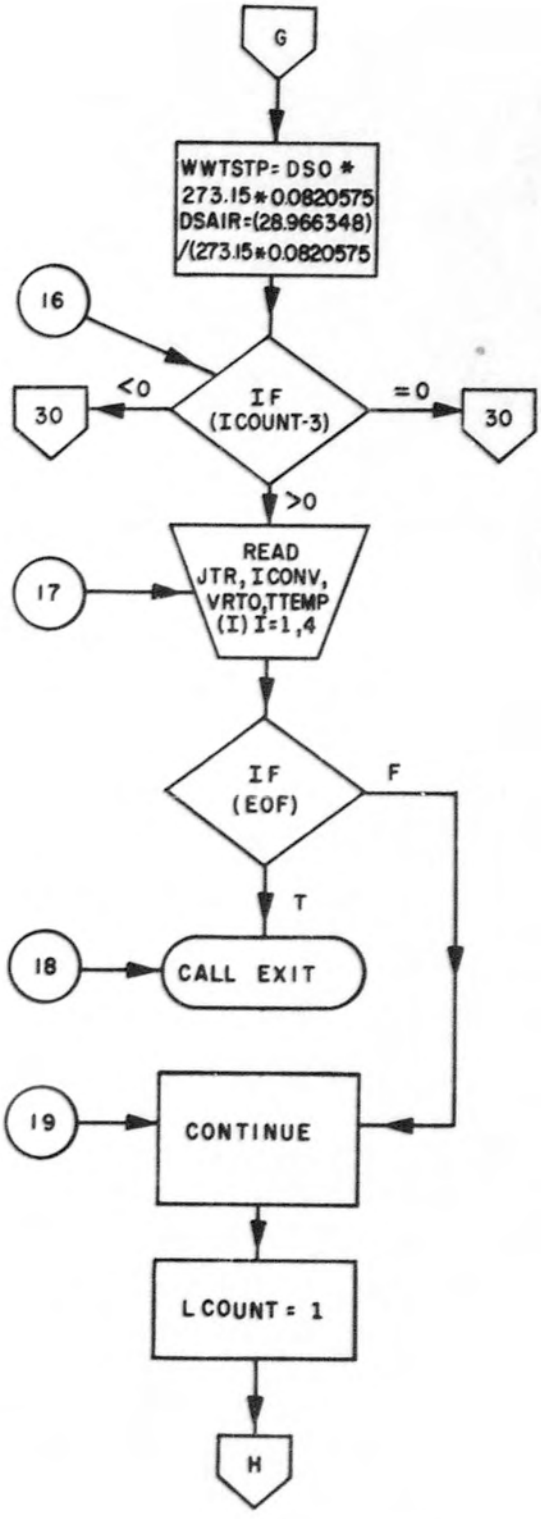
SET UP THE CORRECT FORMULA VECTORS FOR THE CASE

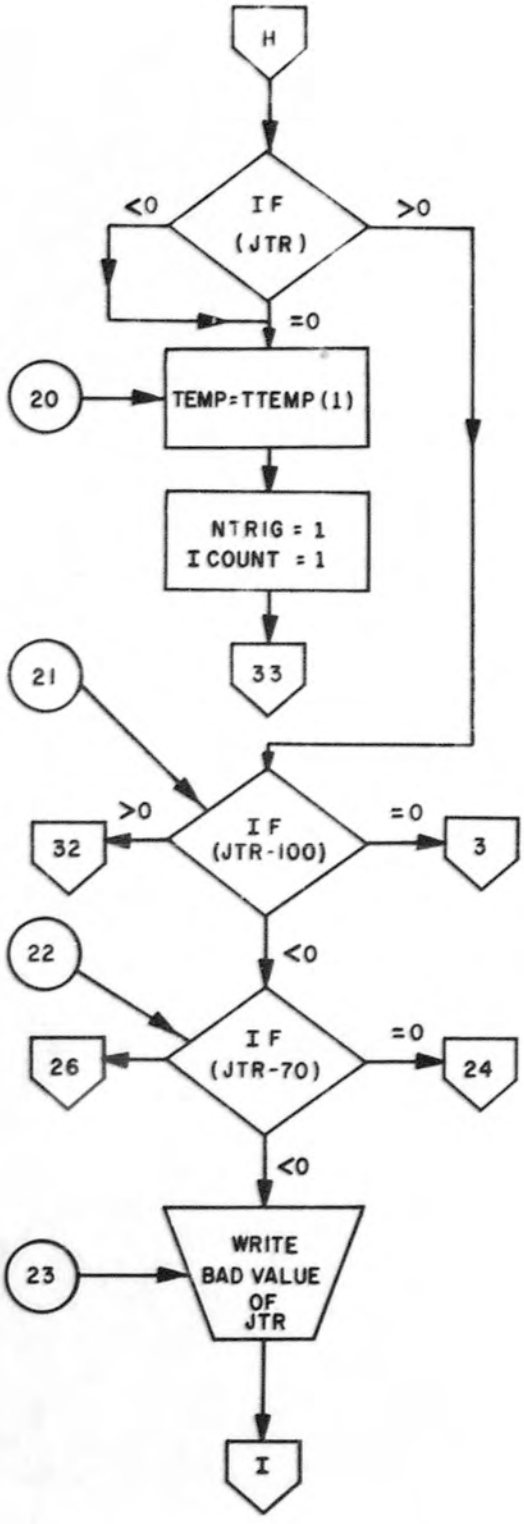


SET UP THE MOLECULAR WEIGHT, ENERGY OF FORMATION, AND CHEMICAL SYMBOL ARRAYS FOR THIS CASE

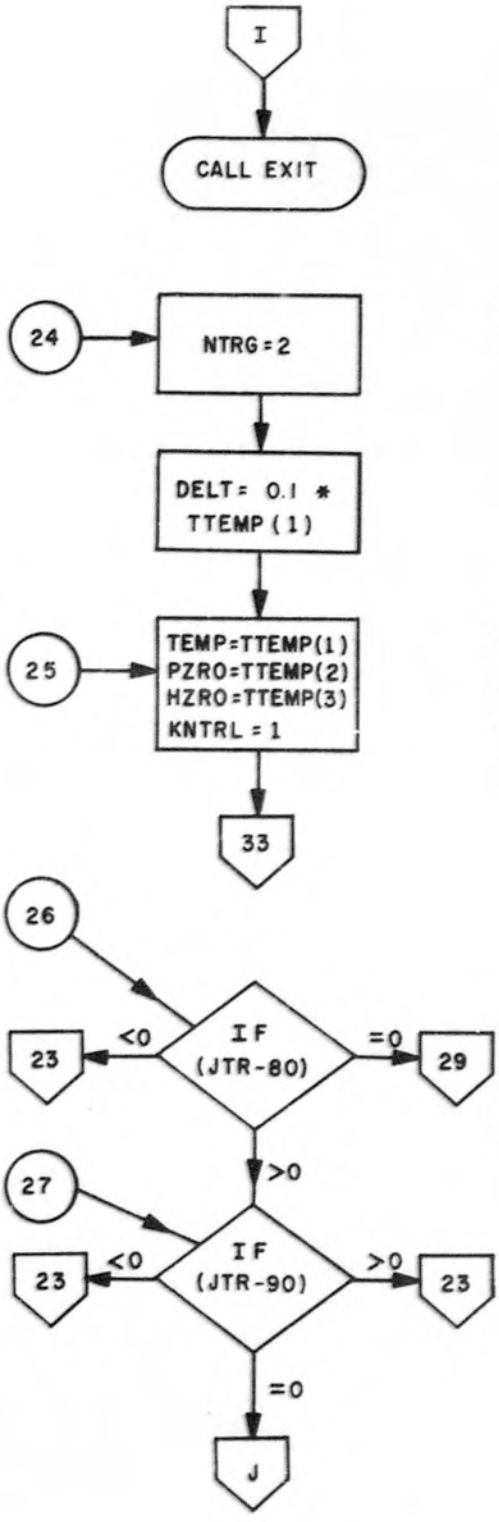
SET UP FREE ENERGY ARRAY FOR THIS CASE

SET UP ENTHALPY ARRAY FOR THIS CASE


G
WWTSTP= DSO *
273.15*0.0820575
DSAIR=(28.966348)
/(273.15*0.0820575
CALCULATE THE DENSITY OF AIR
16
IF
(ICOUNT-3)
<0
=0
>0
30
30
READ
JTR, ICONV,
VRTO, TTEMP
(I) I=1,4
17
THIS CARD SPECIFIES THE THERMO DYNAMIC CONDITIONS FOR A CASE
IF
(EOF)
F
T
18
CALL EXIT
19
CONTINUE
LCOUNT = 1
H



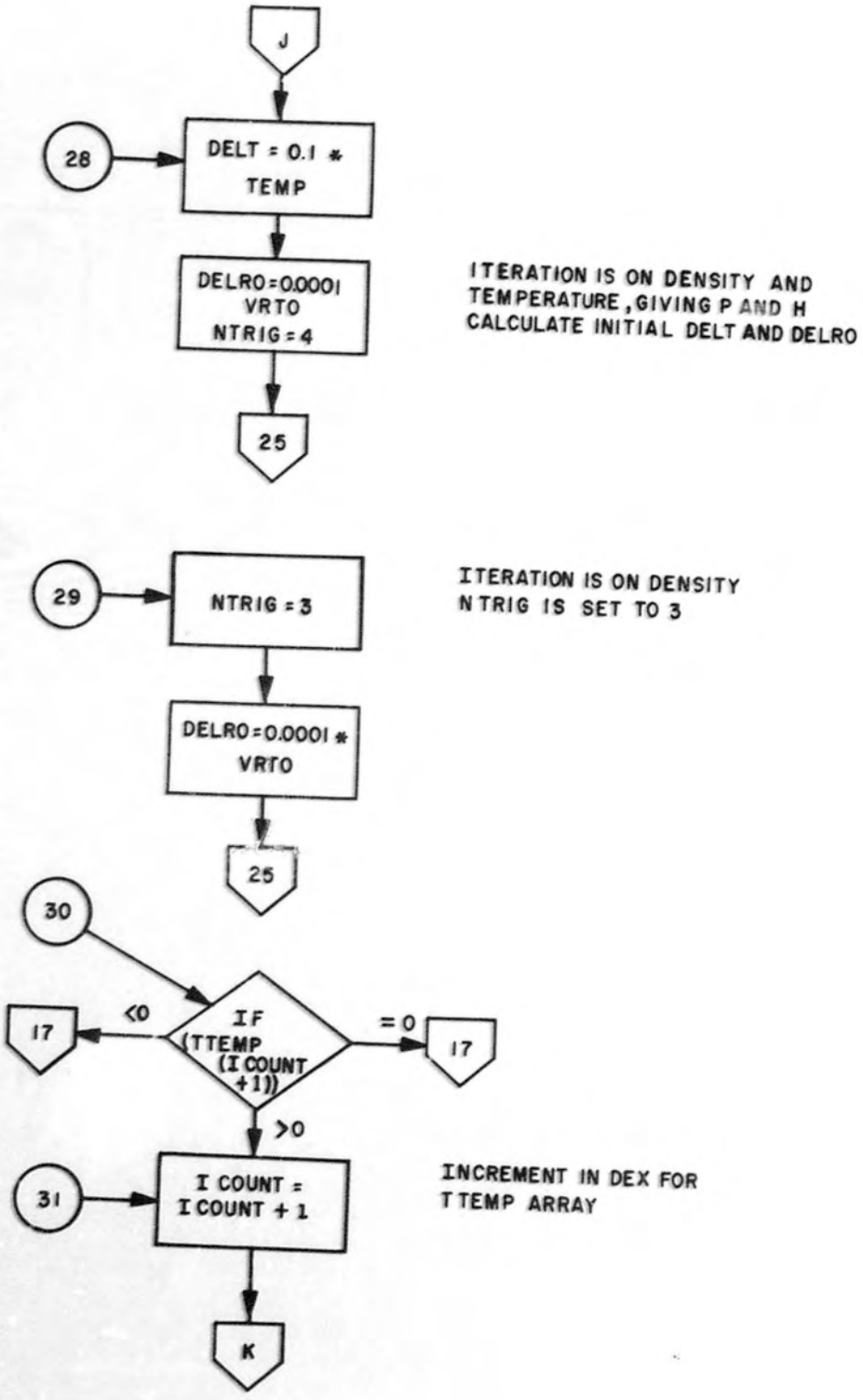
IF A BAD JTR VALVE IS USED AS INPUT, THE RUN IS TERMINATED

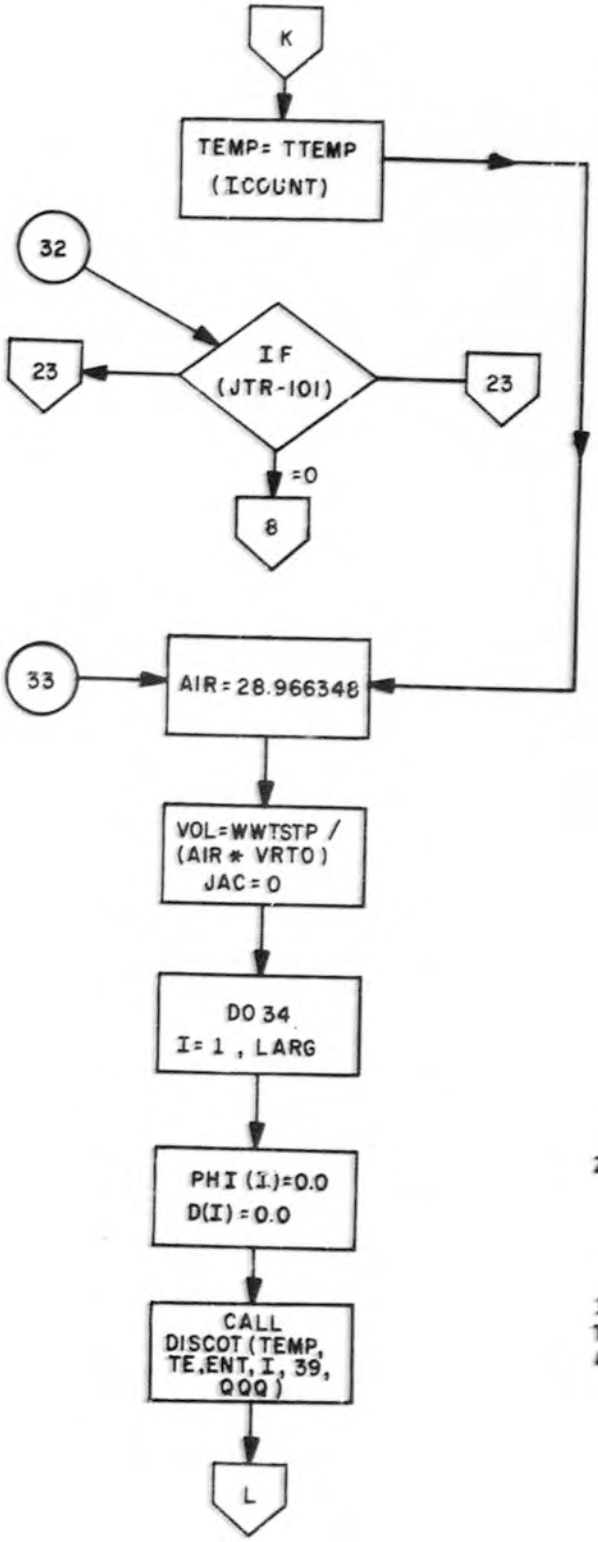


INTERATION IS ON T
NTRIG IS SET TO 2

INITIAL TEMP VALUE SET
TO TTEMP (1)
HZRO IS THE RATIO OF
DENSITY AT SOLUTION POINT
TO DENSITY AT STP

TEST FOR TYPE OF ITERATION

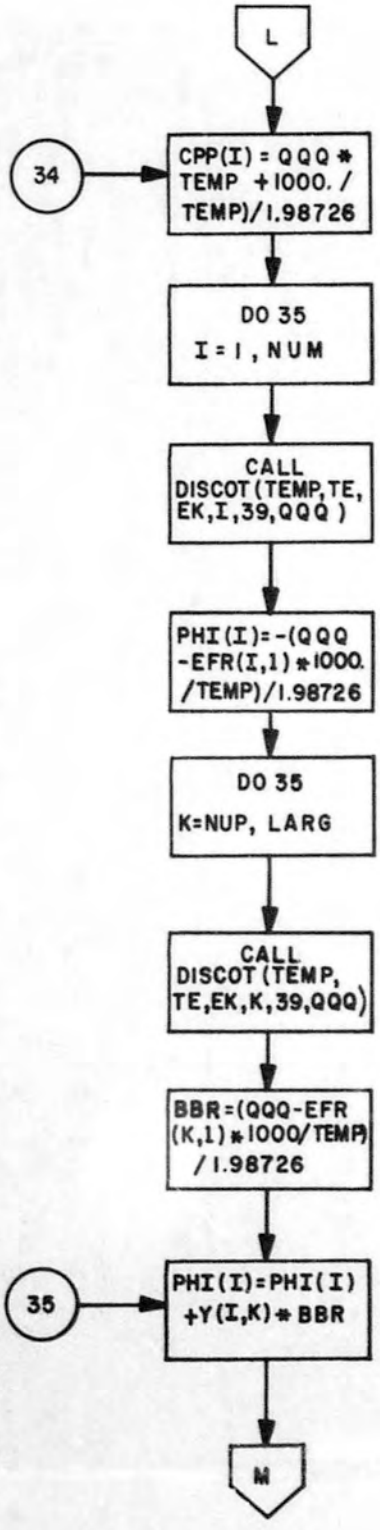

J
28
DELT = 0.1 * TEMP
DELRO=0.0001 VRTO
NTRIG=4
25
ITERATION IS ON DENSITY AND TEMPERATURE, GIVING P AND H
CALCULATE INITIAL DELT AND DELRO
29
NTRIG=3
DELRO=0.0001 * VRTO
25
ITERATION IS ON DENSITY
NTRIG IS SET TO 3
30
IF (TTEMP (I COUNT +1))
<0
17
=0
17
>0
31
I COUNT = I COUNT + 1
K
INCREMENT IN DEX FOR TTEMP ARRAY



SET TEMP TO NEXT VALUE IN TTEMP ARRAY

ZERO OUT PHI AND D ARRAYS

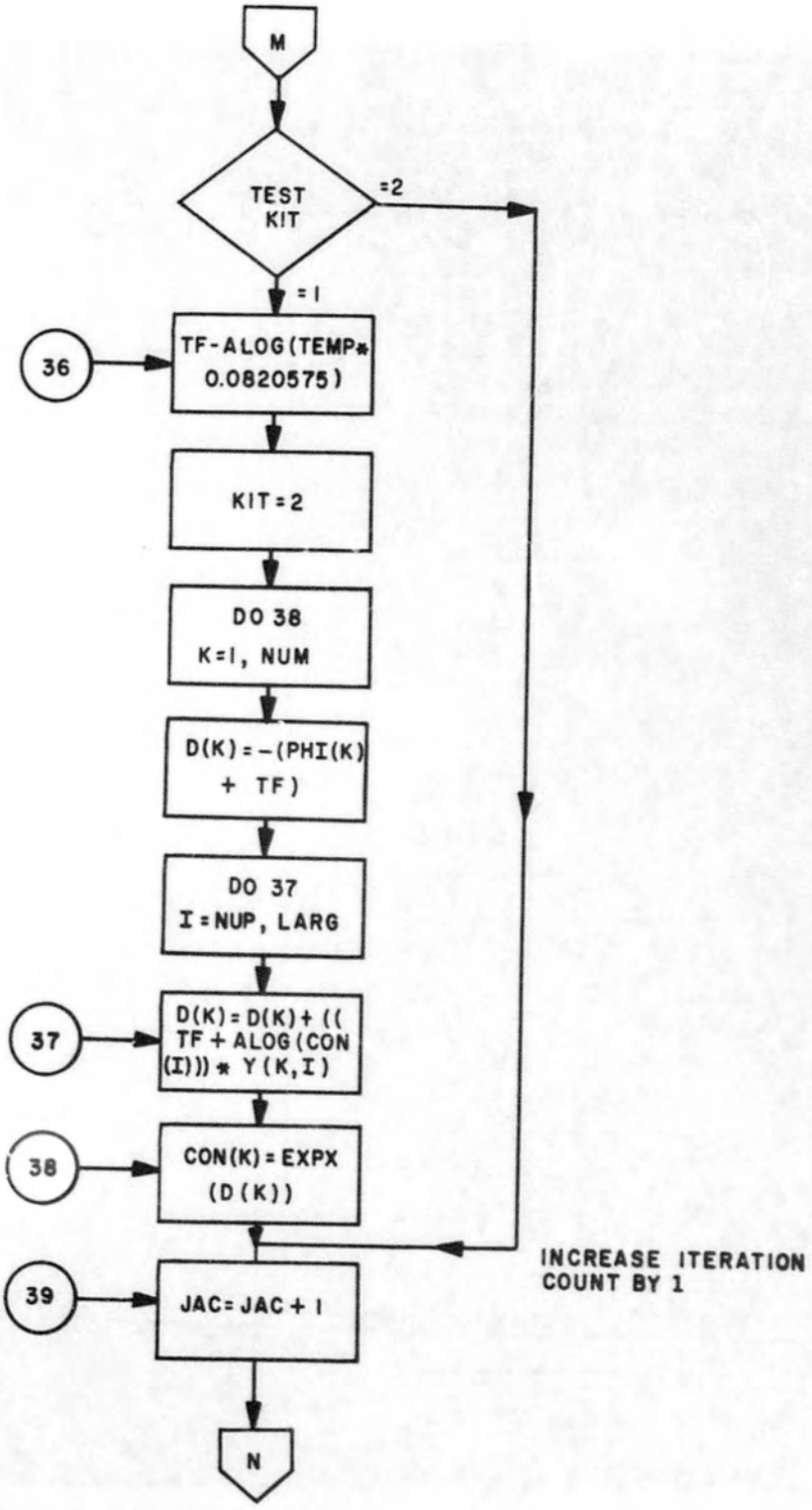
INTERPOLATE LINEARLY TO FIND THE ENTHALPY, QQQ, FROM THE ENT ARRAY AT TEMPERATURE TEMP

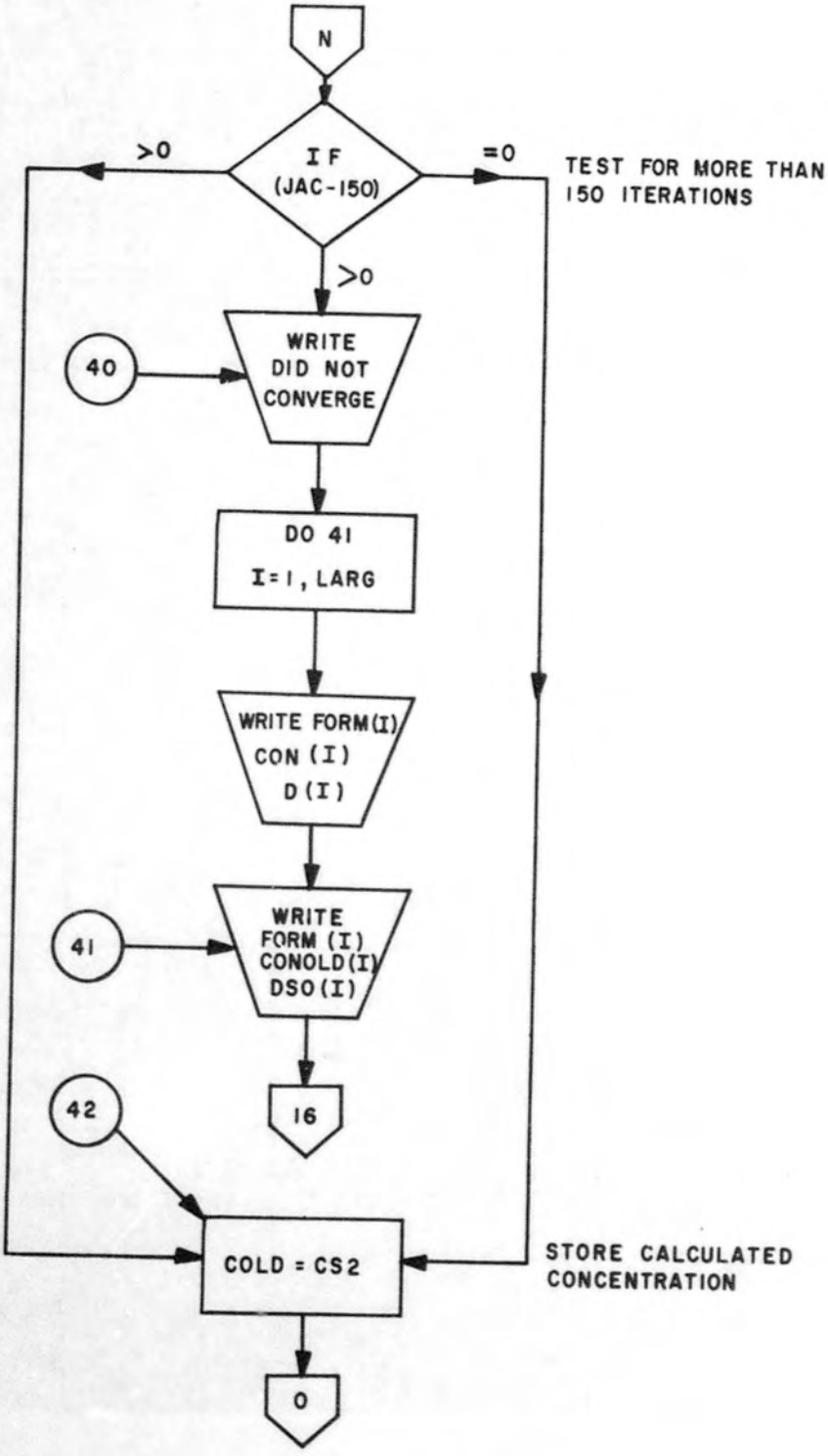


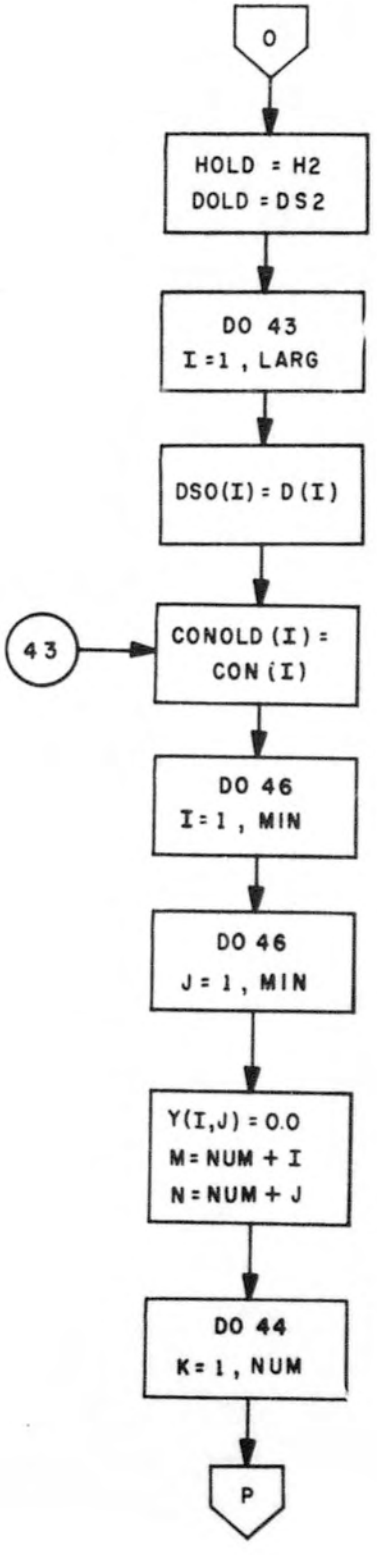
CALCULATE ENTHALPY IN CAL / MOL E

INTERPOLATE LINEARLY TO FIND THE FREE ENERGY, QQQ, FROM THE EK ARRAY AT TEMPERATURE TEMP

INTERPOLATE LINEARLY TO FIND THE FREE ENERGY, QQQ, FROM THE EK ARRAY AT TEMPERATURE TEMP


M
TEST KIT
=2
=1
36
TF-ALOG(TEMP* 0.0820575)
KIT=2
DO 38 K=1, NUM
D(K)=-(PHI(K) + TF)
DO 37 I=NUP, LARG
37
D(K)=D(K)+((TF+ALOG(CON (I)))* Y(K,I)
38
CON(K)=EXPX (D(K))
INCREASE ITERATION COUNT BY 1
39
JAC=JAC+1
N


N
IF
(JAC-150)
>0
=0
TEST FOR MORE THAN
150 ITERATIONS
>0
40
WRITE
DID NOT
CONVERGE
DO 41
I=1, LARG
WRITE FORM(I)
CON (I)
D(I)
41
WRITE
FORM (I)
CONOLD(I)
DSO(I)
16
42
COLD = CS2
STORE CALCULATED
CONCENTRATION
0

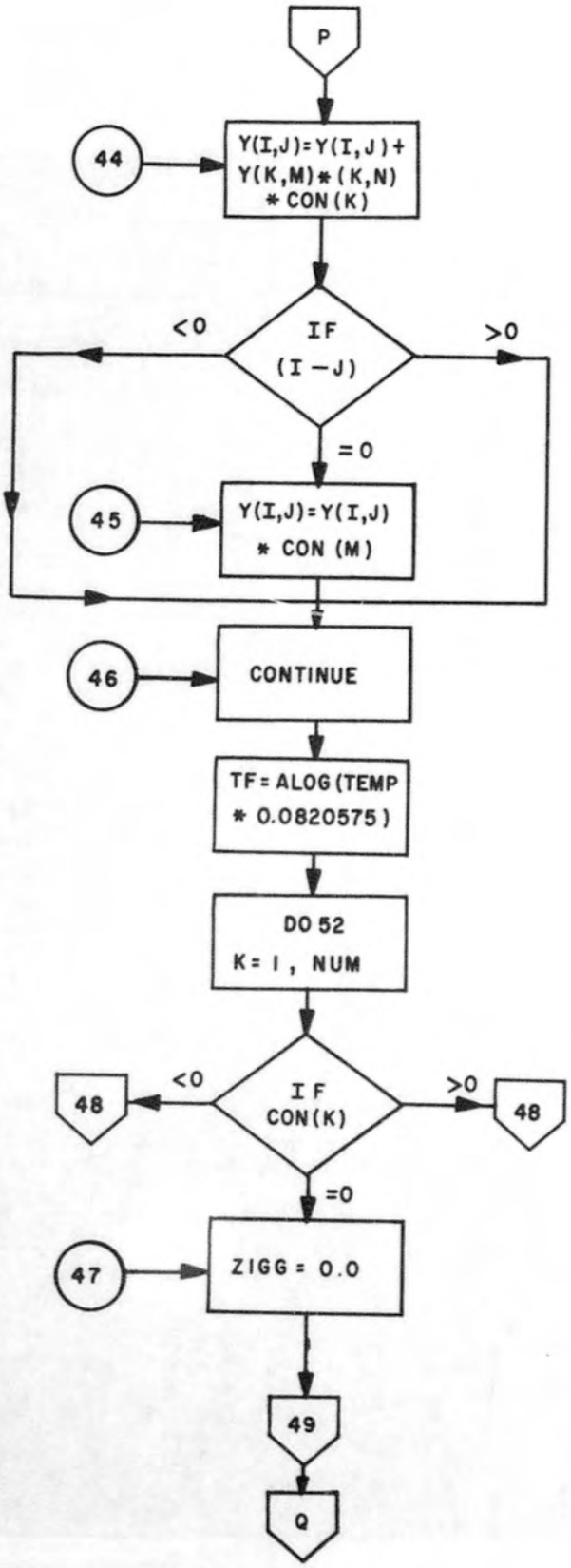


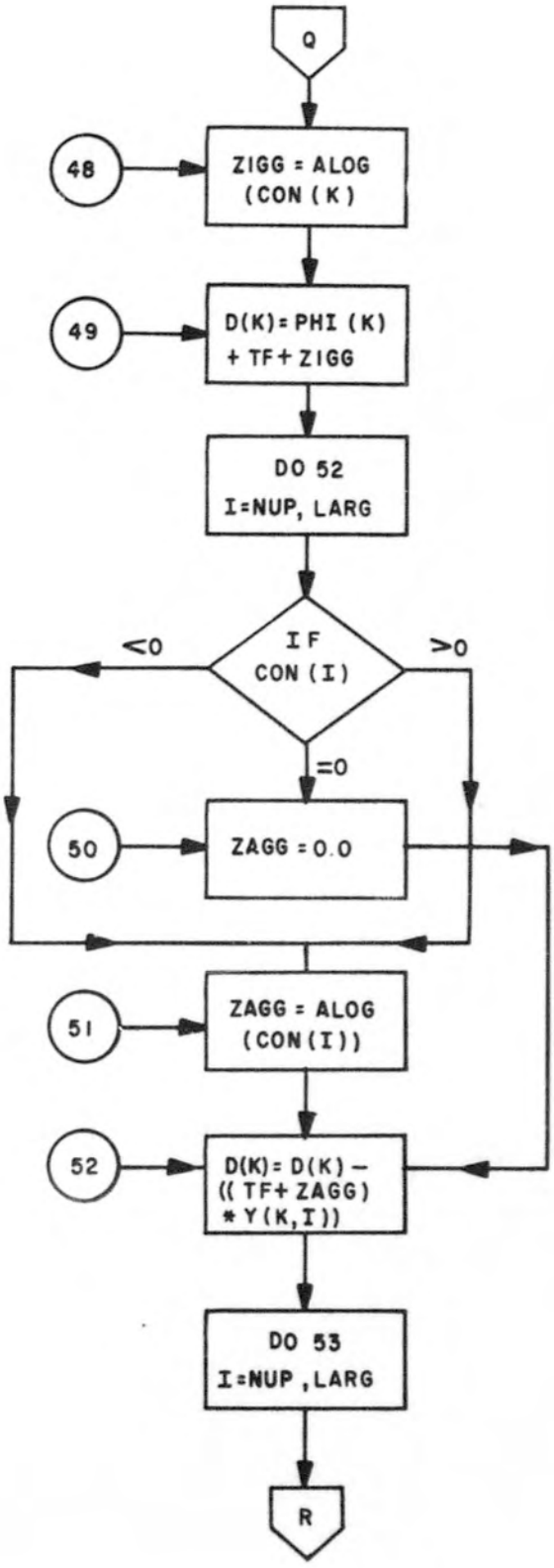
STORE CALCULATED ENTHALPY AND DENSITY

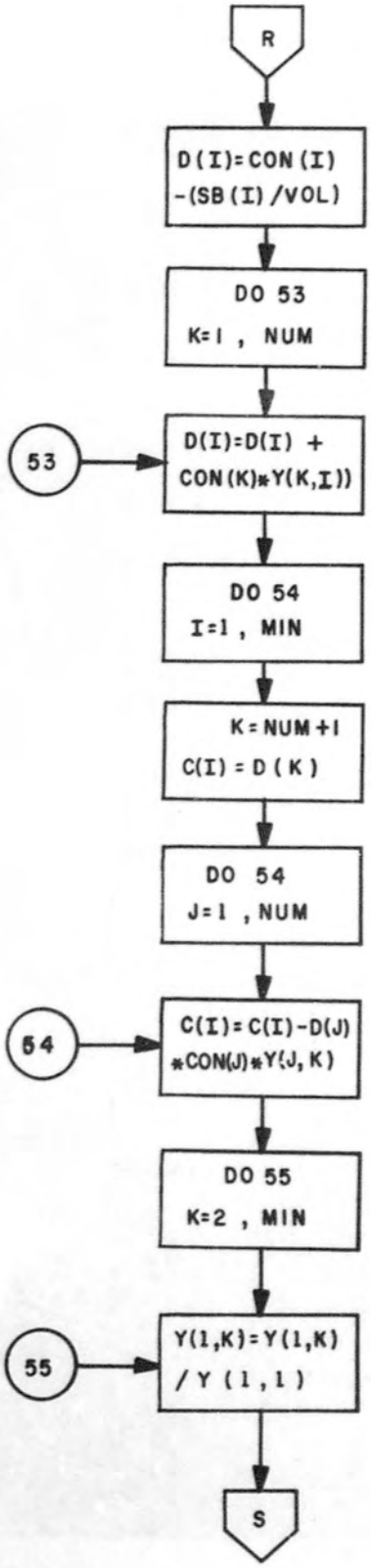
BEGINNING OF SOLUTION OF THE EQUATION

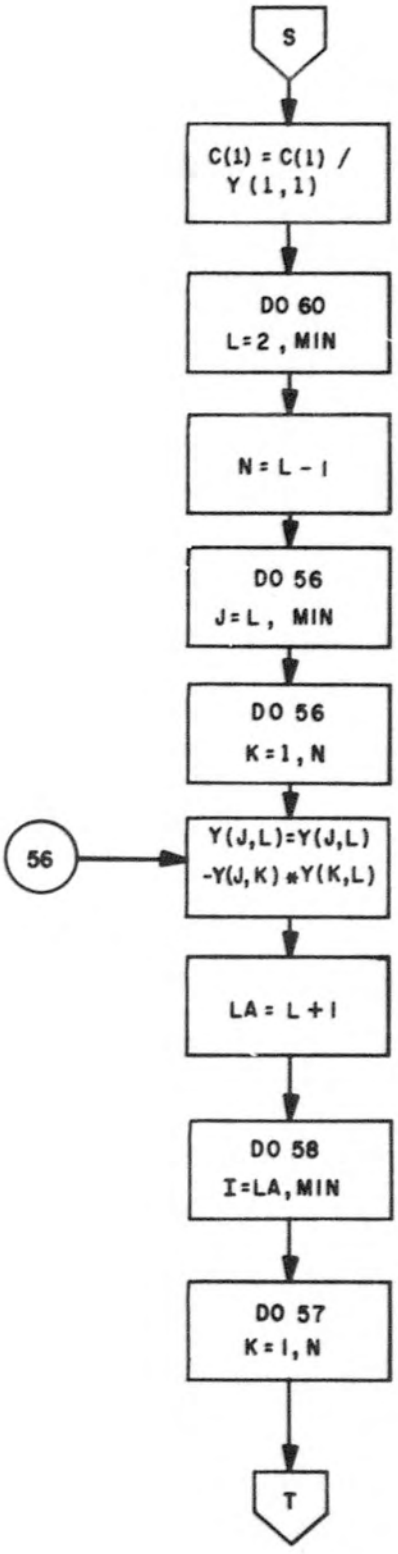
$$\sum_{J=1}^{m} (Y_{ij})_p (Z_{s+j})_{p+1} = (C_i)_p$$

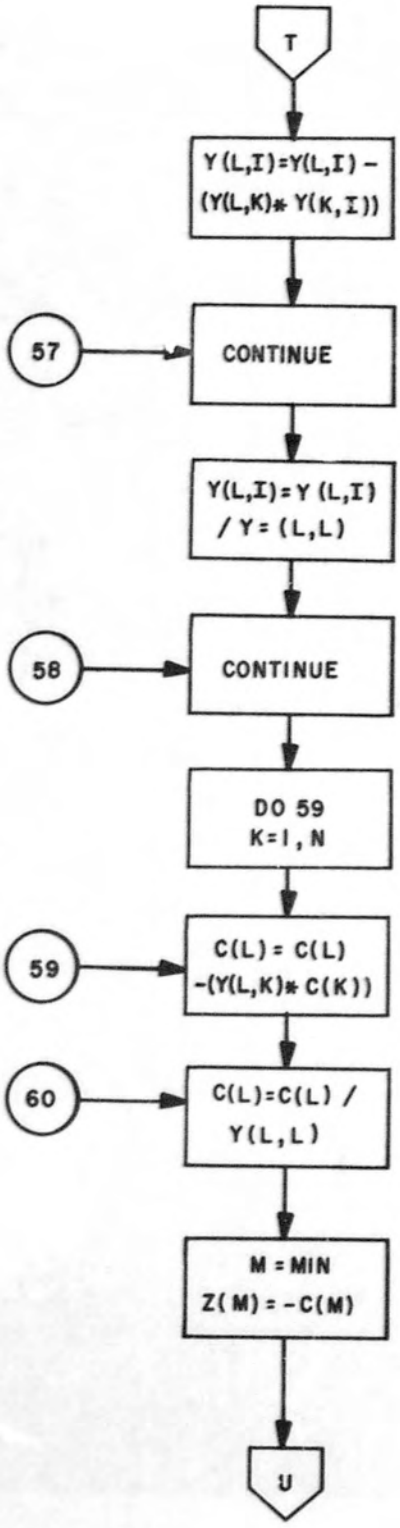
USING THE CROUT METHOD

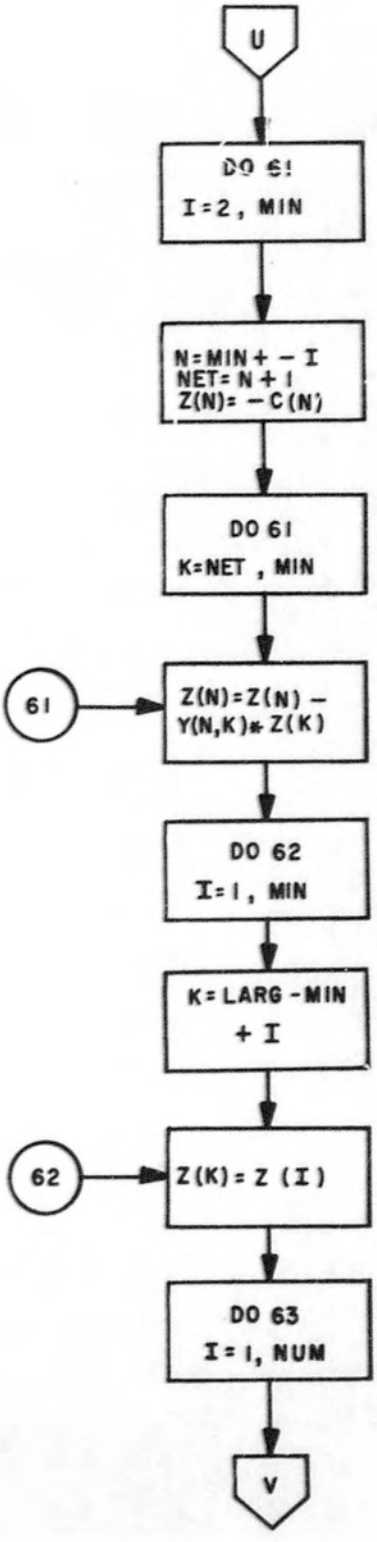

P
44
Y(I,J)=Y(I,J)+
Y(K,M)*(K,N)
*CON(K)
IF
(I-J)
<0
>0
=0
45
Y(I,J)=Y(I,J)
*CON(M)
46
CONTINUE
TF=ALOG(TEMP
*0.0820575)
DO 52
K=1, NUM
IF
CON(K)
<0
>0
48
48
=0
47
ZIGG=0.0
49
Q

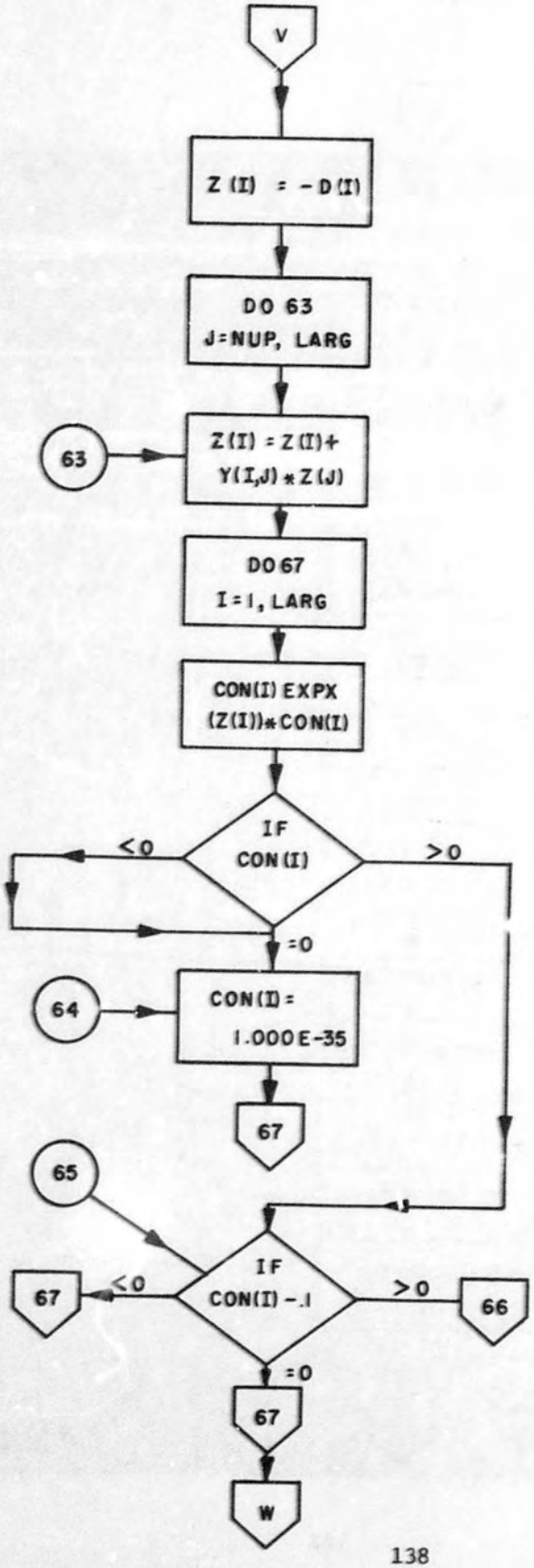

Q
48
ZIGG = ALOG (CON (K)
49
D(K)= PHI (K) + TF+ ZIGG
DO 52 I=NUP, LARG
IF CON (I)
<0
>0
=0
50
ZAGG = 0.0
51
ZAGG = ALOG (CON(I))
52
D(K)= D(K) – ((TF+ ZAGG) * Y(K,I))
DO 53 I=NUP , LARG
R


R
D(I)=CON(I)
-(SB(I)/VOL)
DO 53
K=1 , NUM
53
D(I)=D(I) +
CON(K)*Y(K,I))
DO 54
I=1 , MIN
K=NUM+1
C(I)=D(K)
DO 54
J=1 , NUM
54
C(I)=C(I)-D(J)
*CON(J)*Y(J,K)
DO 55
K=2 , MIN
55
Y(1,K)=Y(1,K)
/Y(1,1)
S

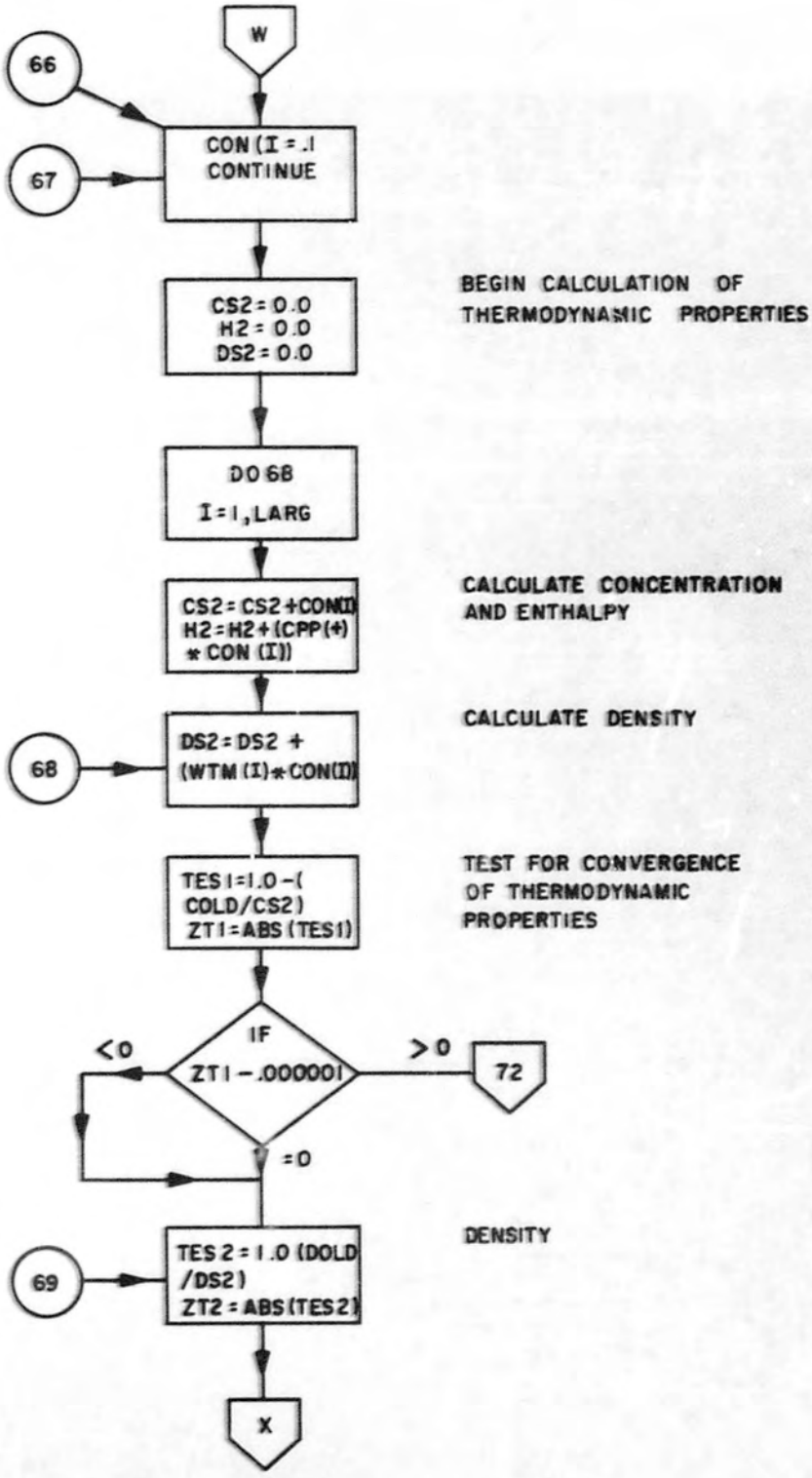

S
C(1) = C(1) / Y (1 , 1)
DO 60 L=2 , MIN
N = L - 1
DO 56 J=L , MIN
DO 56 K=1 , N
56
Y(J,L)=Y(J,L) -Y(J,K) *Y(K,L)
LA = L + 1
DO 58 I=LA, MIN
DO 57 K=1 , N
T

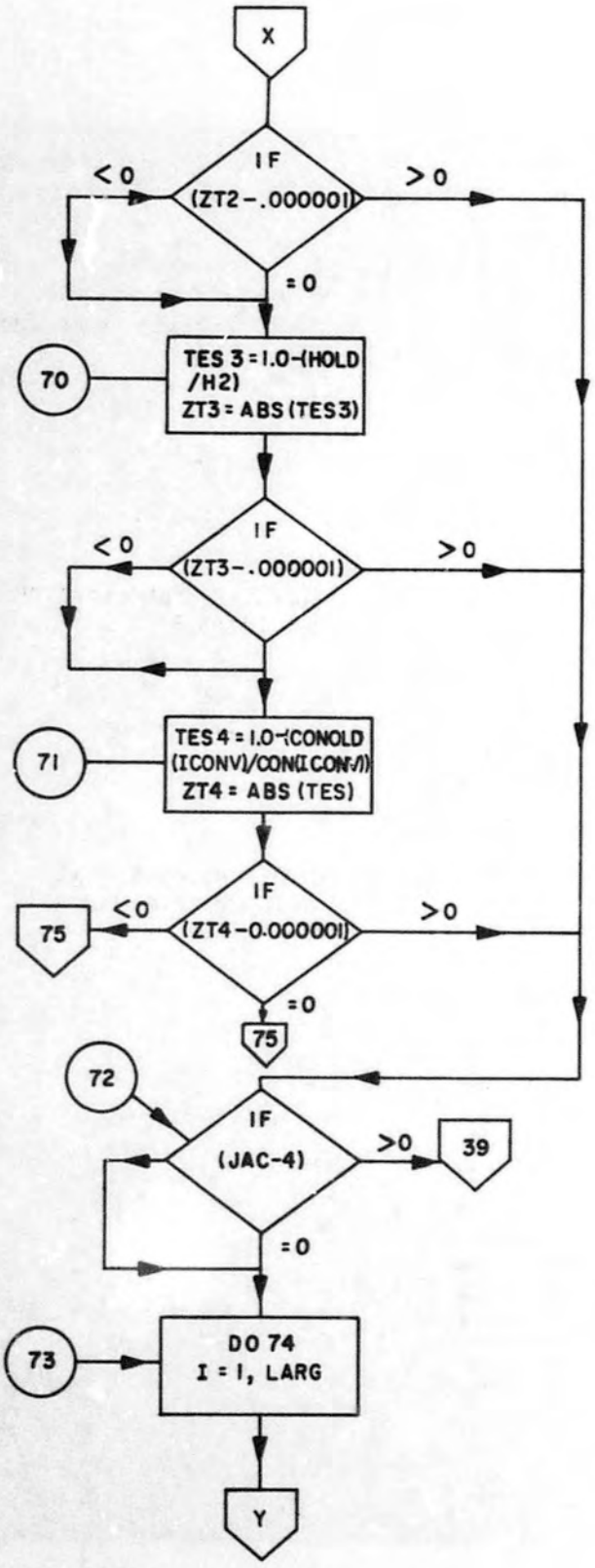

T
Y(L,I)=Y(L,I) -
(Y(L,K)* Y(K,I))
57
CONTINUE
Y(L,I)=Y (L,I)
/ Y= (L,L)
58
CONTINUE
DO 59
K=1, N
59
C(L) = C(L)
-(Y(L,K)* C(K))
60
C(L)=C(L) /
Y(L,L)
M = MIN
Z(M) = -C(M)
U

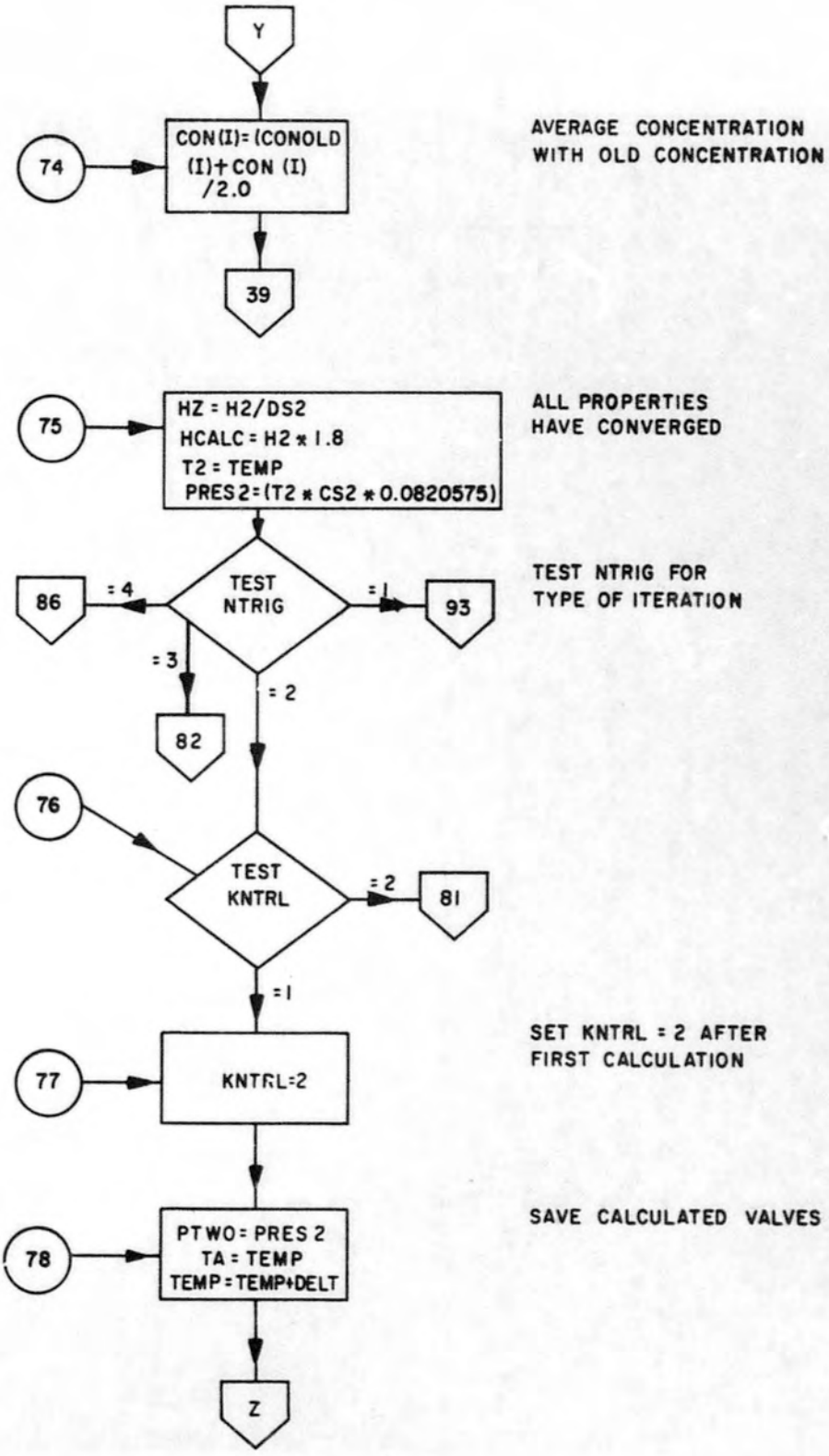

U
DO 61
I=2, MIN
N=MIN+-I
NET=N+1
Z(N)=-C(N)
DO 61
K=NET, MIN
61
Z(N)=Z(N)-
Y(N,K)*Z(K)
DO 62
I=1, MIN
K=LARG-MIN
+I
62
Z(K)=Z(I)
DO 63
I=1, NUM
V

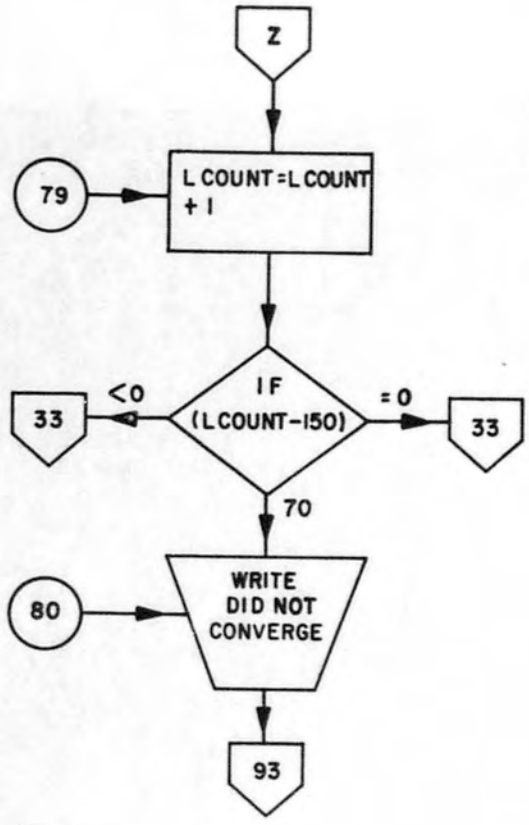


CONCENTRATIONS EQUAL TO ZERO ARE SET TO 10^{-35}


W
66
67
CON(I=.1
CONTINUE
CS2=0.0
H2=0.0
DS2=0.0
BEGIN CALCULATION OF THERMODYNAMIC PROPERTIES
DO 68
I=1,LARG
CS2=CS2+CON(I)
H2=H2+(CPP(I)
*CON(I))
CALCULATE CONCENTRATION AND ENTHALPY
68
DS2=DS2+
(WTM(I)*CON(I))
CALCULATE DENSITY
TES1=1.0-(
COLD/CS2)
ZT1=ABS(TES1)
TEST FOR CONVERGENCE OF THERMODYNAMIC PROPERTIES
IF
ZT1-.000001
<0
>0
72
=0
69
TES2=1.0(DOLD
/DS2)
ZT2=ABS(TES2)
DENSITY
X

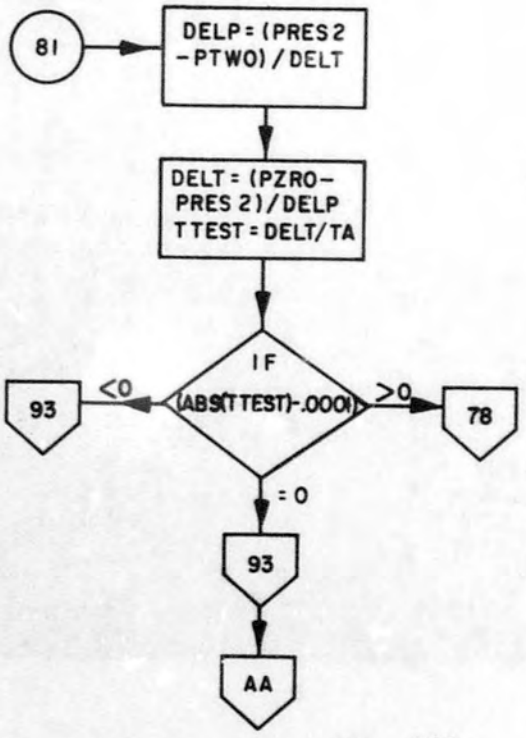

X
IF
(ZT2-.000001)
<0
>0
=0
70
TES 3 = 1.0-(HOLD /H2)
ZT3 = ABS (TES3)
ENTHALPY
IF
(ZT3-.000001)
<0
>0
71
TES 4 = 1.0-(CONOLD (ICONV)/CONI(ICONV))
ZT4 = ABS (TES)
CONCENTRATION
IF
(ZT4-0.000001)
<0
>0
=0
75
75
72
IF
(JAC-4)
>0
39
=0
73
DO 74
I = 1, LARG
Y


Y
74
CON(I)=(CONOLD (I)+CON (I) /2.0
AVERAGE CONCENTRATION WITH OLD CONCENTRATION
39
75
HZ = H2/DS2
HCALC = H2 * 1.8
T2 = TEMP
PRES2=(T2 * CS2 * 0.0820575)
ALL PROPERTIES HAVE CONVERGED
TEST NTRIG
TEST NTRIG FOR TYPE OF ITERATION
86
=4
=1
93
=3
=2
82
76
TEST KNTRL
=2
81
=1
77
KNTRL=2
SET KNTRL = 2 AFTER FIRST CALCULATION
78
PTWO=PRES 2
TA = TEMP
TEMP=TEMP+DELT
SAVE CALCULATED VALVES
Z

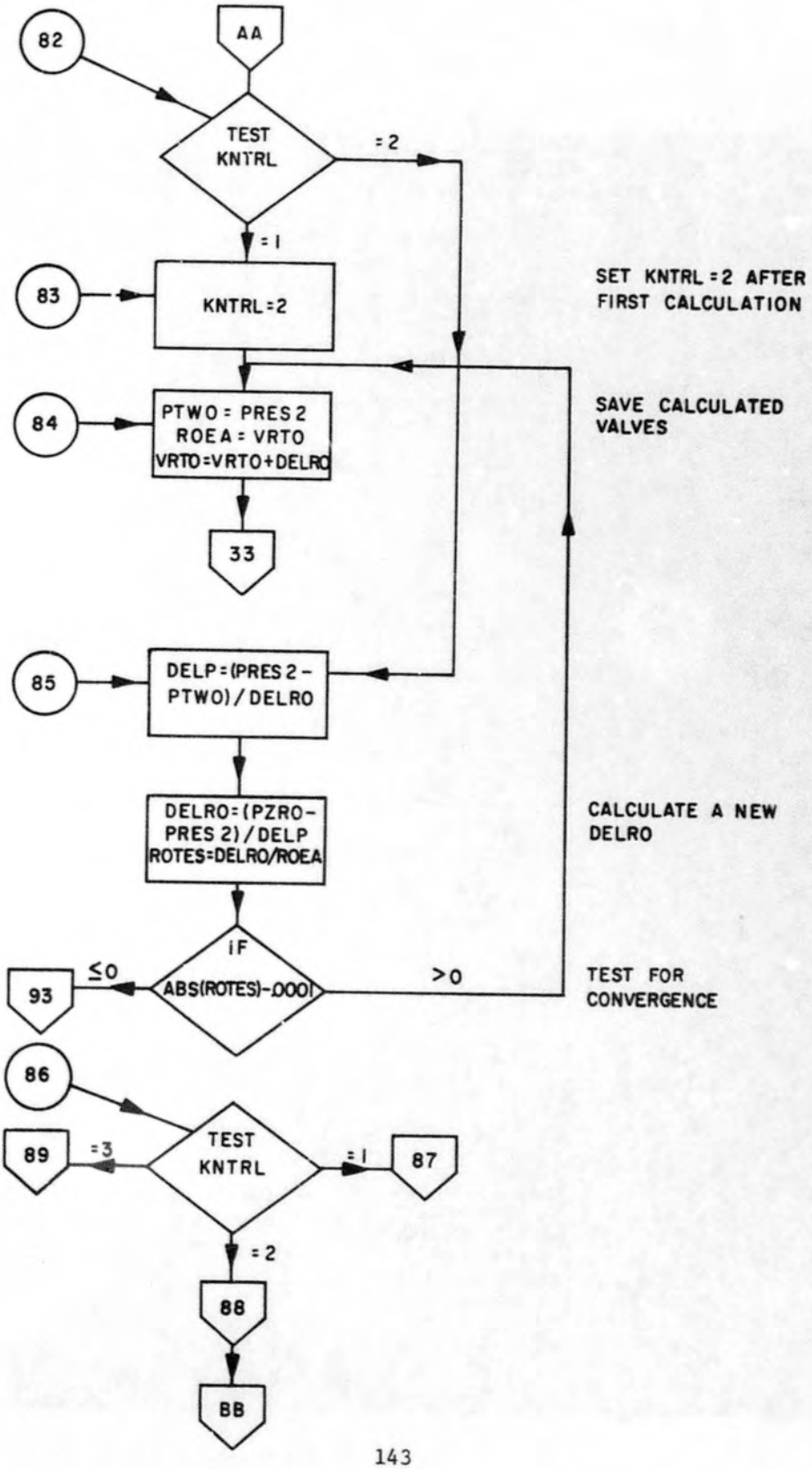


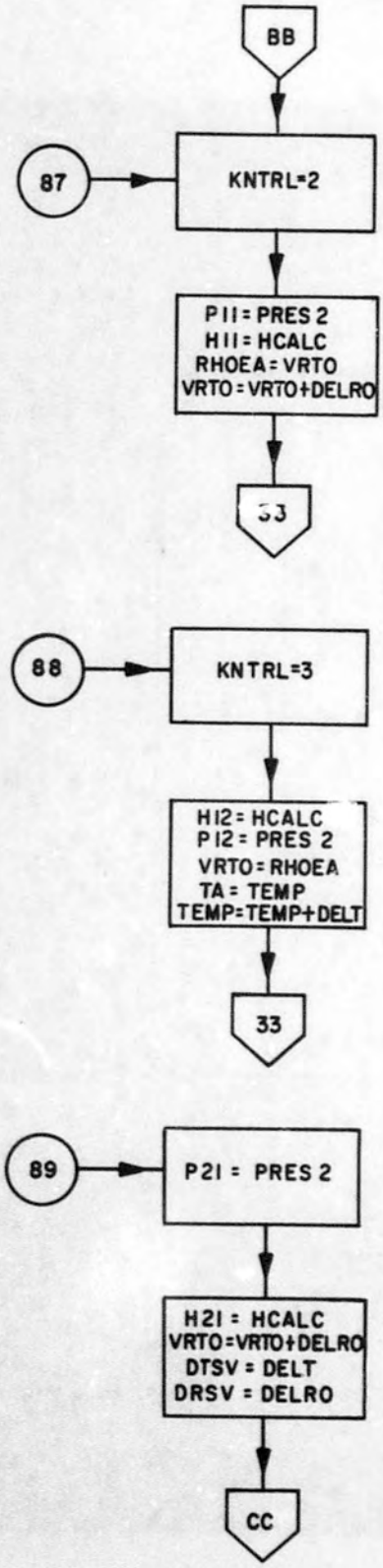
INCREASE ITERATION COUNT BY 1

TEST FOR MORE THAN 150 ITERATION

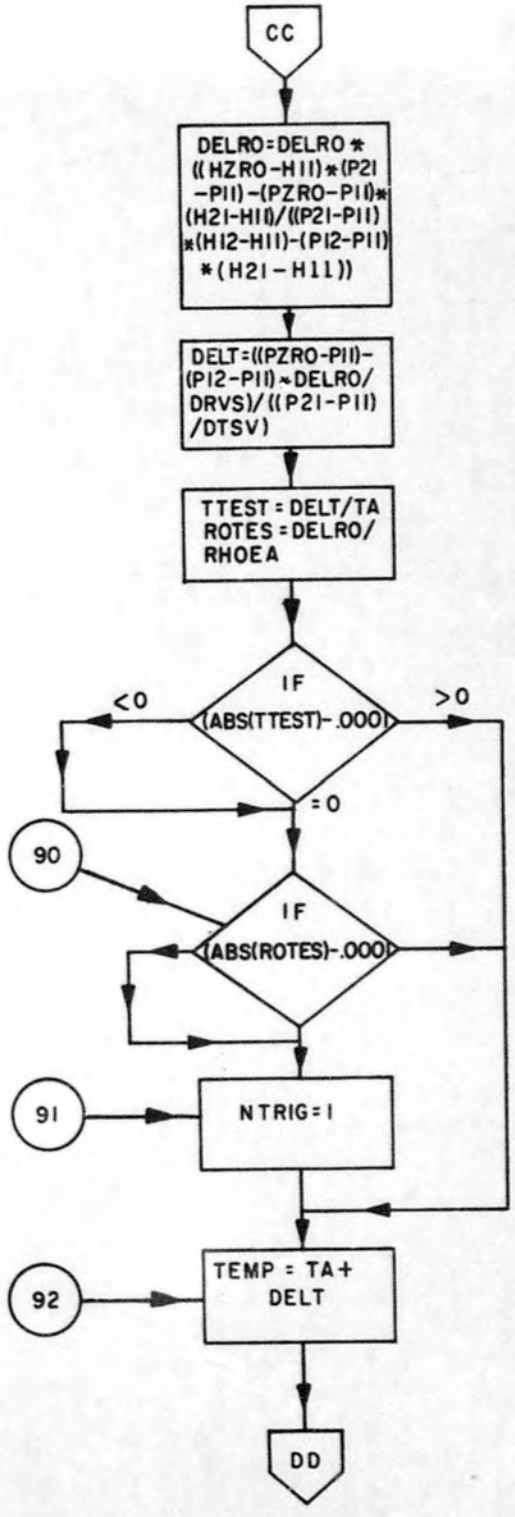


CALCULATE A NEW DELT


82
AA
TEST KNTRL
=2
=1
83
KNTRL=2
SET KNTRL=2 AFTER FIRST CALCULATION
84
PTWO = PRES2
ROEA = VRTO
VRTO=VRTO+DELRO
SAVE CALCULATED VALVES
33
85
DELP=(PRES2 - PTWO)/DELRO
DELRO=(PZRO - PRES2)/DELP
ROTES=DELRO/ROEA
CALCULATE A NEW DELRO
IF ABS(ROTES)-.0001
≤0
>0
93
TEST FOR CONVERGENCE
86
TEST KNTRL
89
=3
=1
87
=2
88
BB



SAVE CALCULATED VALUES FROM THIRD STEP OF THE ITERATION



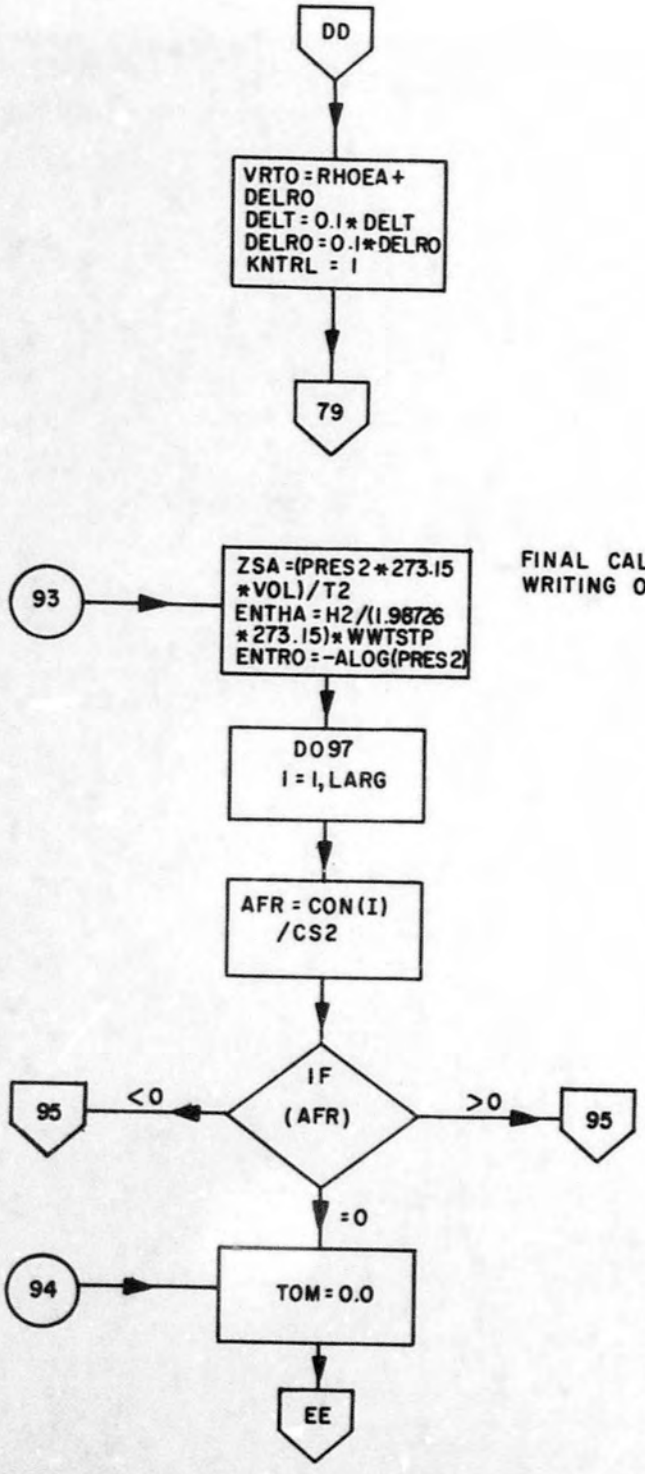
CALCULATE NEW DELRO

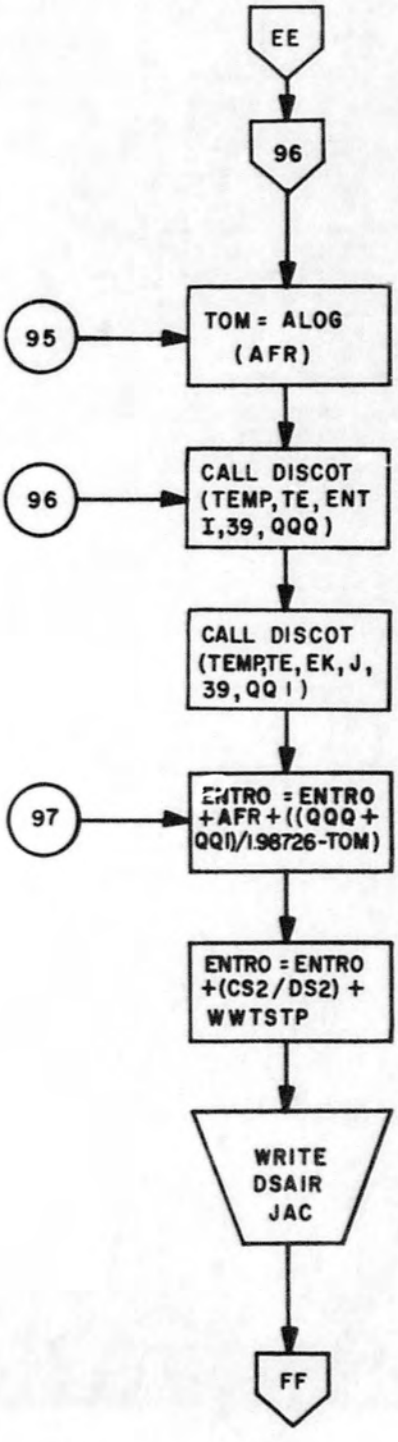
CALCULATE NEW DELT

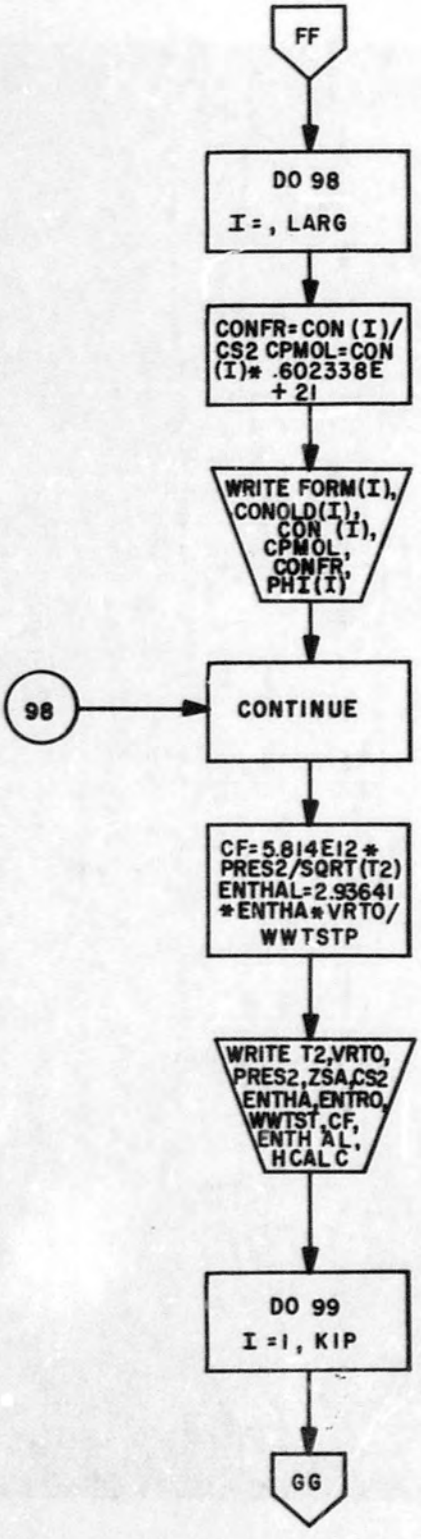
CALCULATE TESTS FOR CONVERGENCE

TEST FOR CONVERSION OF DOUBLE ITERATION

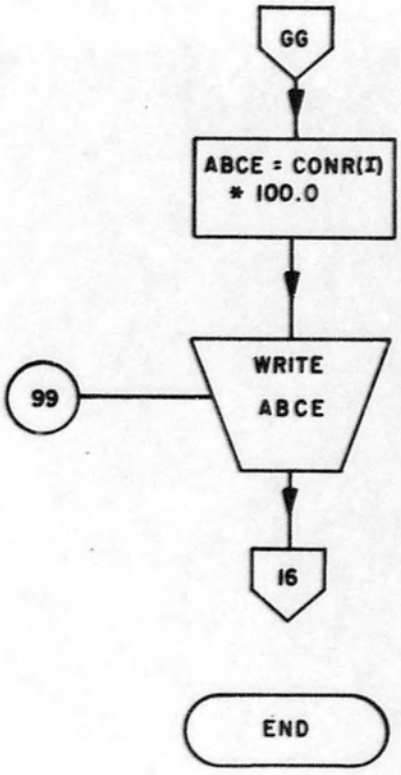
CONVERGENCE TESTS HAVE BEEN PASSED SET NTRIG TO 1 TO ALLOW FOR ONE FINAL CALCULATION


DD
VRTO = RHOEA + DELRO
DELT = 0.1 * DELT
DELRO = 0.1 * DELRO
KNTRL = 1
79
93
ZSA = (PRES2 * 273.15 * VOL) / T2
ENTHA = H2 / (1.98726 * 273.15) * WWTSTP
ENTRO = -ALOG(PRES2)
FINAL CALCULATIONS BEFORE WRITING OUTPUT FOR THIS CASE
DO 97 I = 1, LARG
AFR = CON(I) / CS2
IF (AFR)
<0
>0
95
95
=0
94
TOM = 0.0
EE


EE
96
95
TOM = ALOG (AFR)
96
CALL DISCOT (TEMP, TE, ENT I, 39, QQQ)
CALL DISCOT (TEMP, TE, EK, J, 39, QQ I)
97
ENTRO = ENTRO + AFR + ((QQQ + QQI)/1.98726-TOM)
ENTRO = ENTRO + (CS2/DS2) + WWTSTP
WRITE DSAIR JAC
FF



WRITE FORM(I),CONOLD(I)
CON(I),CPMOL,CONFR,PHI(I)


GG
ABCE = CONR(I)
* 100.0
WRITE
ABCE
99
16
END

```
C.  LISTING

      SEGMENT SIX(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
C      T.E. HORTON  SHOCK AND CHEMICAL EQUILIBRIUM PROGRAM                    00000020
      COMMON M,L,NUM,MIN,LARG,NUP,KSAVE,NNOD
      COMMON          C       (   70),CON     (   70),CONOLD (   70),
     *                CONR    (   70),
     *CPP    (   70),
     *D      (   70),DELDH   (    2),DSO     (   70),
     *EFR    (70,  1),FK      (70,39),ENT     (70,39),
     *FORM   (   70),FR      (   30),
     *IDRC   (   70),IDRF    (   15),IID     (   30),
     *NA     (   15),
     *PHI    (   70),
     *SB     (   70),SEK     (70,39),SEFR    (   70),SENT    (70,39),
     *SFCON  (   70),SFORM   (   70),STSB    (   15),STY     (70,15),
     *SWTM   (   70),MAXE    (    1),MAXS    (    1),JT      (    1),
     *TE     (   39),TTEMP   (    4),
     *WTM    (   70),WTR     (   70),
     *Y      (70,70),
     *Z      (   70)
      COMMON       YY,FS,RHOOW,TRS,CFS,ELS,HZ,HAW,HE,RHOW,TITL(5),
     * C5,C6,UE,B,CIR
      DIMENSION YY(101),TRS(101),CFS(101),ELS(101) ,RHOOW(101)
      EQUIVALENCE(MAXEX,MAXE(1)),(MAXSX,MAXS(1)),(JTX,JT(1))
501   FORMAT(2I2)
4804  FORMAT(15H THE VALUE JTR=,I4,2X,12HIS NOT VALID)                       00000130
6600  FORMAT(4X,1E12.6)                                                        160
7002  FORMAT (I2,A4,E12.6)
7004  FORMAT(12XA4,15I2,F10.6)                                                 210
7008  FORMAT(20X,5F7.1)                                                        220
7009  FORMAT(20X,5E12.6)                                                       230
7110  FORMAT(20I4)                                                             250
9001  FORMAT(3X,A4,2X,2E13.6)                                                  430
9620  FORMAT (1H1,30X,24HTURBULENT BOUNDARY LAYER/1H0,30X,5A10/
     * 1H0,9X,70H CHEMICAL EQUILIBRIUM COMPOSITION AND PROPERTIES OF A COM
     *PLEX MIXTURE //
     *       9H   RHOO =1E14.6,5H GM/L 7X,22HNUMBER OF ITERATIONS =I12/ 00000420
     *       10X,70HCONCENTRATION - MOLE/LITER  CONCENTRATION  MOLE FRAC00000490
     1TION  -LOGE(KP)      /                                                  500
     *       11X,40H(FINAL-1)        (FINAL)     PARTICLES/CC     )         00000510
9604  FORMAT(20H   FINAL CONDITIONS /                                         00450
     *       7X, 70H TEMP DEG-K       RHO/RHOO        P/PO          Z        00000520
     1    SUM CON MOL/L /                                                     530
```

```
     *       6X,5E14.6////                                                    550
     *       10X,19H HMO/RTO       SMO/R/                                 0.00540
     *       6X,2E14.6////                                                  560
     *       26H    COLD MIXTURE MOLE WT. =F10.6////                        00000570
     *       26H      COLLISION FREQUENCY =1PE13.6////
     *       16X10HENTHALPY =1PE13.6,24HJOULES / CC       HCALC =E13.6//)
 9690 FORMAT(3X,A4,2X,5E14.6)                                                480
 9693 FORMAT(3X,23HTOTAL CONCENTRATION OF A4,23HAT STANDARD CONDITIONS  00000580
     1  E13.5, 12H PER CENT    )
10009 FORMAT (I3,5E12.6)
      NCODE = KSAVE
      INTAP = 5
      IOTAP = 6
      REWIND 22
      INDEX=-1
      KIT = 1
      ITER = 0
      ICOUNT = 4
      IF(NCODE.EQ.4) GO TO 6001
      READ (INTAP,501) NCODE,NNOD
   44 GO TO (8,8,4,6000), NCODE
    4 CONTINUE
    8 READ (INTAP,501) MAXE,MAXS
 7153 READ (INTAP,501) JTX
      READ (INTAP,7008) (TE(I),I=1,JTX)
      DO 7013 I=1,MAXSX                                                      660
      READ (INTAP,7004) SW,NA, SWTM(I)
      SFORM(I)=SW                                                            680
      DO 7005 J=1,MAXEX                                                      690
 7005 STY(I,J)=NA(J)                                                         700
      READ (INTAP,7009) (SEX(I,J),J=1,JTX)
      READ (INTAP,7009)(SENT(I,J),J=1,JTX)
 7013 READ (INTAP,7009) SEE(I)
      KSAVE = 4
 6000 READ (INTAP,501) M,L
      DO 7601 J=1,MAXEX                                                      750
 7601 IDRE(J)=0                                                              760
      DO 7605 I=1,MAXSX                                                      770
 7605 IDRC(I)=0                                                              780
      NUM=L-M                                                                790
      MIN=M                                                                  800
      LARG=L                                                                 810
      NUP=L-M+1                                                              820
      READ (INTAP,7110) (IDRC(I),I=1,LARG)
      DO 7011 I=1,MIN                                                        840
      K=NUM+I                                                                850
```

```
 7011 IDRE(I)=IDRC(K)                                                    860
      DO 6601 I=NUP,LARG                                                 890
 6601 READ (INTAP,6600) CON(I)
C
 6001 READ (22,501) IDUM,KIP
      READ (22,7002) (IID(I),FR(I),CONR(I),I=1,KIP)
 3006 IST = 0
 6400 SCON=(1.0)/(0.0820575*273.15)                                     0930
      DO 7112 I=1,MAXEX                                                  940
 7112 STSB(I)=0.0                                                        950
      DO 7114 I=1,KIP                                                    960
      K=IID(I)                                                           970
      DSST=CONR(I)*SCON                                                  980
      DO 7114 J=1,MAXEX                                                  990
 7114 STSB(J)=STSB(J)+(DSST*STY(K,J))                                 0 01000
      N=1                                                               1010
      K=NUP
      DO 6405 M=1,MAXEX                                                 1030
      LOPP=MAXS-MAXE+M                                                  1040
      IF(IDRE(N)-LOPP)6405,6403,6405
 6403 I=M                                                               1060
      DO 6404 L=1,LARG                                                  1070
      J=IDRC(L)                                                         1080
 6404 Y(L,K)=STY(J,I)                                                   1090
      SB(K)=STSB(M)                                                     1100
      N=N+1                                                             1110
      K=K+1                                                             1120
 6405 CONTINUE                                                          1150
      DO 6406 I=1,LARG                                                  1160
      K=IDRC(I)                                                         1170
      WTM(I)=SWTM(K)                                                    1180
      EFR(I,1)=SEFR(K)                                                  1190
      FORM(I)=SFORM(K)                                                  1200
      DO 6406 J=1,JTX                                                   1210
      EK(I,J)=SEK(K,J)                                                  1220
 6406 ENT(I,J)=SENT(K,J)                                                1230
 7602 DS0=0.0                                                           1380
      DO6407 I=NUP,LARG                                                 1390
 6407 DS0=DS0+(SB(I)*WTM(I))                                            1400
      WWTSTR=DS0*273.15*0.0820575                                     0.01410
      DSAIR=(28.966348)/(273.15*0.0820575)                            0.01420
      GO TO(9000,9070),KIT                                              1430
 9070 IF (IVY) 4808,9000,4808                                           1440
 4808 VOL=DS0/(DS1*VOLBA)                                               1450
 9000 IF(ICOUNT-3)10005,10005,10000
10000 READ (22,10009) JTR,VRTO,TTEMP
```

```
      IF (EOF,22) 10333,10300
10333 REWIND 22
      CALL CHAIN (SEVEN)
10300 LCOUNT = 1
      IF(JTR)10001,10001,10002
10001 TEMP = TTEMP(1)
      NTRIG=1
      ICOUNT = 1
      GO TO 6055
10002 IF(JTR-100)5000,6000,4801
 5000 IF(JTR-70)5001,200,250
 5001 WRITE (IOTAP,4804) JTR
      CALL EXIT
  200 NTRIG=2
      DELT=0.1*TTEMP(1)
  201 TEMP=TTEMP(1)
      INDEX = INDEX + 2
      HZRO = TTEMP(3)
      PZRO=TTEMP(2)
      KNTRL=1
  202 PORT=TTEMP(2)/TEMP/0.0820575
      JITER=0
      GO TO 6055
  250 IF (JTR - 80) 5001,251,262
  262 IF (JTR - 90) 5001,265,275
  275 TEMP = TTEMP(1)
      NTRIG = 1
      GO TO 201
  265 DELT  = 0.1 * TTEMP
      DELPO = .0001 * VPTO
      NTRIG = 4
      GO TO 201
  251 NTRIG=3
      DELPO=.0001*VPTO
      GO TO 201
10005 IF(TTEMP(ICOUNT+1))10000,10000,10006
10006 ICOUNT = ICOUNT+1
      TEMP = TTEMP(ICOUNT)
      GO TO 6055
 4801 IF(IST) 4806,4805,4806                                        1500
 4805 IVY=0                                                         1510
      GO TO  4807                                                   1520
 4806 VOLPA=PSO/(PS1*VOL)                                           1530
      IVY=1                                                         1540
 4807 IF (JTR-101)5001,4808,5001                                  0 01560
 4808 GO TO 6001                                                    1570
```

```
 6055 IF (IST)5001,5500,5001                                              1610
 5500 AIR=28.966348                                                       1620
C         SET INITIAL VALUES                                             01750
 5511 JAC = 0                                                             1760
      DO 7012 I=1,LARG                                                    1770
      PHI(I)=0.0                                                          1780
      D(I)=0.0                                                            1790
C         CALCULATION OF SPECIFIC ENTHALPIES AND EQUILIBRIUM CONSTANTS   00001800
      CALL DISCT(TEMP,TE,ENT,I,39,000)
 7012 CPP(I) = 000*TEMP + 1000.*EER(I,1)
      DO 7007 I=1,NUM                                                     1820
      CALL DISCT(TEMP,TE,EK,I,39,000)
      PHI(I) = -(000-EER(I,1)*1000./TEMP)/1.98726
      DO 7007 K=NUP,LARG                                                  1840
      CALL DISCT(TEMP,TE,EK,K,39,000)
      BBR = (000-EER(K,1)*1000./TEMP)/1.98726
 7007 PHI(I)=PHI(I) +Y(I,K)*BBR                                           01860
C         TEST FOR FIRST RUN KIT=1 FOR FIRST RUN,KIT=2 FOR ALL OTHERS    00001870
      GO TO(7901,9200),KIT                                                1880
C         CALCULATION OF INITIAL GUESS FOR FIRST RUN                     00001890
 7901 TE=ALOG(TEMP*0.0820575)
      VRTO=PORT*WWTSTP/AIR
      KIT=2
      DO 5777 K=1,NUM
      D(K) = -(PHI(K)+TE)
      DO 5797 I=NUP,LARG
 5797 D(K) = D(K) + ((TE+ALOG(CON(I)))*Y(K,I))
 5777 CON(K) = EXPX(5777,D(K))
C         BEGINNING OF EQUILIBRIUM ITERATION LOOP
 9200 JAC=JAC+1
      VOL=WWTSTP/(AIR*VRTO)
C         TEST FOR NUMBER OF ITERATIONS
      IF(JAC-150)5003,5003,5991
C         EQUILIBRIUM BLOCK NOT CONVERGENT BLOCK
 5002 FORMAT(48H EQUILIBRIUM STATE NOT REACHED IN 150 ITERATIONS)
 5991 WRITE (IOTAP,5002)
      DO 5004 I=1,LARG
      WRITE (IOTAP,9601) FORM(I),CON(I),D(I)
 5004 WRITE (IOTAP,9601) FORM(I),CONOLD(I),DSO(I)
      GO TO 9000
C         EQUILIBRIUM BLOCK
 5003 COLD=CS2
      HOLD=H2                                                             2110
      DOLD=DS2                                                            2120
      DO 9514 I=1,LARG                                                    2130
      DSO(I)=D(I)                                                         2140
```

```
 9514 CONOLD(I)=CON(I)                                                    2150
      DO 9203 I=1,MIN                                                     2160
      DO 9203 J=1,MIN                                                     2170
      Y(I,J)=0.0                                                          2180
      M=NUM+I                                                             2190
      N=NUM+J                                                             2200
C                                                                         2210
      DO 9202 K=1,NUM                                                    00002220
 9202 Y(I,J)=Y(I,J)+Y(K,M)*Y(K,N)*CON(K)                                  2230
      IF(I-J) 9203,9201,9203                                              2240
C                                                                         2250
 9201 Y(I,J)=Y(I,J)+CON(M)                                               02260
 9203 CONTINUE                                                            2270
      TF=ALOG(TEMP*0.0820575)                                            0 02280
      DO 9205 K=1,NUM                                                     2290
      IF(CON(K)) 9313,9312,9313                                           2300
 9312 ZIGG=0.0                                                            2310
      GO TO 9315                                                          2320
 9313 ZIGG=ALOG(CON(K))                                                   2330
C                                                                         2340
 9315 D(K)=DHI(K)+TF+ZIGG                                                2350
      DO 9205 I=NUP,LARG                                                  2360
      IF(CON(I)) 9311,9310,9311                                           2370
 9310 ZAGG=0.0                                                           0 02380
C                                                                         2390
      GO TO 9205                                                          2400
 9311 ZAGG=ALOG(CON(I))                                                   2410
 9205 D(K)=D(K)-((TF+  ZAGG        )*Y(K,I))                             02420
      DO 9206 I=NUP,LARG                                                  2430
      D(I)=CON(I)-(SB(I)/VOL)                                             2440
      DO 9206 K=1,NUM                                                     2450
 9206 D(I)=D(I)+(CON(K)*Y(K,I))                                           2460
      DO 9207 I=1,MIN                                                     2470
      K=NUM+I                                                             2480
      C(I)=D(K)                                                           2490
      DO 9207 J=1,NUM                                                     2500
 9207 C(I)=C(I)-D(J)*CON(J)*Y(J,K)                                        2510
      DO 9301 K=2,MIN                                                     2520
 9301 Y(1,K)=Y(1,K)/Y(1,1)                                                2530
      C(1)=C(1)/Y(1,1)                                                    2540
      DO 9306 L=2,MIN                                                    0 02550
      N=L-1                                                               2560
      DO 9302 J=L,MIN
      DO 9302 K=1,N
 9302 Y(J,L)=Y(J,L)-Y(J,K)*Y(K,L)
      LA=L+1
```

```
      DO 9304 I=LA,MIN                                                  2570
      DO 9303 K=1,N                                                     2580
      Y(L,I) = Y(L,I) - (Y(L,K)*Y(K,I))
 9303 CONTINUE
C
      Y(L,I) = Y(L,I)/Y(L,L)
 9304 CONTINUE
      DO 9305 K=1,N                                                     2610
 9305 C(L)=C(L)-(Y(L,K)*C(K))                                           02620
 9306 C(L)=C(L)/Y(L,L)                                                  2630
      M=MIN                                                             2640
      Z(M)=-C(M)                                                        2650
      DO 9307 I=2,MIN                                                   2660
      N=MIN+1-I                                                         2670
      NET=N+1                                                           2680
      Z(N)=-C(N)                                                        2690
      DO 9307 K=NET,MIN                                                 2700
 9307 Z(N)=Z(N)-Y(N,K)*Z(K)                                             2710
      DO 9501 I=1,MIN                                                   2720
      K=LARG-MIN+I                                                      2730
 9501 Z(K)=Z(I)                                                         2740
      DO 9503 I=1,NUM                                                   2750
      Z(I)=-D(I)                                                        2760
      DO 9503 J=NUP,LARG                                                2770
 9503 Z(I)=Z(I)+Y(I,J)*Z(J)                                             02780
      DO 9602 I=1,LARG                                                  2790
      CON(I) = EXPX(9503,Z(I)) * CON(I)
      IF(CON(I))9602,9506,9507                                          02810
 9506 CON(I) = 1.000E-35                                                2812
      GO TO 9602                                                        2814
 9507 IF(CON(I) - .1)9602,9602,9508                                     00002816
 9508 CON(I) = .1                                                       2818
 9602 CONTINUE                                                          2830
C         CALCULATION OF THERMODYNAMIC PROPERTIES                      00002840
      CS2=0.0                                                           2850
      H2=0.0                                                            2860
      DS2=0.0                                                           2870
      DO 9072 I=1,LARG                                                  2880
      CS2=CS2+CON(I)                                                    2890
      H2=H2+(CPR(I)*CON(I))                                             2900
 9172 DS2=DS2+(WTM(I)*CON(I))                                           02910
      VRTO=PORT*DS2/(CS2*RSAIR)
C         TEST FOR CONVERGENCE OF EQUILIBRIUM PROPERTIES               00002920
 4066 TES1=1.0-(COLD/CS2)                                               2930
      ZT1=ABS(TES1)                                                     2940
      IF(ZT1-0.000001) 9901, 9501, 9211                                 0002950
```

```
 9901 TES2=1.0-(DOLD/DS2)                                                2960
      ZT2=ABS(TES2)                                                      2970
      IF(ZT2-0.000001) 9902, 9902, 9211                                 0000298C
 9902 TES3=1.0-(HOLD/H2)                                                 2990
      ZT3=ABS(TES3)                                                      3000
      IF(ZT3-0.000001) 25000,25000,9211
25000 TES4 = 1.0 - (CONOLD(NNOD)/CON(NNOD))
      ZT4 = ABS(TES4)
      IF(ZT4-0.000001) 9903,9903,9211
 9211 IF(JAC-4) 9221, 9221, 9200                                         03020
 9221 DO 9220 I=1,LARG                                                   3030
 9220 CON(I)=(CONOLD(I)+CON(I))/2.0                                    0 03040
      GO TO 9200                                                         3050
 9903 IF(JITER)9904,9904,9905
 9904 JITER=1
      GO TO 9211
 9905 CONTINUE
      H2=H2/DS2
      HCALC = H2 * 1.8
      T2=TEMP                                                            3070
      P2=(1.98726*T2*CS2)                                                3080
      PRES2 = (T2*CS2*0.0820575)
      GO TO (299,259,299,500),NTRIG
  259 GO TO (260,270),KNTRL
  260 KNTRL=2
  261 PTWO=PRES2
      TA=TEMP
      TEMP=TEMP+DELT
  300 LCOUNT =LCOUNT +1
      IF(LCOUNT-150)202,202,301
  301 WRITE (IOTAP,302)
  302 FORMAT(41H CASE DID NOT CONVERGE IN 150 ITERATIONS.)
      GO TO 299
  270 DELP=(PRES2-PTWO)/DELT
      DELT=(PZRO-PRES2)/DELP
      TTEST=DELT/TA
      IF(ABS(TTEST)-.0001)299,299,261
  500 GO TO (510,520),KNTRL
  510 KNTRL = 2
  530 HTWO=HCALC
      TA=TEMP
      TEMP=TEMP+DELT
      IF (TEMP) 1263,1263,300
 1263 TEMP = 0.9 * TA
      DELT = TEMP - TA
      GO TO 300
```

```
  520 DELH=(HCALC-HTWO)/DELT
      DELT=(HZRO-HCALC)/DELH
      TTEST=1.0-(HZRO/HCALC)
      IF(ABS(TTEST)-0.000001)299,299,530
  299 ZSA=(PRES2*273.15*VOL)/T2                                          0003110
      ENTHA=H2/(1.98726*273.15)*WWTSTP                                     3120
      ENTRO=-ALOG(PRES2)                                                   3130
      DO 9695 I=1,LARG                                                     3140
      AFR=CON(I)/CS2                                                       3150
      IF(AFR)9694,9696,9694                                                3160
 9696 TOM=0.0
      GO TO 19695
 9694 TOM=ALOG(AFR)                                                        3180
19695 CALL DISCT(TEMP,TE,ENT,I,39,QQQ)
      CALL DISCT(TEMP,TE,EK,I,39,QQ1)
 9695 ENTRO=ENTRO+AFR*((QQQ+QQ1)/1.98726 -TOM)
      ENTRO=ENTRO*(CS2/DS2)*WWTSTP                                      03200
      IF (IST)5001,9090,5001
C        EQUILIBRIUM SOLUTION PRINT                                   *0003960
 9090 WRITE (6,9620) TITL, DSAIR,JAC
      DO 9095  I=1,LARG                                                    4010
      CONFR=CON(I)/CS2                                                     4020
      CPMOL=CON(I)*.602338E+21                                          04030
 9095 WRITE (IOTAP,9690) FORM(I),CONOLD(I),CON(I),CPMOL,CONFR,PHI(I)
      VPP=VRTO                                                             4070
      CF=5.814E12*PRES2/SQRT(T2)
      ENTHAL = 2.93641 * ENTHA * VPP / WWTSTP                           00004116
      WRITE (IOTAP,9604) T2,VPP,PRES2,ZSA,CS2,ENTHA,ENTRO,WWTSTP,CF,
     *ENTHAL, HCALC
      DO 9699 I=1,KIP                                                      4120
      ABCE=CONR(I)*100.0                                                   4130
 9699 WRITE (IOTAP,9693) FR(I),ABCE
      ELS(INDEX) = CON(NNOD) * .602338E+21
      TRS(INDEX) = T2
      CFS(INDEX) = CF
      RHOOW(INDEX) = VPP / RHOW * 1.29233 / 16.018432
      KIT=1
      GO TO 9000                                                           4150
      END                                                                  5220
```

d. Symbols

Symbolic representation	Meaning or use
ABCE	Mole percent at STP
AFR	Mole fraction
AIR	Molecular weight of air at STP
BBR	Variable used in PHI calculation
CF	Collision frequency
COLD	Previous concentration
CON	Initial guesses and final concentration
CONFR	Mole fraction (final)
CONOLD	Concentration previous to final one
CONR	Mole fraction at STP
CPMOL	Moles per liter (final)
CPP	Individual enthalpy (cal/mole)
CSZ	Calculated concentration
DELP	Partial of P with respect to ρ or T
DELRO	Density increment
DELT	Temperature increment
DOLD	Previous density
DRSV	Previous calculated density
DSZ	Calculated density
DSAIR	Density of air at STP
DTSV	Previously calculated T
EFR	Energy of formation for initial condition group
EK	Free energy for initial condition group
ENT	Enthalpy for initial condition group
ENTHA	Enthalpy (dimensionless)

ENTHAL	Enthalpy (joules/cc)
ENTRO	Entropy (dimensionless)
FORM	Chemical symbols for initial condition group
FR	Chemical species symbol
H11	Enthalpy values used in double iteration
H12	Second pressure value used in double iteration
H2	Calculated enthalpy
H21	Third enthalpy value used in double iteration
HCALC	Calculated enthalpy (Btu/lbm)
HOLD	Previous enthalpy
HZRO	H_o - enthalpy iterated to
I	DO LOOP index
ICOUNT	Trigger and index for TTEMP array
IDRC	Identification numbers of species in mixture
IDRE	identification numbers of elements on mixture
IID	Species identification numbers
INTAP	System input tape unit number
IOTAP	System output tape unit number
J	DO LOOP index
JAC	Iteration loop count
JTR	Control word on final condition card
JTX	Number of temperatures in TE array
K	DO LOOP index
KIP	Number of compounds present in the mixture at standard conditions
KIT	Trigger for first run or other run

KNTRL	Internal control used in iteration
L	DO LOOP index
LA	Dummy used in DO LOOP
LARG	Total number of chemical species in mixture
LCOUNT	Iteration count
LOPP	Index used in storing formula vectors in Y
M	DO LOOP index
MAXEX	Number of elements in library
MAXSX	Number of species in library
MIN	Total number of elements in mixture
N	Miscellaneous index
NA	Formula vectors for library data
NET	Dummy used in DO LOOP
NTRIG	Trigger for type of condition calculation
NUM	Number of species which are not elements in mixture
NUP	NUM + 1
P11	Pressure value used in double iteration
P12	Second enthalpy value in double iteration
P21	Third pressure value used in double iteration
PH1	Log (-KP) where KP = equilibrium constant
PRES2	Calculated pressure ratio
PTWO	Previously calculated pressure
PZRO	P_o - pressure iterated to
QQ1	Dummy used in table look-up
QQQ	Dummy value from table look-up

RHOEA	Previously calculated VRTO for double iteration
ROEA	Previously calculated VRTO
ROTES	Convergence test for density
SCON	Constant; 1.0/0.0820575*273.15
SEFR	Energy of formation h° (Kcal/mole)
SEK	Free energy - $(f^\circ - h_o^\circ)/T$, (cal/mole - °K)
SENT	Enthalpy $(h^\circ - h_o^\circ)/T$, (cal/mole - °K)
SFORM	Array of chemical formulas in library
STY	Formula vectors $(\alpha_{ij}'s)$
SW	Chemical formula read from card
SWTM	Molecular weight array
T2	Calculated temperature
TA	Previously calculated temperature
TE	Temperature array
TEMP	Temperature
TES1, TES2, TES3, TES4	Variables used in test for convergence
TF	Log (TEMP*0.0820575)
TOM	Logarithm of mole fraction
TTEMP	Values of T, P, H for final condition calculation
TTEST	Convergence test value for T
VOL	Volume
VRTO	Ratio of density at solution point to density of air at STP
WTM	Molecular weights for initial condition group
WWSTP	Molecular weight at STP (cold)
Y	Formular vectors for initial condition group

ZAGG, ZIGG	Log of concentration (dummy)
ZSA	Compressibility
ZT1	Test value for convergence of concentration
ZT2	Test value for convergence of density
ZT3	Test value for convergence of enthalpy
ZT4	Test value for convergence of individual specie concentration

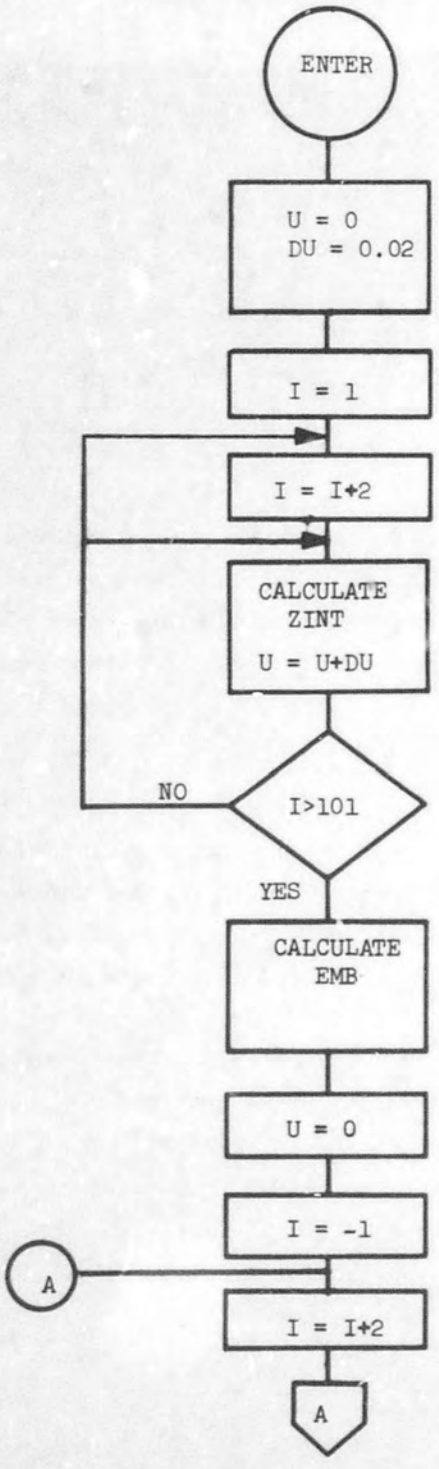
10. SEGMENT SEVEN

a. Function

In this segment, the important boundary layer profiles are collected and printed. These profiles are velocity ratio, density ratio, mass fraction, temperature, collision frequency, electron density, normal distance from the surface (both physical and nondimensional), and enthalpy. In addition, the ablation gas flow and total gas flow in the boundary layer are computed and printed. Any profile of interest may also be punched on 80-column cards by the addition of PUNCH statements.

The profiles of density, temperature, collision frequency, and electron density are calculated in Segment SIX and are transmitted to Segment SEVEN via common statements. The boundary condition of wall mass fraction, wall enthalpy, edge enthalpy, and adiabatic wall enthalpy are calculated in Segment ONE and are also transmitted to Segment SEVEN via common. The distance normal to the surface at which a given velocity ratio is found is also calculated in Segment ONE and transmitted via common. These boundary conditions are used to calculate the mass fraction and enthalpy profiles. The density, mass fraction, and distance profiles are used in the integration to determine the ablation gas flow in the boundary layer. These same profiles, with the exception of mass fraction, are also used to calculate the total boundary layer gas flow. Once the profiles are printed, control is transferred to Segment EIGHT.

b. Flow Chart

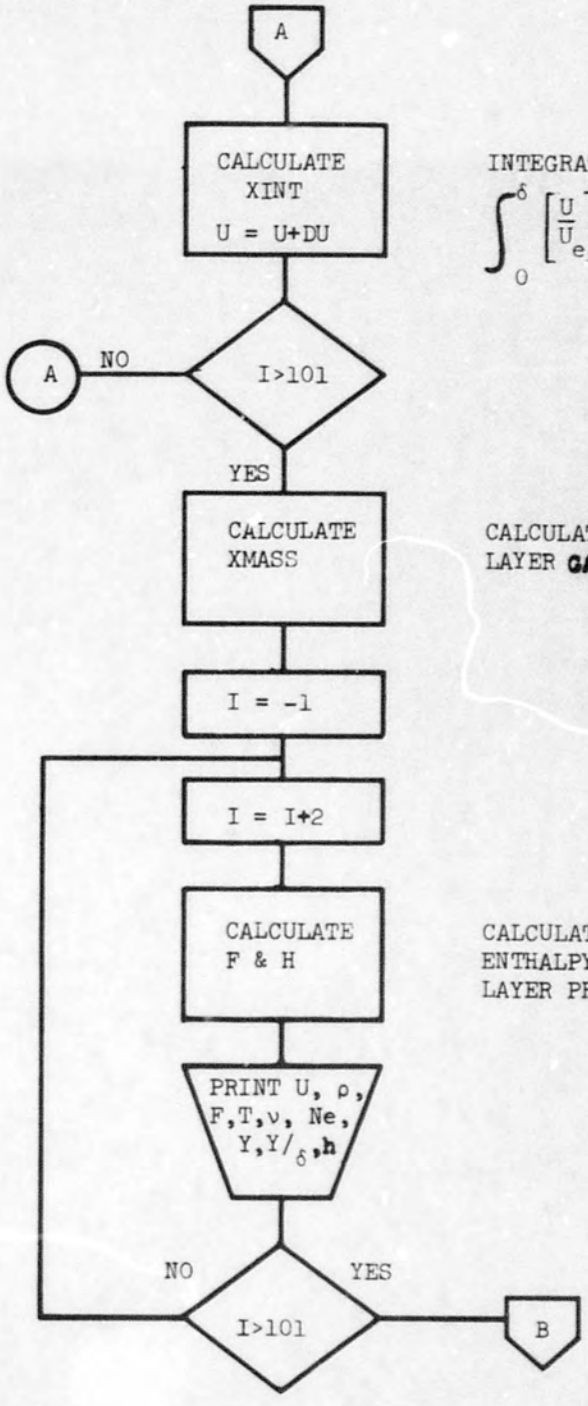


INITIALIZE VELOCITY RATIO AND SET INCREMENT

INTEGRATION OF

$$\int_0^{\delta} \left[\frac{U}{U_e}\right] \left[\frac{\rho}{\rho_w}\right] \left[1-\frac{U}{U_e}\right] dy$$

CALCULATE ABLATION GAS FLOW IN BOUNDARY LAYER

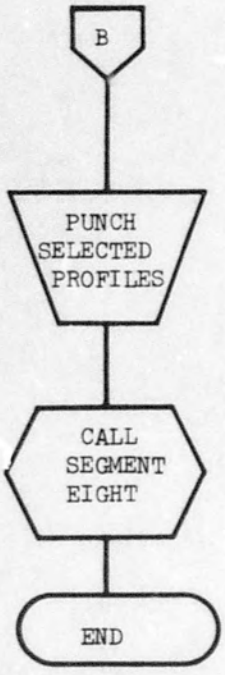


INTEGRATION OF

$$\int_0^{\delta} \left[\frac{U}{U_e}\right] \left[\frac{\rho}{\rho_w}\right] dy$$

CALCULATE TOTAL BOUNDARY LAYER GAS FLOW

CALCULATE MASS FRACTION AND ENTHALPY AND PRINT BOUNDARY LAYER PROFILE



PUNCH SELECTED BOUNDARY LAYER PROFILES

```
C.  LISTING

      SEGMENT SEVEN(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
      COMMON M,L,NUM,MIN,LARG,NUP,KSAVE,NNOD
      COMMON          C       (   39),CON     (   70),CONOLD (   70),
     *                CONR    (   70),
     *CPP    (   70),
     *D      (   70),DELDH   (    2),DSO     (   70),
     *EFR    (70,39),EK      (70,39),ENT     (70,39),
     *FORM   (   70),FR      (   30),
     *IDRC   (   70),IDRE    (   15),IID     (   30),
     *NA     (   15),
     *PHI    (   70),
     *SB     (   70),SEK     (70,39),SEFR    (   70),SENT    (70,39),
     *SFCON  (   70),SFORM   (   70),STSB    (   15),STY     (70,15),
     *SWTM   (   70),MAXE    (    1),MAXS    (    1),JT      (    1),
     *TE     (   39),TTEMP   (    4),
     *WTM    (   70),WTR     (   70),
     *Y      (70,70),
     *Z      (   70)
      COMMON        YY,FS,RHOOW,TRS,CFS,ELS,HW,HAW,HE,RHOW,TITL(5),
     * C5,C6,UE,B,CIR
      DIMENSION         YY(101) ,TRS(101),CFS(101),ELS(101),RHOOW(101)
      DIMENSION SAVEA(101),SAVEB(101),SAVEC(101),SAVED(101),SAVEE(101),
     *SAVEF(101),SAVEG(101),SAVEH(101),SAVEI(101)
C
C       THIS SEGMENT PRINTS OUT THE VELOCITY, DENSITY, MASS FRACTION,
C       TEMPERATURE, COLLISION FREQUENCY, ELECTRON CONCENTRATION,
C       Y, AND YD PROFILES CALCULATED IN SEGMENTS ONE AND SIX.
C
      U = 0.0
      DU = .02
      DO 500 I = 1,101,2
      X = SQRT(RHOOW(I) / (1.0+B*U))
      Z = U * RHOOW(I) * (1.0-U)
      IF (I-1) 300,200,300
  200 XINT = 0.0
      GO TO 400
  300 XINT = XINT + DU * (X+X0) / 2.0
  400 X0 = X
      YY(I) = C5* EXP(C6*XINT)
      IF (I-1) 440,420,440
  420 ZINT = 0.0
      GO TO 460
```

```
  440 ZINT = ZINT + (Z+ZO) * (YY(I) - YY(I-1)) / 2.0
  460 ZO = Z
  500 U = U + DU
      EMB = CIR * RHOW * UE * FS * ZINT
      U = 0.0
      DO 900 I = 1,101,2
      X = RHOOW(I) * U
      IF (I-1) 700,600,700
  600 XINT = 0.0
      GO TO 800
  700 XINT = XINT + (X+XO)*(YY(I)-YY(I-1))/2.0 /YY(101)
  800 XO = X
  900 U = U + DU
      XMASS = CIR * RHOW * UE *YY(101) * XINT
      U = 0.0
      WRITE (6,2000) TITL
      DO 1000 I = 1,50,2
      YD = YY(I) / YY(101)
      F = FS * (1.0 - U)
      H = HW + (HAW- HW) * U - (HAW-HE) * U**2
C
      WRITE (6,3000) U,RHOOW(I),F,TRS(I),CFS(I),ELS(I),YY(I),YD    ,H
C
      U = U + DU
 1000 CONTINUE
 2000 FORMAT (1H130X24HTURBULENT BOUNDARY LAYER/ 1H030X5A10/
     * 1H035X23HBOUNDARY LAYER PROFILES/
     *         1H04X1HU6X5HRHOOW8X1HF11X1HT13X2HCF12X2HEL13X1HY11X4HY/YD
     *12X1HH)
 3000 FORMAT (F7.3,2F11.6,6E14.5)
      WRITE (6,2000) TITL
      DO 4000 I = 51,101,2
      YD = YY(I) / YY(101)
      F = FS * (1.0 - U)
      H = HW + (HAW- HW) * U - (HAW-HE) * U**2
      WRITE (6,3000) U,RHOOW(I),F,TRS(I),CFS(I),ELS(I),YY(I),YD    ,H
      U = U + DU
 4000 CONTINUE
 5000 FORMAT (1H1)
      PUNCH 2020, TITL
 2020 FORMAT(1H1,4X,5A10)
      PUNCH 6000,     (  YY(I), I = 1,101,2)
      PUNCH 6000,     ( ELS(I), I = 1,101,2)
      PUNCH 6000,     ( CFS(I), I = 1,101,2)
 6000 FORMAT (5E15.8)
      WRITE (6,8000) XMASS,EMB
```

```
8000 FORMAT (1H020X7HXMASS =E16.7,10X5HEMB =E16.7)
     WRITE (6,5000)
     U = 0.0
     DO 100 I = 1,101,2
     SAVEA(I) = U
     SAVEB(I) = RHOOW(I)
     F = FS*(1.0 - U)
     SAVEC(I) = F
     SAVED(I) = TRS(I)
     SAVEE(I) = CFS(I)
     SAVEF(I) = ELS(I)
     SAVEG(I) = YY(I)
     YD = YY(I)/YY(101)
     SAVEH(I) = YD
     H = HW + (HAW-HW)*U - (HAW-HE)*U**2
     SAVEI(I) = H
     U = U + DU
 100 CONTINUE
     PUNCH 2020, TITL
     PUNCH 6000, (SAVEA(I), I=1,101,2)
     PUNCH 6000, (SAVEB(I), I=1,101,2)
     PUNCH 6000, (SAVEC(I), I=1,101,2)
     PUNCH 6000, (SAVED(I), I=1,101,2)
     PUNCH 6000, (SAVEE(I), I=1,101,2)
     PUNCH 6000, (SAVEF(I), I=1,101,2)
     PUNCH 6000, (SAVEG(I), I=1,101,2)
     PUNCH 6000, (SAVEH(I), I=1,101,2)
     PUNCH 6000, (SAVEI(I), I=1,101,2)
     CALL CHAIN (FIGHT)
     END
```

d. Symbols

Symbolic representation	Meaning or use
U	Velocity ratio, U/U_e
RHOOW	Density ratio, ρ/ρ_w
F	Mass fraction
TRS	Temperature
CFS	Collision frequency
ELS	Electron density
YY	Normal distance from surface
YD	YY/δ (nondimensional normal distance)
H	Enthalpy
DU	Velocity ratio increment
HW	Wall enthalpy
HAW	Adiabatic wall enthalpy
HE	Edge enthalpy
TITL	Title of case
UE	Edge velocity
RHOW	Wall density
C5	$\frac{\mu_W}{\left(\tau_W/\rho_W\right)^{1/2}}\left(\frac{1}{S}\right)$ Intermediate calculation
C6	$KU_e/\left(\tau_W/\rho_W\right)^{1/2}$ Intermediate calculation
B	$\dot{m}U_e/\tau_W$ Injection parameter
CIR	Circumference at body station = $2\pi R_b$
XMASS	Boundary layer total gas flow
DSTDEL	Ratio displacement thickness to boundary layer thickness
EMB	Boundary layer injected gas flow
X	$\sqrt{\frac{\rho/\rho_W}{1 + B\ U/U_e}}$

Z $\left(\frac{U}{U_e}\right)\left(\frac{\rho}{\rho_W}\right)\left(1 - \frac{U}{U_e}\right)$

XO (I-1) value of X (used in integration)

ZO (I-1) value of Z (used in integration)

XINT $\int_0^{U/U_e} \left[\frac{\rho/\rho_W}{1 + B\ U/U_e}\right]^{1/2} d\left(\frac{U}{U_e}\right)$

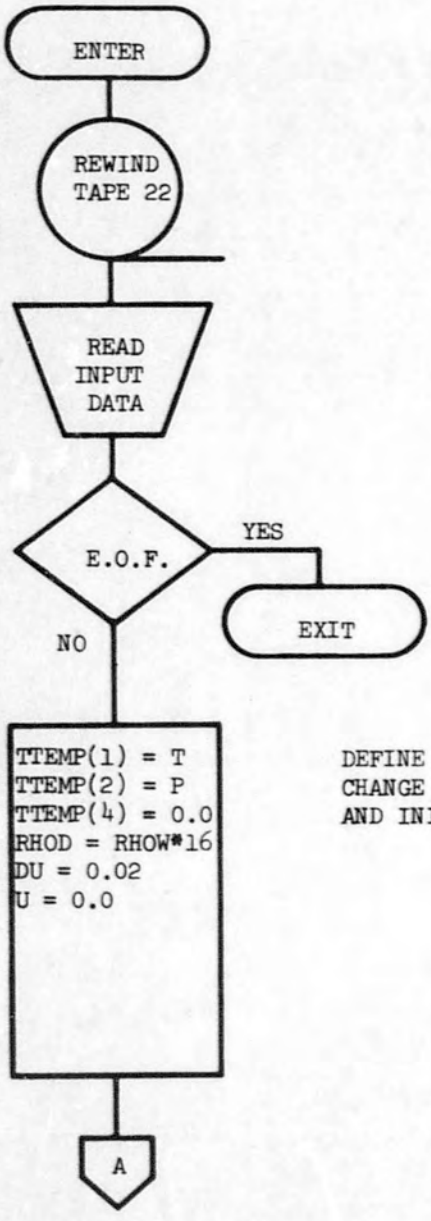
ZINT $\int_0^{\delta} \left(\frac{U}{U_e}\right)\left(\frac{\rho}{\rho_W}\right)\left(1 - \frac{U}{U_e}\right) dy$

11. SEGMENT EIGHT

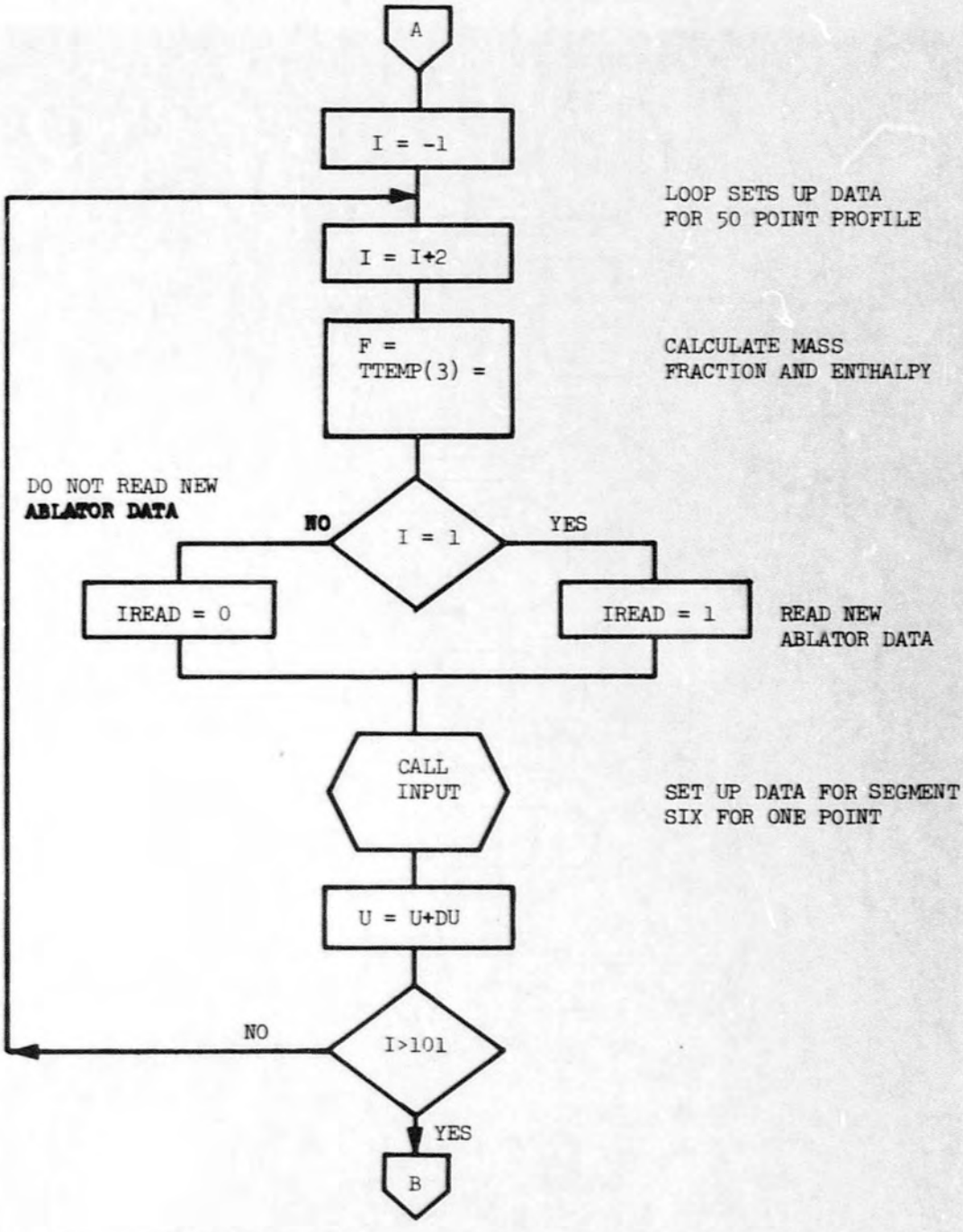
a. Function

Segment EIGHT is used when multiple chemical composition cases are desired for a number of "short" boundary layer cases; i.e., cases for which only 10 points were calculated instead of 50. Previously computed values of the boundary conditions of mass fraction, pressure, and enthalpy are read after the segment is entered. Using these values, the enthalpy profile is generated and the enthalpy, pressure, and mass fraction are used by Subroutine INPUT to prepare an input tape for Segment SIX, the chemical equilibrium calculation. As described earlier, Segment SIX calls Segment SEVEN, which prints the profile. Segment SEVEN calls Segment EIGHT, which reads new data; this process is repeated until an end of file is detected by the data read statement. This end of file causes the program to terminate.

b. Flow Chart



DEFINE PART OF TTEMP ARRAY,
CHANGE UNITS OF DENSITY,
AND INITIALIZE VELOCITY


A
I = -1
LOOP SETS UP DATA
FOR 50 POINT PROFILE
I = I+2
F =
TTEMP(3) =
CALCULATE MASS
FRACTION AND ENTHALPY
DO NOT READ NEW
ABLATOR DATA
NO
I = 1
YES
IREAD = 0
IREAD = 1
READ NEW
ABLATOR DATA
CALL
INPUT
SET UP DATA FOR SEGMENT
SIX FOR ONE POINT
U = U+DU
NO
I>101
YES
B

B
ENDFILE
TAPE 22
REWIND
TAPE 22
CALL
SEGMENT
SIX
END

```
C.  LISTING

      SEGMENT EIGHT(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
      COMMON M,L,NUM,MIN,LARG,NUP,KSAVE,NNOD
      COMMON          C       (    39),CON     (    70),CONOLD (    70),
     *                CONR    (    70),
     *CPP     (    70),
     *D       (    70),DELDH   (     2),DSO     (    70),
     *EFR     (70,39),EK      (70,39),ENT     (70,39),
     *FORM    (    70),FR      (    30),
     *IDRC    (    70),IDRE    (    15),IID     (    30),
     *NA      (    15),
     *PHI     (    70),
     *SB      (    70),SEK     (70,39),SEFR    (    70),SENT    (70,39),
     *SFCON   (    70),SFORM   (    70),STSB    (    15),STY     (70,15),
     *SWTM    (    70),MAXE    (     1),MAXS    (     1),JT      (     1),
     *TE      (    39),TTEMP   (     4),
     *WTM     (    70),WTR     (    70),
     *Y       (70,70),
     *Z       (    70)
      COMMON        YY,FS,RHOOW,TRS,CFS,ELS,HW,HAW,HE,RHOW,TITL(5),
     * C5,C6,UE,B,CIR
      DIMENSION YY(101),TRS(101),CFS(101),ELS(101) ,RHOOW(101)
      REWIND 22
      READ(5,1010) TITL
799   FORMAT(1H1,11X,5A10)
      IF (EOF,5) 6000,100
100   READ (5,1000) FS,HW,HAW,HE,RHOW,C5,C6,UE,B,CIR,T,P
      WRITE (6,799) TITL
      WRITE(6,800) FS,HW,HAW,HE,RHOW,C5,C6,UE,B,CIR,T,P
800   FORMAT(/,20X,3HFS=,E15.5/20X,3HHW=,E15.5/19X,4HHAW=,E15.5/20X,3HHE
     *=,E15.5/18X,5HRHOW=,E15.5/20X,3HC5=,E15.5/20X,3HC6=,E15.5/20X,
     *3HUE=,E15.5/21X,2HB=,E15.5/19X,4HCIR=,E15.5/21X,2HT=,E15.5/
     *21X,2HP=,E15.5)
      TTEMP(1) = T
      TTEMP(2) = P
      TTEMP(4) =0.0
      RHOD = RHOW * 16.018432
      DU = .02
      U = 0.0
      DO 400 I=1,99,2
      F = FS * (1.0-U)
      TTEMP(3) = HW + (HAW - HW) *U - (HAW-HE) * U**2
      IF (I-1) 220,230,230
```

```
  220 IREAD = 0
      GO TO 250
  230 IREAD = 1
  250 CALL INPUT (IREAD,F,RHOD,P,TTEMP,U,KODEW)
      U = U + DU
  400 CONTINUE
      ENDFILE 22
      REWIND 22
      CALL CHAIN (SIX)
 6000 CONTINUE
      CALL EXIT
 1000 FORMAT(10X,4F15.0)
 1010 FORMAT (5A10)
      END
```

d. Symbols

Symbolic representation	Meaning or use
TITL	Title of case
FS	Wall mass fraction
HW	Wall enthalpy
HAW	Adiabatic wall enthalpy
HE	Edge enthalpy
RHOW	Wall density (lb/ft^3)
C5	Constant
C6	Constant
UE	Edge velocity
B	Mass injection parameter
CIR	Circumferance of local body station
T	Wall temperature
P	Wall and boundary layer pressure
RHOD	Density (gm/l)
TTEMP	Array of values for chemistry calculations (state variables)
DU	Velocity ratio increment
U	Velocity ratio
F	Mass fraction
IREAD	0 - Subroutine INPUT does not read new ablator data 1 - Reads new ablator data

12. SEGMENT NINE

a. Function

If no ablation is occurring, this segment computes the chemical composition of a "clean" air boundary layer. The calculation procedure is identical to that of Segment SIX and reference 2. A listing of this segment is

included because the chemical library is obtained through a common block and small differences exist in the output statements. When the composition at all of the boundary layer points have been calculated, Segment SEVEN is called and the program terminates as previously described.

B. LISTING

```
      SEGMENT NINE(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,TAPE22)
C       T.E. HORTON  SHOCK AND CHEMICAL EQUILIBRIUM PROGRAM                    00000020
C
      COMMON /PERM/ C(    17),    CON(    17),CONCLD(    17),   CONR(    03),
     1              CPP(    17),      D(    17), DELDH(     2),    DSO(    17),
     2              EER(17, 1),     EK(17,39),    ENT(17,39),   FORM(    17),
     3               ER(    3),    IID(     3),    PHI(    17),     SB(    17),
     4            SECON(    17),  STSB(     4),    STY(17, 4),   MAXE(     1),
     5            MAXS(     1),     JT(     1),     TE(39    ), TTEMP(     4),
     6             WTM(    17),    WTR(    17),      Y(17,17),      Z(    17),
     7            NMOD         ,     M         ,      L         ,   KIP       ,
     8             JTP
C
      COMMON        YY,ES,PHOOW,TRS,CES,ELS,HZ,HAW,HE,RHOW,TITL(5),
     * C5,C6,UE,B,CIP
      DIMENSION YY(101),TRS(101),CES(101),ELS(101) ,RHOOW(101)
      EQUIVALENCE(MAXEX,MAXE(1)),(MAXSX,MAXS(1)),(JTX,JT(1))
501   FORMAT(3I2)
4104  FORMAT(15H THE VALUE JTP=,I4,2X,12HIS NOT VALID)                         00000130
6600  FORMAT(4X,1E12.6)                                                        160
7002  FORMAT (I2,A4,F12.6)
7004  FORMAT(12XA4,1512,F10.6)                                                 210
7008  FORMAT(20X,5E7.1)                                                        220
7009  FORMAT(20X,5E12.6)                                                       230
7110  FORMAT(2014)                                                             250
9301  FORMAT(3X,A4,2X,2E13.6)                                                  430
9620  FORMAT (1H1,30X,24HTURBULENT BOUNDARY LAYER//1H0,30X,5A10//
     * 1H0,9X,70H CHEMICAL EQUILIBRIUM COMPOSITION AND PROPERTIES OF A COM
     *PLEX MIXTURE //
     *       9H   RHOO =1E14.6,5H GM/L 7X,22HNUMBER OF ITERATIONS =112// 00000420
     *       10X,70HCONCENTRATION - MOLE/LITER  CONCENTRATION  MOLE FRAC00000490
     1TION  -LOGE(KP)      /                                                  500
     *       11X,40H(FINAL-1)        (FINAL)     PARTICLES/CC     )          00000510
9604  FORMAT(20H   FINAL CONDITIONS /                                          0450
     *       7X, 7HT, DEG K       RHO/RHOO        P/PO           Z           00000520
     1    SUM CON MOL/L /                                                     530
     *       6X,5E14.6////                                                    550
     *       10X,12H HMO/RTO      SMO/R/                                     0 00540
     *       6X,2E14.6////                                                    560
     *       26H   COLD MIXTURE MOLE WT. =F10.6////                           00000570
     *       26H     COLLISION FREQUENCY =1PE13.6////
     *       16X10HENTHALPY =1PE13.6,24HJOULES / CC      HCALC =E13.6//)
```

```
 9690 FORMAT(3X,A4,2X,5E14.6)                                                    480
 9693 FORMAT(3X,23HTOTAL CONCENTRATION OF A4,23HAT STANDARD CONDITIONS  00000580
     1 1F8.4 ,12H PER CENT     )                                                590
10009 FORMAT (I3,5E12.6)
      INTAP = 5
      IOTAP = 6
      REWIND 22
      INDEX=-1
      ICOUNT = 4
      NUM=L-M                                                                   790
      MIN=M                                                                     800
      LARG=L                                                                    810
      NUP=L-M+1                                                                 820
      ITER=0                                                                    870
      KIT=1                                                                     880
C
 6000 CONTINUE
 6001 CONTINUE
 6006 IST = 0
 6400 SCON=(1.0)/(0.0820575*273.15)                                             0930
      DO 7112 I=1,MAXEX                                                         940
 7112 STSB(I)=0.0                                                               950
      DO 7114 I=1,KIP                                                           960
      K=IID(I)                                                                  970
      DSST=CONR(I)*SCON                                                         980
      DO 7114 J=1,MAXEX                                                         990
 7114 STSB(J)=STSB(J)+(DSST*STY(K,J))                                   0 01000
      N=1                                                                      1010
      K=NUP                                                                    1020
      DO 6405 M=1,MAXEX                                                        1030
      LOPP=MAXS-MAXF+M                                                         1040
      IF (N+NUM - LOPP) 6405,6403,6405
 6403 I=M                                                                      1060
      DO 6404 L=1,LARG                                                         1070
      J=L
 6404 Y(L,K)=STY(J,I)                                                          1090
      SB(K)=STSB(M)                                                            1100
      N=N+1                                                                    1110
      K=K+1                                                                    1120
 6405 CONTINUE                                                                 1150
 7602 DS0=0.0                                                                  1380
      DO6407 I=NUP,LARG                                                        1390
 6407 DS0=DS0+(SB(I)*WTM(I))                                                   1400
      WWTSTP=DS0*273.15*0.0820575                                      0 01410
      DSAIR=(28.966348)/(273.15*0.0820575)                             00001420
11000 READ (22,10002) JTR,VPTO,TTEMP
```

```
      IF (EOF,22) 10333,10300
10333 REWIND 22
      CALL CHAIN (SEVEN)
10300 LCOUNT = 1
 5000 IF(JTR-70)5001,200,250
 5001 WRITE (IOTAP,4804) JTR
      CALL EXIT
  200 NTRIG=2
      DELT=0.1*TTEMP(1)
  201 TEMP=TTEMP(1)
      INDEX=INDEX + 2
      HZRO = TTEMP(3)
      PZRO=TTEMP(2)
      KNTRL=1
  202 PORT=TTEMP(2)/TEMP/0.0820575
      JITER=0
      GO TO 6055
  250 IF (JTR - 80) 5001,251,262
  262 IF (JTR - 90) 5001,265,275
  275 TEMP = TTEMP(1)
      NTRIG = 1
      GO TO 201
  265 DELT  = 0.1 * TTEMP (1)
      DELRO = .0001 * VRTO
      NTRIG = 4
      GO TO 201
  251 NTRIG=3
      DELPO=.0001*VRTO
      GO TO 201
10005 IF(TTEMP(ICOUNT+1))10000,10000,10006
10006 ICOUNT = ICOUNT+1
      TEMP = TTEMP(ICOUNT)
 6055 CONTINUE
 6500 AIR=28.966348                                                    1620
C         SET INITIAL VALUES                                          1750
 5511 JAC = 0                                                         1760
      DO 7012 I=1,LARG                                                1770
      PHI(I)=0.0                                                      1780
      D(I)=0.0                                                        1790
C         CALCULATION OF SPECIFIC ENTHALPIES AND EQUILIBRIUM CONSTANTS  00001800
      CALL DISCT(TEMP,TE,ENT,I,39,000)
7012  CPP(I) = 000*TEMP + 1000.*EFR(I,1)
      DO 7007 I=1,NUM                                                 1920
      CALL DISCT(TEMP,TE,EK,I,39,000)
      PHI(I) = -(000-EFR(I,1)*1000./TEMP)/1.98726
      DO 7007 K=NUP,LARG                                              1940
```

```
      CALL DISCT(TEMP,TF,EK,K,39,QQQ)
      BBR = (QQQ-EFR(K,1)*1000./TEMP)/1.98726
 7007 PHI(I)=PHI(I) +Y(I,K)*BBR                                          1860
C        TEST FOR FIRST RUN KIT=1 FOR FIRST RUN,KIT=2 FOR ALL OTHERS     00001870
      GO TO(7901,9200),KIT                                               1880
C        CALCULATION OF INITIAL GUESS FOR FIRST RUN                      00001890
 7901 TF=ALOG(TEMP*0.0820575)                                            1900
      VRTO = PORT * WWTSTP / DSAIR
      KIT=2                                                              1910
      DO 5777 K=1,NUM                                                    1920
      D(K) = -(PHI(K)+TF)
      DO 5797 I=NUP,LARG                                                 1940
 5797 D(K) = D(K) + ((TF+ALOG(CON(I)))*Y(K,I))
 5777 CON(K) = EXPX(5777,D(K))
C        BEGINNING OF EQUILIBRIUM ITERATION LOOP                         00001980
 9200 JAC=JAC+1                                                          1990
      VOL=WWTSTP/(AIR*VRTO)                                              1630
C        TEST FOR NUMBER OF ITERATIONS                                   00002000
      IF(JAC-150)5003,5003,5991
C        EQUILIBRIUM BLOCK NOT CONVERGENT BLOCK                          00002020
 5002 FORMAT(48H EQUILIBRIUM STATE NOT REACHED IN 150 ITERATIONS)
 5991 WRITE (IOTAP,5002)
      DO 5004 I=1,LARG                                                   2050
      WRITE (IOTAP,9601) FORM(I),CON(I),D(I)
 5004 WRITE (IOTAP,9601) FORM(I),CONOLD(I),DSO(I)
      GO TO 6000
C        EQUILIBRIUM BLOCK                                               2090
 5003 COLD=CS2                                                           2100
      HOLD=H2                                                            2110
      DOLD=DS2                                                           2120
      DO 9514 I=1,LARG                                                   2130
      DSO(I)=D(I)                                                        2140
 9514 CONOLD(I)=CON(I)                                                   2150
      DO 9203 I=1,MIN                                                    2160
      DO 9203 J=1,MIN                                                    2170
      Y(I,J)=0.0                                                         2180
      M=NUM+I                                                            2190
      N=NUM+J                                                            2200
C
      DO 9202 K=1,NUM                                                    2210
 9202 Y(I,J)=Y(I,J)+Y(K,M)*Y(K,N)*CON(K)                                 0002220
      IF(I-J) 9203,9201,9203                                             2230
C
 9201 Y(I,J)=Y(I,J)+CON(M)                                               2240
 9203 CONTINUE                                                           2250
      TF=ALOG(TEMP*0.0820575)                                            02260
```

```
      DO 9205 K=1,NUM                                                    2270
      IF(CON(K)) 9313,9312,9313                                         0 02280
 9312 ZIGG=0.0                                                           2290
      GO TO 9215                                                         2300
 9313 ZIGG=ALOG(CON(K))                                                  2310
C
 9215 D(K)=PHI(K)+TF+ZIGG                                                2320
      DO 9205 I=NUP,LARG                                                 2330
      IF(CON(I)) 9311,9310,9311                                         0 02340
 9310 ZAGG=0.0                                                           2350
C
      GO TO 9205                                                         2360
 9311 ZAGG=ALOG(CON(I))                                                  2370
 9205 D(K)=D(K)-((TF+  ZAGG        )*Y(K,I))                           00002380
      DO 9206 I=NUP,LARG                                                 2390
      D(I)=CON(I)-(SB(I)/VOL)                                            2400
      DO 9206 K=1,NUM                                                    2410
 9206 D(I)=D(I)+(CON(K)*Y(K,I))                                         0 02420
      DO 9207 I=1,MIN                                                    2430
      K=NUM+1                                                            2440
      C(I)=D(K)                                                          2450
      DO 9207 J=1,NUM                                                    2460
 9207 C(I)=C(I)-D(J)*CON(J)*Y(J,K)                                      0 02470
      DO 9301 K=2,MIN                                                    2480
 9301 Y(1,K)=Y(1,K)/Y(1,1)                                               2490
      C(1)=C(1)/Y(1,1)                                                   2500
      DO 9306 L=2,MIN                                                    2510
      N=L-1                                                              2520
      DO 9302 J=L,MIN                                                    2530
      DO 9302 K=1,N                                                      2540
 9302 Y(J,L)=Y(J,L)-Y(J,K)*Y(K,L)                                       0 02550
      LA=L+1                                                             2560
      DO 9304 I=LA,MIN                                                   2570
      DO 9303 K=1,N                                                      2580
      Y(L,I) = Y(L,I) - (Y(L,K)*Y(K,I))
 9303 CONTINUE
C
      Y(L,I) = Y(L,I)/Y(L,L)
 9304 CONTINUE
      DO 9305 K=1,N                                                      2610
 9305 C(L)=C(L)-(Y(L,K)*C(K))                                           02620
 9306 C(L)=C(L)/Y(L,L)                                                   2630
      M=MIN                                                              2640
      Z(M)=-C(M)                                                         2650
      DO 9307 I=2,MIN                                                    2660
      N=MIN+1-I                                                          2670
```

```
      NET=N+1                                                          2680
      Z(N)=-C(N)                                                       2690
      DO 9307 K=NET,MIN                                                2700
 9307 Z(N)=Z(N)-Y(N,K)*Z(K)                                            2710
      DO 9501 I=1,MIN                                                  2720
      K=LARG-MIN+I                                                     2730
 9501 Z(K)=Z(I)                                                        2740
      DO 9503 I=1,NUM                                                  2750
      Z(I)=-D(I)                                                       2760
      DO 9503 J=NUP,LARG                                               2770
 9503 Z(I)=Z(I)+Y(I,J)*Z(J)                                            2780
      DO 9602 I=1,LARG                                                 2790
      CON(I) = EXPX(9503,Z(I)) * CON(I)
      IF(CON(I))9602,9506,9507                                        02810
 9506 CON(I) = 1.000E-35                                               2812
      GO TO 9602                                                       2814
 9507 IF(CON(I) - .1)9602,9602,9508                                 00002816
 9508 CON(I) = .1                                                      2818
 9602 CONTINUE                                                         2830
C         CALCULATION OF THERMODYNAMIC PROPERTIES                    00002840
      CS2=0.0                                                          2850
      H2=0.0                                                           2860
      DS2=0.0                                                          2870
      DO 9072 I=1,LARG                                                 2880
      CS2=CS2+CON(I)                                                   2890
      H2=H2+(CPP(I)*CON(I))                                            2900
 9072 DS2=DS2+(WTM(I)*CON(I))                                         02910
      VRTO=PORT*DS2/(CS2*DSAIR)
C         TEST FOR CONVERGENCE OF EQUILIBRIUM PROPERTIES             00002920
 4066 TES1=1.0-(COLD/CS2)                                              2930
      ZT1=ABS(TES1)                                                    2940
      IF(ZT1-0.000001) 9901, 9901, 9211                          00002950
 9901 TES2=1.0-(DOLD/DS2)                                              2960
      ZT2=ABS(TES2)                                                    2970
      IF(ZT2-0.000001) 9902, 9902, 9211                          00002980
 9902 TES3=1.0-(HOLD/H2)                                               2990
      ZT3=ABS(TES3)                                                    3000
      IF(ZT3-0.000001) 9903, 9903, 9211                          00003010
 9211 IF(JAC-4) 9221, 9221, 9200                                      03020
 9221 DO 9220 I=1,LARG                                                3030
 9220 CON(I)=(CONOLD(I)+CON(I))/2.0                                 0 03040
      GO TO 9200                                                       3050
 9903 IF(JITER)9904,9904,9905
 9904 JITER=1
      GO TO 9211
 9905 CONTINUE
```

```
      H2=H2/DS2
      HCALC = H2 * 1.8
      T2=TEMP                                                          3070
      P2=(1.98726*T2*CS2)                                              3080
      PRES2=(T2*CS2*0.0820575)                                        03090
      GO TO (299,259,299,500),NTRIG
  259 GO TO (260,270),KNTRL
  260 KNTRL=2
  261 PTWO=PRES2
      TA=TEMP
      TEMP=TEMP+DELT
  300 LCOUNT =LCOUNT +1
      IF(LCOUNT-150)202,202,301
  301 WRITE (IOTAP,302)
  302 FORMAT(41H CASE DID NOT CONVERGE IN 150 ITERATIONS.)
      GO TO 299
  270 DELP=(PRES2-PTWO)/DELT
      DELT=(PZRO-PRES2)/DELP
      TTEST=DELT/TA
      IF(ABS(TTEST)-.0001)299,299,261
  500 GO TO (510,520),KNTRL
  510 KNTRL = 2
  530 HTWO=HCALC
      TA=TEMP
      TEMP=TEMP+DELT
      IF (TEMP) 1263,1263,300
 1263 TEMP = 0.9 * TA
      DELT = TEMP - TA
      GO TO 300
  520 DELH=(HCALC-HTWO)/DELT
      DELT=(HZRO-HCALC)/DELH
      TTEST=1.0-(HZRO/HCALC)
      IF(ABS(TTEST)-0.000001)299,299,530
  299 ZSA=(PRES2*273.15*VOL)/T2
      ENTHA=H2/(1.98726*273.15)*WWTSTP                               0003110
      ENTRO=-ALOG(PRES2)                                               3120
      DO 9695 I=1,LARG                                                 3130
      AFR=CON(I)/CS2                                                   3140
      IF(AFR)9694,9696,9694                                            3150
 9696 TOM=0.0                                                          3160
      GO TO 19695
 9694 TOM=ALOG(AFR)                                                    3180
19695 CALL DISCT(TEMP,TE,ENT,I,39,000)
      CALL DISCT(TEMP,TE,EK,I,39,001)
 9695 ENTRO=ENTRO+AFR*((000+001)/1.98726 -TOM)
      ENTRO=ENTRO*(CS2/DS2)*WWTSTP                                   03200
```

```
C         EQUILIBRIUM SOLUTION PRINT                                          0003960
 9090 WRITE (6,9620) TITL, DSAIP,JAC
      DO 9095  I=1,LARG                                                        4010
      CONFR=CON(I)/CS2                                                         4020
      CPMOL=CON(I)*.60233BE+21                                                 04030
 9095 WRITE (IOTAP,9690) FORM(I),CONOLD(I),CON(I),CPMOL,CONFR,PHI(I)
      VPP=VPTO                                                                 4070
      CF=5.814E12*PRES2/SQRT(T2)
      ENTHAL = 2.93641 * ENTHA * VPP / WWTSTP                                  00004116
      WRITE (IOTAP,9604) T2,VPP,PRES2,ZSA,CS2,ENTHA,ENTRO,WWTSTP,CF,
     *ENTHAL, HCALC
      DO 9699 I=1,KIP                                                          4120
      ABCE=CONR(I)*100.0                                                       4130
 9699 WRITE (IOTAP,9693) FR(I),ABCE
      ELS(INDEX) = CON(NNOD) * .602338E+21
      TPS(INDEX) = T2
      CFS(INDEX) = CF
      PHOOW(INDEX) = VPP / PHOW * 1.29233 / 16.018432
      GO TO 6000
      END                                                                      -
```

c. Block Data PERM

This block data package contains the thermodynamic library data for the clean air model, and consists of data for the following species:

NO	0-	N+	E-
NO2	02-	0+	A
N20	NO+	A+	
N2	N2+	N	
02	02+	0	

```
      BLOCK DATA
      COMMON /PERM/ C(   17),   CON(   17),CONOLD(   17),  CONR(    3),
     1             CPP(   17),     D(   17), DELDH(    2),   DSO(   17),
     2            EFR(17, 1),    EK(17,39),   ENT(17,39),  FORM(   17),
     3              FR(    3),   IID(    3),   PHI(   17),    SB(   17),
     4           SFCON(   17),  STSB(    4),   STY(17, 4),  MAXE(    1),
     5            MAXS(    1),    JT(    1),    TE(39   ), TTEMP(    4),
     6            WTM(   17),   WTR(   17),     Y(17,17),     Z(   17),
     7           NNOD        ,     M       ,     L       ,   KIP       ,
     8            JTR
      DATA MAXE, MAXS, M, L, KIP, JT, NNOD
     1 / 4, 17, 4, 17, 3, 39, 16 /
C
      DATA TE
     1 /  1000.,    1500.,    2000.,    2500.,    3000.,
     2    3500.,    4000.,    4500.,    5000.,    5500.,
     3    6000.,    6500.,    7000.,    7500.,    8000.,
     4    8500.,    9000.,    9500.,   10000.,   10500.,
     5   11000.,   11500.,   12000.,   12500.,   13000.,
     6   13500.,   14000.,   14500.,   15000.,   15500.,
     7   16000.,   16500.,   17000.,   17500.,   18000.,
     8   18500.,   19000.,   19500.,   20000./
C
      DATA (CON(I),I=14,17), FR, IID, CONR(1), CONR(2), CONR(03)
     1 / 1.0E-4,   1.0E-4,   1.0E-4,   1.0E-4,
     2     2HN2,     2HO2,      1HA,
     3        4,        5,       17,
     4 0.780881, 0.209795, 0.009324 /
C
      DATA FORM(  1),(STY(I),I=  1,68,17), WTM(  1) /
     * 4HNO  , 1., 1., 0., 0.,    30.007996/
      DATA (EK(I),I=  1,663,17)/
     * 0.520302E 02, 0.550782E 02, 0.573251E 02, 0.591150E 02,
     * 0.606060E 02, 0.618850E 02, 0.630058E 02, 0.640036E 02,
     * 0.649032E 02, 0.657224E 02, 0.664747E 02, 0.671703E 02,
     * 0.678175E 02, 0.684227E 02, 0.689914E 02, 0.695089E 02,
     * 0.700264E 02, 0.705191E 02, 0.709896E 02, 0.714400E 02,
     * 0.718723E 02, 0.722880E 02, 0.726886E 02, 0.730752E 02,
     * 0.734489E 02, 0.738107E 02, 0.741614E 02, 0.745016E 02,
     * 0.748322E 02, 0.751536E 02, 0.754663E 02, 0.757709E 02,
     * 0.760678E 02, 0.763573E 02, 0.766398E 02, 0.769156E 02,
     * 0.771850E 02, 0.774483E 02, 0.777057E 02/
      DATA (ENT(I),I=  1,663,17)/
     * 0.735381E 01, 0.768780E 01, 0.793139E 01, 0.810929E 01,
     * 0.824402E 01, 0.834986E 01, 0.843571E 01, 0.850727E 01,
     * 0.856834E 01, 0.863500E 01, 0.870000E 01, 0.876500E 01,
     * 0.883000E 01, 0.889000E 01, 0.896696E 01, 0.902488E 01,
     * 0.908353E 01, 0.914283E 01, 0.920256E 01, 0.926252E 01,
     * 0.932249E 01, 0.938224E 01, 0.944153E 01, 0.950013E 01,
     * 0.955785E 01, 0.961449E 01, 0.966984E 01, 0.972375E 01,
     * 0.977604E 01, 0.982656E 01, 0.987517E 01, 0.992173E 01,
     * 0.996611E 01, 0.100082E 02, 0.100480E 02, 0.100852E 02,
     * 0.101200E 02, 0.101522E 02, 0.101819E 02/
      DATA EFR(  1) /   0.214770E 02 /
      DATA FORM(  2),(STY(I),I=  2,68,17), WTM(  2) /
     * 4HNO2 , 1., 2., 0., 0.,     46.007996/
      DATA (EK(I),I=  2,663,17)/
     * 0.600121E 02, 0.643205E 02, 0.675903E 02, 0.702328E 02,
     * 0.724518E 02, 0.743649E 02, 0.760464E 02, 0.775463E 02,
```

```
*  0.789002E 02,  0.801339E 02,  0.812671E 02,  0.823150E 02,
*  0.832894E 02,  0.842000E 02,  0.850547E 02,  0.858598E 02,
*  0.866210E 02,  0.873426E 02,  0.880287E 02,  0.887065E 02,
*  0.893528E 02,  0.899704E 02,  0.905617E 02,  0.911288E 02,
*  0.916737E 02,  0.921980E 02,  0.927033E 02,  0.931908E 02,
*  0.936618E 02,  0.941174E 02,  0.945585E 02,  0.949860E 02,
*  0.954007E 02,  0.958034E 02,  0.961948E 02,  0.965755E 02,
*  0.969460E 02,  0.973069E 02,  0.976586E 02/
 DATA (ENT(I),I=  2,663,17)/
*  0.101652E 02,  0.110735E 02,  0.116454E 02,  0.120305E 02,
*  0.123055E 02,  0.125111E 02,  0.126705E 02,  0.127974E 02,
*  0.129009E 02,  0.129869E 02,  0.130594E 02,  0.131214E 02,
*  0.131751E 02,  0.132219E 02,  0.132631E 02,  0.132996E 02,
*  0.133323E 02,  0.133617E 02,  0.133882E 02,  0.134122E 02,
*  0.134341E 02,  0.134540E 02,  0.134723E 02,  0.134891E 02,
*  0.135047E 02,  0.135191E 02,  0.135324E 02,  0.135448E 02,
*  0.135565E 02,  0.135673E 02,  0.135775E 02,  0.135871E 02,
*  0.135961E 02,  0.136045E 02,  0.136126E 02,  0.136201E 02,
*  0.136273E 02,  0.136341E 02,  0.136406E 02/
 DATA EFR(  2) /   0.876600E 01 /
 DATA FORM(  3),(STY(I),I=  3,68,17), WTM(  3) /
*  4HN2O , 2., 1., 0., 0.,     44.015991/
 DATA (EK(I),I=  3,663,17)/
*  0.557014E 02,  0.601498E 02,  0.635538E 02,  0.663192E 02,
*  0.686500E 02,  0.706649E 02,  0.724395E 02,  0.740252E 02,
*  0.754583E 02,  0.767657E 02,  0.779676E 02,  0.790798E 02,
*  0.801148E 02,  0.810826E 02,  0.819913E 02,  0.828479E 02,
*  0.836579E 02,  0.844261E 02,  0.851567E 02,  0.858826E 02,
*  0.865748E 02,  0.872363E 02,  0.878695E 02,  0.884769E 02,
*  0.890605E 02,  0.896221E 02,  0.901632E 02,  0.906854E 02,
*  0.911898E 02,  0.916777E 02,  0.921501E 02,  0.926080E 02,
*  0.930522E 02,  0.934835E 02,  0.939027E 02,  0.943104E 02,
*  0.947072E 02,  0.950937E 02,  0.954704E 02/
 DATA (ENT(I),I=  3,663,17)/
*  0.104381E 02,  0.114904E 02,  0.121602E 02,  0.126164E 02,
*  0.129450E 02,  0.131921E 02,  0.133845E 02,  0.135383E 02,
*  0.136640E 02,  0.137686E 02,  0.138570E 02,  0.139327E 02,
*  0.139982E 02,  0.140555E 02,  0.141060E 02,  0.141508E 02,
*  0.141908E 02,  0.142269E 02,  0.142594E 02,  0.142889E 02,
*  0.143157E 02,  0.143402E 02,  0.143627E 02,  0.143834E 02,
*  0.144025E 02,  0.144201E 02,  0.144366E 02,  0.144518E 02,
*  0.144661E 02,  0.144794E 02,  0.144919E 02,  0.145037E 02,
*  0.145147E 02,  0.145252E 02,  0.145350E 02,  0.145443E 02,
*  0.145531E 02,  0.145615E 02,  0.145695E 02/
 DATA EFR(  3) /   0.203090E 02 /
 DATA FORM(  4),(STY(I),I=  4,68,17), WTM(  4) /
*  4HN2  , 2., 0., 0., 0.,     28.015991/
 DATA (EK(I),I=  4,663,17)/
*  0.473055E 02,  0.502839E 02,  0.524769E 02,  0.542261E 02,
*  0.556854E 02,  0.569392E 02,  0.580391E 02,  0.590194E 02,
*  0.599037E 02,  0.607096E 02,  0.614499E 02,  0.621347E 02,
*  0.627719E 02,  0.633681E 02,  0.639283E 02,  0.644571E 02,
*  0.649582E 02,  0.654348E 02,  0.658800E 02,  0.663100E 02,
*  0.666359E 02,  0.670440E 02,  0.674388E 02,  0.678218E 02,
*  0.681942E 02,  0.685573E 02,  0.689120E 02,  0.692591E 02,
*  0.695993E 02,  0.699332E 02,  0.702613E 02,  0.705839E 02,
*  0.709015E 02,  0.712142E 02,  0.715224E 02,  0.718260E 02,
*  0.721254E 02,  0.724204E 02,  0.727113E 02/
 DATA (ENT(I),I=  4,663,17)/
*  0.720392E 01,  0.750194E 01,  0.774497E 01,  0.793134E 01,
*  0.807572E 01,  0.819024E 01,  0.828331E 01,  0.836065E 01,
*  0.842622E 01,  0.848290E 01,  0.853293E 01,  0.857816E 01,
```

```
     * 0.862029E 01, 0.866094E 01, 0.870169E 01, 0.874411E 01,
     * 0.878970E 01, 0.883986E 01, 0.891000E 01, 0.899900E 01,
     * 0.909900E 01, 0.920000E 01, 0.932708E 01, 0.943825E 01,
     * 0.955813E 01, 0.968605E 01, 0.982116E 01, 0.996234E 01,
     * 0.101084E 02, 0.102580E 02, 0.104098E 02, 0.105623E 02,
     * 0.107144E 02, 0.108645E 02, 0.110117E 02, 0.111547E 02,
     * 0.112927E 02, 0.114248E 02, 0.115504E 02/
      DATA EFR(  4) /   0.0                 /
      DATA FORM(  5),(STY(I),I=  5,68,17), WTM(  5) /
     * 4HO2  , 0., 2., 0., 0.,     32.000000/
      DATA (EK(I),I=  5,663,17)/
     * 0.506889E 02, 0.538017E 02, 0.560979E 02, 0.579308E 02,
     * 0.594640E 02, 0.607868E 02, 0.619530E 02, 0.629974E 02,
     * 0.639015E 02, 0.647769E 02, 0.655840E 02, 0.663350E 02,
     * 0.670379E 02, 0.676989E 02, 0.683233E 02, 0.689150E 02,
     * 0.694775E 02, 0.700138E 02, 0.705261E 02, 0.710166E 02,
     * 0.714870E 02, 0.719390E 02, 0.723738E 02, 0.727927E 02,
     * 0.731966E 02, 0.735866E 02, 0.739635E 02, 0.743281E 02,
     * 0.746810E 02, 0.750228E 02, 0.753543E 02, 0.756758E 02,
     * 0.759879E 02, 0.762911E 02, 0.765856E 02, 0.768720E 02,
     * 0.771506E 02, 0.774217E 02, 0.776857E 02/
      DATA (ENT(I),I=  5,663,17)/
     * 0.750328E 01, 0.785394E 01, 0.811068E 01, 0.831960E 01,
     * 0.850109E 01, 0.866200E 01, 0.880473E 01, 0.893084E 01,
     * 0.907500E 01, 0.919900E 01, 0.931120E 01, 0.943459E 01,
     * 0.953473E 01, 0.962934E 01, 0.971857E 01, 0.980247E 01,
     * 0.988101E 01, 0.995415E 01, 0.100219E 02, 0.100842E 02,
     * 0.101411E 02, 0.101928E 02, 0.102394E 02, 0.102809E 02,
     * 0.103175E 02, 0.103495E 02, 0.103769E 02, 0.104000E 02,
     * 0.104189E 02, 0.104338E 02, 0.104449E 02, 0.104524E 02,
     * 0.104565E 02, 0.104573E 02, 0.104551E 02, 0.104501E 02,
     * 0.104424E 02, 0.104321E 02, 0.104196E 02/
      DATA EFR(  5) /   0.0                 /
      DATA FORM(  6),(STY(I),I=  6,68,17), WTM(  6) /
     * 4HO-  , 0., 1., 1., 0.,     16.000549/
      DATA (EK(I),I=  6,663,17)/
     * 0.388608E 02, 0.408752E 02, 0.423045E 02, 0.434131E 02,
     * 0.443189E 02, 0.450847E 02, 0.457481E 02, 0.463333E 02,
     * 0.468567E 02, 0.473303E 02, 0.477625E 02, 0.481602E 02,
     * 0.485284E 02, 0.488711E 02, 0.491918E 02, 0.494930E 02,
     * 0.497769E 02, 0.500456E 02, 0.503004E 02, 0.505428E 02,
     * 0.507739E 02, 0.509948E 02, 0.512062E 02, 0.514090E 02,
     * 0.516039E 02, 0.517914E 02, 0.519720E 02, 0.521464E 02,
     * 0.523148E 02, 0.524777E 02, 0.526354E 02, 0.527883E 02,
     * 0.529366E 02, 0.530807E 02, 0.532206E 02, 0.533567E 02,
     * 0.534892E 02, 0.536183E 02, 0.537441E 02/
      DATA (ENT(I),I=  6,663,17)/
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
     * 0.496815E 01, 0.496815E 01, 0.496815E 01/
      DATA EFR(  6) /   0.251935E 02 /
      DATA FORM(  7),(STY(I),I=  7,68,17), WTM(  7) /
     * 4HO2- , 0., 2., 1., 0.,     32.000000/
      DATA (EK(I),I=  7,663,17)/
     * 0.517626E 02, 0.549472E 02, 0.572891E 02, 0.591492E 02,
```

```
     * 0.606968E 02, 0.620265E 02, 0.631962E 02, 0.642441E 02,
     * 0.653263E 02, 0.662048E 02, 0.670214E 02, 0.677862E 02,
     * 0.685066E 02, 0.691881E 02, 0.698350E 02, 0.704507E 02,
     * 0.710378E 02, 0.715986E 02, 0.721349E 02, 0.726483E 02,
     * 0.731403E 02, 0.736122E 02, 0.740651E 02, 0.745001E 02,
     * 0.749182E 02, 0.753203E 02, 0.757073E 02, 0.760801E 02,
     * 0.764394E 02, 0.767859E 02, 0.771203E 02, 0.774432E 02,
     * 0.777552E 02, 0.780569E 02, 0.783488E 02, 0.786315E 02,
     * 0.789053E 02, 0.791707E 02, 0.794280E 02/
      DATA (ENT(I),I=  7,663,17)/
     * 0.767920E 01, 0.802564E 01, 0.825221E 01, 0.841903E 01,
     * 0.856021E 01, 0.869406E 01, 0.882922E 01, 0.898000E 01,
     * 0.915000E 01, 0.930102E 01, 0.947134E 01, 0.963991E 01,
     * 0.980202E 01, 0.995394E 01, 0.100929E 02, 0.102169E 02,
     * 0.103250E 02, 0.104167E 02, 0.104923E 02, 0.105523E 02,
     * 0.105979E 02, 0.106301E 02, 0.106502E 02, 0.106594E 02,
     * 0.106590E 02, 0.106501E 02, 0.106338E 02, 0.106111E 02,
     * 0.105830E 02, 0.105502E 02, 0.105135E 02, 0.104734E 02,
     * 0.104307E 02, 0.103858E 02, 0.103391E 02, 0.102909E 02,
     * 0.102417E 02, 0.101917E 02, 0.101412E 02/
      DATA EFR(  7) /  -0.230000E 02 /
      DATA FORM(  8),(STY(I),I=  8,68,17), WTM(  8) /
     * 4HNO+ , 1., 1.,-1., 0.,     38.007996/
      DATA (EK(I),I=  8,663,17)/
     * 0.488839E 02, 0.518608E 02, 0.540527E 02, 0.558014E 02,
     * 0.572606E 02, 0.585145E 02, 0.596147E 02, 0.605953E 02,
     * 0.614803E 02, 0.622870E 02, 0.630285E 02, 0.637153E 02,
     * 0.643553E 02, 0.649553E 02, 0.655209E 02, 0.660567E 02,
     * 0.665667E 02, 0.670541E 02, 0.675220E 02, 0.679990E 02,
     * 0.684100E 02, 0.688600E 02, 0.692510E 02, 0.697786E 02,
     * 0.701736E 02, 0.705604E 02, 0.709400E 02, 0.713128E 02,
     * 0.716793E 02, 0.720398E 02, 0.723946E 02, 0.727439E 02,
     * 0.730879E 02, 0.734265E 02, 0.737599E 02, 0.740880E 02,
     * 0.744110E 02, 0.747289E 02, 0.750415E 02/
      DATA (ENT(I),I=  8,663,17)/
     * 0.720056E 01, 0.749812E 01, 0.774206E 01, 0.792974E 01,
     * 0.807553E 01, 0.819147E 01, 0.828602E 01, 0.836514E 01,
     * 0.843335E 01, 0.849437E 01, 0.855154E 01, 0.860794E 01,
     * 0.866644E 01, 0.872961E 01, 0.879968E 01, 0.887850E 01,
     * 0.896749E 01, 0.906758E 01, 0.917927E 01, 0.928500E 01,
     * 0.941000E 01, 0.960000E 01, 0.980999E 01, 0.998099E 01,
     * 0.101593E 02, 0.103434E 02, 0.105302E 02, 0.107176E 02,
     * 0.109038E 02, 0.110869E 02, 0.112654E 02, 0.114378E 02,
     * 0.116030E 02, 0.117600E 02, 0.119080E 02, 0.120466E 02,
     * 0.121755E 02, 0.122943E 02, 0.124032E 02/
      DATA EFR(  8) /   0.234800E 03 /
      DATA FORM(  9),(STY(I),I=  9,68,17), WTM(  9) /
     * 4HN2+ , 2., 0.,-1., 0.,     28.015991/
      DATA (EK(I),I=  9,663,17)/
     * 0.487704E 02, 0.517713E 02, 0.539883E 02, 0.557708E 02,
     * 0.572779E 02, 0.585943E 02, 0.597689E 02, 0.608319E 02,
     * 0.618035E 02, 0.626982E 02, 0.635268E 02, 0.642979E 02,
     * 0.650184E 02, 0.656942E 02, 0.663259E 02, 0.669303E 02,
     * 0.674984E 02, 0.680374E 02, 0.685988E 02, 0.690922E 02,
     * 0.695642E 02, 0.700165E 02, 0.704507E 02, 0.708681E 02,
     * 0.712701E 02, 0.716577E 02, 0.720318E 02, 0.723934E 02,
     * 0.727432E 02, 0.730819E 02, 0.734101E 02, 0.737284E 02,
     * 0.740373E 02, 0.743375E 02, 0.746290E 02, 0.749125E 02,
     * 0.751884E 02, 0.754569E 02, 0.757184E 02/
      DATA (ENT(I),I=  9,663,17)/
     * 0.724876E 01, 0.756875E 01, 0.785459E 01, 0.813177E 01,
     * 0.840849E 01, 0.867476E 01, 0.891811E 01, 0.913095E 01,
```

```
     * 0.931122E 01, 0.946064E 01, 0.958280E 01, 0.968189E 01,
     * 0.976194E 01, 0.982652E 01, 0.987866E 01, 0.995000E 01,
     * 0.100000E 02, 0.100510E 02, 0.100972E 02, 0.101295E 02,
     * 0.101603E 02, 0.101886E 02, 0.102146E 02, 0.102383E 02,
     * 0.102599E 02, 0.102792E 02, 0.102963E 02, 0.103111E 02,
     * 0.103236E 02, 0.103338E 02, 0.103416E 02, 0.103471E 02,
     * 0.103503E 02, 0.103512E 02, 0.103499E 02, 0.103465E 02,
     * 0.103410E 02, 0.103336E 02, 0.103243E 02/
      DATA EFR(  9) /   0.359306E 03 /
      DATA FORM( 10),(STY(I),I= 10,68,17), WTM( 10) /
     * 4HO2+ , 0., 2.,-1., 0.,    32.000000/
      DATA (EK(I),I= 10,663,17)/
     * 0.508912E 02, 0.539459E 02, 0.561981E 02, 0.579924E 02,
     * 0.594870E 02, 0.607694E 02, 0.618932E 02, 0.628941E 02,
     * 0.637971E 02, 0.646204E 02, 0.653779E 02, 0.660806E 02,
     * 0.667369E 02, 0.673541E 02, 0.679376E 02, 0.685388E 02,
     * 0.690749E 02, 0.695898E 02, 0.700860E 02, 0.705656E 02,
     * 0.710300E 02, 0.714806E 02, 0.719184E 02, 0.723442E 02,
     * 0.727586E 02, 0.731622E 02, 0.735555E 02, 0.739387E 02,
     * 0.743123E 02, 0.746764E 02, 0.750314E 02, 0.753775E 02,
     * 0.757149E 02, 0.760439E 02, 0.763646E 02, 0.766774E 02,
     * 0.769823E 02, 0.772797E 02, 0.775698E 02/
      DATA (ENT(I),I= 10,663,17)/
     * 0.736820E 01, 0.770560E 01, 0.795043E 01, 0.812917E 01,
     * 0.826480E 01, 0.837189E 01, 0.845994E 01, 0.853581E 01,
     * 0.860505E 01, 0.867240E 01, 0.874192E 01, 0.882000E 01,
     * 0.892500E 01, 0.905000E 01, 0.917867E 01, 0.931008E 01,
     * 0.945087E 01, 0.959881E 01, 0.975151E 01, 0.990641E 01,
     * 0.100610E 02, 0.102129E 02, 0.103600E 02, 0.105005E 02,
     * 0.106330E 02, 0.107561E 02, 0.108693E 02, 0.109720E 02,
     * 0.110640E 02, 0.111452E 02, 0.112159E 02, 0.112764E 02,
     * 0.113272E 02, 0.113688E 02, 0.114016E 02, 0.114264E 02,
     * 0.114438E 02, 0.114542E 02, 0.114584E 02/
      DATA EFR( 10) /   0.277918E 03 /
      DATA FORM( 11),(STY(I),I= 11,68,17), WTM( 11) /
     * 4HN+  , 1., 0.,-1., 0.,    14.007500/
      DATA (EK(I),I= 11,663,17)/
     * 0.390197E 02, 0.411164E 02, 0.425873E 02, 0.437211E 02,
     * 0.446443E 02, 0.454235E 02, 0.460984E 02, 0.466945E 02,
     * 0.472288E 02, 0.477137E 02, 0.481579E 02, 0.485680E 02,
     * 0.489492E 02, 0.493055E 02, 0.496401E 02, 0.499556E 02,
     * 0.502541E 02, 0.505374E 02, 0.508070E 02, 0.510643E 02,
     * 0.513102E 02, 0.515459E 02, 0.517722E 02, 0.519898E 02,
     * 0.521993E 02, 0.524013E 02, 0.525964E 02, 0.527851E 02,
     * 0.529677E 02, 0.531447E 02, 0.533163E 02, 0.534830E 02,
     * 0.536450E 02, 0.538026E 02, 0.539560E 02, 0.541055E 02,
     * 0.542512E 02, 0.543934E 02, 0.545323E 02/
      DATA (ENT(I),I= 11,663,17)/
     * 0.521346E 01, 0.513395E 01, 0.509355E 01, 0.507021E 01,
     * 0.505760E 01, 0.505350E 01, 0.505662E 01, 0.506561E 01,
     * 0.507909E 01, 0.509574E 01, 0.511442E 01, 0.513419E 01,
     * 0.515435E 01, 0.517441E 01, 0.519403E 01, 0.521297E 01,
     * 0.523113E 01, 0.524845E 01, 0.526491E 01, 0.528053E 01,
     * 0.529536E 01, 0.530944E 01, 0.532284E 01, 0.533560E 01,
     * 0.534780E 01, 0.535949E 01, 0.537072E 01, 0.538156E 01,
     * 0.539206E 01, 0.540227E 01, 0.541223E 01, 0.542200E 01,
     * 0.543161E 01, 0.544110E 01, 0.545052E 01, 0.545990E 01,
     * 0.546927E 01, 0.547867E 01, 0.548813E 01/
      DATA EFR( 11) /   0.447564E 03 /
      DATA FORM( 12),(STY(I),I= 12,68,17), WTM( 12) /
     * 4HO+  , 0., 1.,-1., 0.,    15.999499/
      DATA (EK(I),I= 12,663,17)/
```

```
* 0.380548E 02, 0.400693E 02, 0.414985E 02, 0.426071E 02,
* 0.435129E 02, 0.442788E 02, 0.449425E 02, 0.455283E 02,
* 0.460530E 02, 0.465288E 02, 0.469647E 02, 0.473677E 02,
* 0.477431E 02, 0.480952E 02, 0.484274E 02, 0.487425E 02,
* 0.490426E 02, 0.493296E 02, 0.496050E 02, 0.498699E 02,
* 0.501254E 02, 0.503723E 02, 0.506113E 02, 0.508430E 02,
* 0.510679E 02, 0.512864E 02, 0.514989E 02, 0.517057E 02,
* 0.519070E 02, 0.521033E 02, 0.522946E 02, 0.524811E 02,
* 0.526632E 02, 0.528409E 02, 0.530145E 02, 0.531841E 02,
* 0.533497E 02, 0.535117E 02, 0.536701E 02/
 DATA (ENT(I),I= 12,663,17)/
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496817E 01,
* 0.496832E 01, 0.496905E 01, 0.497127E 01, 0.497629E 01,
* 0.498551E 01, 0.500017E 01, 0.502118E 01, 0.504897E 01,
* 0.508357E 01, 0.512464E 01, 0.517154E 01, 0.522344E 01,
* 0.527937E 01, 0.533835E 01, 0.539937E 01, 0.546150E 01,
* 0.552387E 01, 0.558573E 01, 0.564641E 01, 0.570539E 01,
* 0.576222E 01, 0.581656E 01, 0.586818E 01, 0.591690E 01,
* 0.596264E 01, 0.600535E 01, 0.604505E 01, 0.608178E 01,
* 0.611564E 01, 0.614672E 01, 0.617514E 01, 0.620104E 01,
* 0.622457E 01, 0.624587E 01, 0.626508E 01/
 DATA EFR( 12) /   0.372942E 03 /
 DATA FORM( 13),(STY(I),I= 13,68,17), WTM( 13) /
* 4HA+  , 0., 0.,-1., 1.,    39.943390/
 DATA (EK(I),I= 13,663,17)/
* 0.409048E 02, 0.430334E 02, 0.445521E 02, 0.457284E 02,
* 0.466861E 02, 0.474927E 02, 0.481888E 02, 0.488007E 02,
* 0.493464E 02, 0.498387E 02, 0.502871E 02, 0.506986E 02,
* 0.510789E 02, 0.514323E 02, 0.517624E 02, 0.520721E 02,
* 0.523636E 02, 0.526391E 02, 0.529001E 02, 0.531482E 02,
* 0.533845E 02, 0.536101E 02, 0.538259E 02, 0.540328E 02,
* 0.542314E 02, 0.544224E 02, 0.546063E 02, 0.547837E 02,
* 0.549550E 02, 0.551207E 02, 0.552810E 02, 0.554363E 02,
* 0.555870E 02, 0.557333E 02, 0.558754E 02, 0.560137E 02,
* 0.561483E 02, 0.562795E 02, 0.564074E 02/
 DATA (ENT(I),I= 13,663,17)/
* 0.521336E 01, 0.527489E 01, 0.527819E 01, 0.526272E 01,
* 0.524251E 01, 0.522229E 01, 0.520360E 01, 0.518680E 01,
* 0.517186E 01, 0.515860E 01, 0.514682E 01, 0.513631E 01,
* 0.512690E 01, 0.511845E 01, 0.511081E 01, 0.510389E 01,
* 0.509760E 01, 0.509185E 01, 0.508657E 01, 0.508173E 01,
* 0.507726E 01, 0.507313E 01, 0.506932E 01, 0.506579E 01,
* 0.506253E 01, 0.505953E 01, 0.505678E 01, 0.505430E 01,
* 0.505209E 01, 0.505018E 01, 0.504860E 01, 0.504738E 01,
* 0.504658E 01, 0.504626E 01, 0.504649E 01, 0.504734E 01,
* 0.504892E 01, 0.505131E 01, 0.505462E 01/
 DATA EFR( 13) /   0.363345E 03 /
 DATA FORM( 14),(STY(I),I= 14,68,17), WTM( 14) /
* 4HN   , 1., 0., 0., 0.,    14.007999/
 DATA (EK(I),I= 14,663,17)/
* 0.376586E 02, 0.396730E 02, 0.411023E 02, 0.422109E 02,
* 0.431172E 02, 0.438844E 02, 0.445509E 02, 0.451420E 02,
* 0.456748E 02, 0.461618E 02, 0.466120E 02, 0.470321E 02,
* 0.474270E 02, 0.478005E 02, 0.481556E 02, 0.484944E 02,
* 0.488186E 02, 0.491297E 02, 0.494287E 02, 0.497167E 02,
* 0.499943E 02, 0.502623E 02, 0.505213E 02, 0.507719E 02,
* 0.510145E 02, 0.512498E 02, 0.514782E 02, 0.517001E 02,
* 0.519160E 02, 0.521263E 02, 0.523315E 02, 0.525320E 02,
* 0.527282E 02, 0.529205E 02, 0.531092E 02, 0.532948E 02,
* 0.534774E 02, 0.536575E 02, 0.538354E 02/
 DATA (ENT(I),I= 14,663,17)/
* 0.496815E 01, 0.496815E 01, 0.496822E 01, 0.496901E 01,
```

```
* 0.497272E 01, 0.498289E 01, 0.500312E 01, 0.503583E 01,
* 0.508187E 01, 0.514055E 01, 0.521007E 01, 0.528799E 01,
* 0.537159E 01, 0.545821E 01, 0.554546E 01, 0.563132E 01,
* 0.571421E 01, 0.579301E 01, 0.586699E 01, 0.593582E 01,
* 0.599949E 01, 0.605827E 01, 0.611264E 01, 0.616328E 01,
* 0.621101E 01, 0.625673E 01, 0.630140E 01, 0.634603E 01,
* 0.639161E 01, 0.643910E 01, 0.648943E 01, 0.654343E 01,
* 0.660185E 01, 0.666532E 01, 0.673436E 01, 0.680934E 01,
* 0.689050E 01, 0.697793E 01, 0.707159E 01/
 DATA EFR( 14) /   0.112507E 03 /
 DATA FORM( 15),(STY(I),I= 15,68,17), WTM( 15) /
* 4HO   , 0., 1., 0., 0.,     16.000000/
 DATA (EK(I),I= 15,663,17)/
* 0.394596E 02, 0.415394E 02, 0.430027E 02, 0.441323E 02,
* 0.450526E 02, 0.458297E 02, 0.465028E 02, 0.470973E 02,
* 0.476302E 02, 0.481136E 02, 0.485564E 02, 0.489653E 02,
* 0.493453E 02, 0.497004E 02, 0.500338E 02, 0.503482E 02,
* 0.506456E 02, 0.509278E 02, 0.511963E 02, 0.514525E 02,
* 0.516975E 02, 0.519321E 02, 0.521574E 02, 0.523739E 02,
* 0.525826E 02, 0.527839E 02, 0.529784E 02, 0.531667E 02,
* 0.533493E 02, 0.535266E 02, 0.536991E 02, 0.538672E 02,
* 0.540312E 02, 0.541916E 02, 0.543487E 02, 0.545027E 02,
* 0.546541E 02, 0.548031E 02, 0.549500E 02/
 DATA (ENT(I),I= 15,663,17)/
* 0.515925E 01, 0.510249E 01, 0.507170E 01, 0.505329E 01,
* 0.504317E 01, 0.504007E 01, 0.504325E 01, 0.505179E 01,
* 0.506460E 01, 0.508054E 01, 0.509856E 01, 0.511776E 01,
* 0.513740E 01, 0.515695E 01, 0.517600E 01, 0.519429E 01,
* 0.521168E 01, 0.522809E 01, 0.524356E 01, 0.525819E 01,
* 0.527212E 01, 0.528560E 01, 0.529890E 01, 0.531234E 01,
* 0.532633E 01, 0.534126E 01, 0.535761E 01, 0.537585E 01,
* 0.539649E 01, 0.542003E 01, 0.544698E 01, 0.547782E 01,
* 0.551303E 01, 0.555302E 01, 0.559819E 01, 0.564886E 01,
* 0.570530E 01, 0.576769E 01, 0.583615E 01/
 DATA EFR( 15) /   0.589725E 02 /
 DATA FORM( 16),(STY(I),I= 16,68,17), WTM( 16) /
* 4HE-  , 0., 0., 1., 0.,      0.000549/
 DATA (EK(I),I= 16,663,17)/
* 0.603262E 01, 0.804704E 01, 0.947628E 01, 0.105849E 02,
* 0.114907E 02, 0.122565E 02, 0.129199E 02, 0.135051E 02,
* 0.140286E 02, 0.145021E 02, 0.149344E 02, 0.153320E 02,
* 0.157002E 02, 0.160430E 02, 0.163636E 02, 0.166648E 02,
* 0.169488E 02, 0.172174E 02, 0.174722E 02, 0.177146E 02,
* 0.179457E 02, 0.181666E 02, 0.183780E 02, 0.185808E 02,
* 0.187757E 02, 0.189632E 02, 0.191439E 02, 0.193182E 02,
* 0.194866E 02, 0.196495E 02, 0.198073E 02, 0.199601E 02,
* 0.201085E 02, 0.202525E 02, 0.203924E 02, 0.205285E 02,
* 0.206610E 02, 0.207901E 02, 0.209159E 02/
 DATA (ENT(I),I= 16,663,17)/
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01/
 DATA EFR( 16) /   0.0      /
 DATA FORM( 17),(STY(I),I= 17,68,17), WTM( 17) /
* 4HA   , 0., 0., 0., 1.,     39.943985/
```

```
 DATA (EK(I),I= 17,663,17)/
* 0.380272E 02, 0.400416E 02, 0.414709E 02, 0.425795E 02,
* 0.434853E 02, 0.442511E 02, 0.449145E 02, 0.454997E 02,
* 0.460231E 02, 0.464966E 02, 0.469289E 02, 0.473266E 02,
* 0.476948E 02, 0.480375E 02, 0.483582E 02, 0.486594E 02,
* 0.489434E 02, 0.492120E 02, 0.494669E 02, 0.497094E 02,
* 0.499408E 02, 0.501620E 02, 0.503742E 02, 0.505782E 02,
* 0.507750E 02, 0.509653E 02, 0.511502E 02, 0.513306E 02,
* 0.515075E 02, 0.516818E 02, 0.518548E 02, 0.520275E 02,
* 0.522010E 02, 0.523766E 02, 0.525554E 02, 0.527385E 02,
* 0.529267E 02, 0.531209E 02, 0.533217E 02/
 DATA (ENT(I),I= 17,663,17)/
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496815E 01, 0.496815E 01, 0.496815E 01,
* 0.496815E 01, 0.496816E 01, 0.496819E 01, 0.496827E 01,
* 0.496848E 01, 0.496894E 01, 0.496989E 01, 0.497174E 01,
* 0.497512E 01, 0.498096E 01, 0.499056E 01, 0.500568E 01,
* 0.502860E 01, 0.506214E 01, 0.510967E 01, 0.517513E 01,
* 0.526288E 01, 0.537756E 01, 0.552393E 01, 0.570652E 01,
* 0.592933E 01, 0.619541E 01, 0.650645E 01, 0.686245E 01,
* 0.726144E 01, 0.769933E 01, 0.817002E 01/
 DATA EFR( 17) /   0.0          /
 END
```

SECTION IV

SUBROUTINES

This section contains all of the subroutines used in the program. Each subroutine is discussed in a separate paragraph.

1. SUBROUTINE GINPUT

a. General

Subroutine GINPUT is a reasonably flexible general data loading routine. GINPUT is independent of the particular calling program. This subprogram is somewhat similar to other symbolic loader schemes, such as NAMELIST.

b. Prototype Formats

GINPUT has three prototype formats for input data cards, one each for BCD data, integer data, and floating point data.

(1) Prototype BCD Format

Each BCD data card has the following format:

Columns	Information
1-6	BCD variable symbol, left justified with no imbedded blanks.
7-10	Index for the storage array, integer variable right justified.
11-14	Card sequence number, integer variable right justified. (Unused by this version of the subroutine.)
15-20	Data word count, integer variable right justified. Determines the number of data words on this card to be stored (≤ 5).
21-70	Five data fields, each 10 columns wide. Format is 5A10.
71-72	Unused.
73-80	Subroutine name in BCD, left justified with no imbedded blanks.

(2) Prototype Integer Format

All fields are the same as for the prototype BCD format except columns 21-70, which has five fields, 10 columns wide, in I10 format.

(3) Prototype Floating Point Format

Same as prototype BCD format, except for columns 21-70, which has five fields, 10 columns wide, in E10.0 format.

c. Prefix Cards

The data deck for each prototype format must have a prefix card. The prefix cards have the following format:

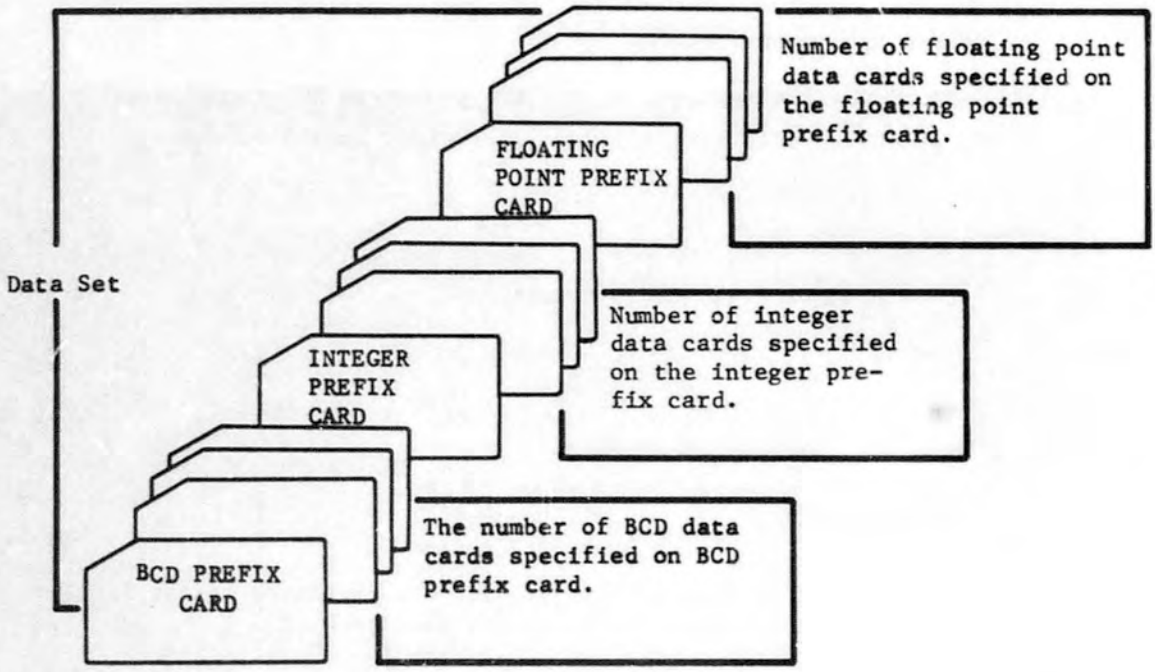
Columns	Information
1-6	Identifies the prototype format for the following data pack.
7-10	Unused.
11-14	Sequence number, integer right justified. (unprocessed by this version of GINPUT.)
15-26	Unused.
27-30	The number of data cards that follow for this prototype format. (Can be zero.)
31-72	Unused.
73-80	Subroutine name in BCD, left justified with no imbedded blanks.

The following table gives the setup for columns 1-6 of the prefix cards:

Column 1	Data type
NBCD	BCD data card(s) follow.
NFIXED	Integer data card(s) follow.
NFLOAT	Floating point data card(s) follow.

d. General Data Deck Setup

The data deck must be set up in the following order. (The exceptions are given later in this section.)



The data set illustrated must be input in the order

BCD Data
Integer Data
Floating Point Data

with the following exceptions:

Prototype data decks can be deleted from the front of data set.

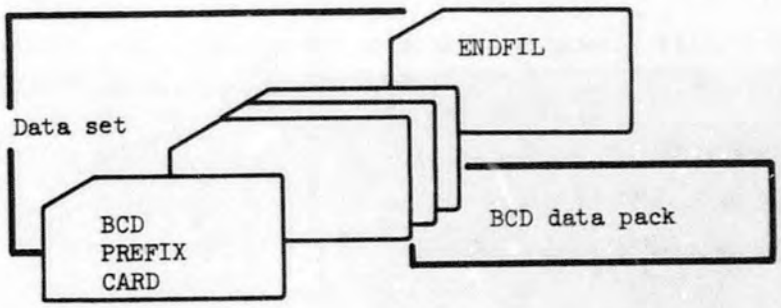
Hence, the following options are possible:

[BCD Data]	deleted
Integer Data	
Floating Point Data	

and

[BCD Data]	deleted
[Integer Data]	deleted
Floating Point Data	

A card with ENDFIL in columns 1-6 is used to prototype data packs from the back of a data set. Hence, if it is desired to load only BCD data, the deck setup would appear as



e. Storage of Data

The data items are stored in either ARRAY or IARRAY. (ARRAY and IARRAY should refer to the same storage location--in the calling program this would appear as CALL GINPUT (A, A, . . .). This setup is used to prevent the FORTRAN compiler from converting integer information to floating point structure.)

The first data word on a card is stored in ARRAY(I), where I is the number in columns 7-10 of the data card.

A data card contains 1-5 words of information. The number of words transmitted from the data card to ARRAY is obtained from card columns 15-20.

f. User Linkage to ARRAY

The calling program normally would link other program variable names to ARRAY with EQUIVALENCE statements. Hence, if X is a variable name in the calling program, it could be linked to the data card by

```
DIMENSION ARRAY (100) EQUIVALENCE (ARRAY (50), X)

CALL GINPUT (ARRAY, ARRAY, . . . .)
```

and punching 50 in card columns 9-10 on the data card.

g. Structure of Array BCD

The array named BCD in the subroutine calling sequence must be set up in the calling program. This array is set up as follows:

(1) BCD(1)

BCD(1) contains a subroutine name in BCD. The bit configuration is checked against card columns 73-80. This is used to help isolate errors in the input data. (See the section on error checking for more details.)

(2) BCD(2)

This location contains a floating point number that gives the number of different variable names that can validly be stored in ARRAY.

(3) Variable Names

It is desirable to use variable names on input data cards similar to--if not the same as--the symbols internal to the computer program. These names are entered into BCD(3) - BCD (BCD(2) + 2) by the calling program. (Normally a DATA statement would be used with left justified BCD conversion.) The characters in columns 1-6 of the data cards are checked against these names. The linkage between the data cards and the calling program is

(a) The ARRAY indices

(b) The EQUIVALENCE statements in the calling program

h. Input-Output Unit Assignments

Input-output unit assignments are symbolic and made in the calling sequence of GINPUT.

The logical input device is designated as INUNT; the logical output device as IOUTUT.

These values must be supplied by the calling program.

i. Data Card Error Checking

Subroutine GINPUT does a substantial amount of error checking of the input data. If errors are detected, messages are printed, and in one situation a salvage effort is made. The following error checks are performed.

(1) Subroutine Name

The characters in card columns 73-80 are checked against BCD(1).

(2) Program Variable Name

The symbol in card columns 1-6 is checked against the entries in the BCD array.

(3) The word count per card is checked against the upper limit (5). (A salvage operation is attempted by loading 5 data items.)

(4) The absence of prefix cards at the necessary places in the data deck is sensed. (This usually would result from a discrepancy between the card count and the actual number of cards present.)

GINPUT lists an error number for each prototype error that is sensed. The following list gives the error numbers and their probable cause.

Error number	Probable cause
1	Prefix card does not have the correct format code in columns 1-6. (Applies to all three prefix cards.) Too few or too many data cards.
2	Incorrect subroutine name on BCD prefix card.
4	Incorrect subroutine name on a BCD data card.
6	Variable name on BCD data card not in the name table.
7	Word count error on BCD data card.
8	Prefix card does not have correct format code. (Integer and floating point modes.) Too few or too many data cards.
9	Incorrect subroutine name on an integer prefix card.
11	Incorrect subroutine name on an integer data card.
13	Integer variable name on a data card could not be matched with name table.
14	Word count error on an integer data card.
15	Prototype format error for a floating point prefix card. Card order error probable.
16	Incorrect subroutine name on a floating point prefix card.
18	Subroutine name on a floating point data card could not be matched.
20	Variable name on a floating point data card could not be matched.
21	Word count error on a floating point data card.

j. Partial Data Loading

One of the advantages of an input routine of this type is that all decisions for loading part of the data are made by subroutine GINPUT and the user. Normally, the user would make prototype data sheets for all information that would not change, variable symbol, etc., and insert only those items that change. This flexibility is illustrated in paragraph k below.

k. Sample Program Setup

This section gives a sample program setup using GINPUT. The variables and the number of locations reserved for each variable are given in the table below.

Variable	Mode	Storage
TITLE 1	BCD	1
TITLE 2	BCD	10
I1	INTEGER	1
I2	INTEGER	3
X	FLOATING POINT	1
Y	FLOATING POINT	6

Assume the subroutine name is MAIN, then a calling program to just load the data could be the one shown below.

```
      DIMENSION ARRAY(22), BCD(8), TITLE2(10), 1 I2(3), Y(6)

      EQUIVALENCE   (ARRAY(1), TITLE1),
     1              (ARRAY(2), TITLE2(1)),
     2              (ARRAY(12), I1),
     3              (ARRAY(13), I2(1)),
     4              (ARRAY(16), X),
     5              (ARRAY(17), Y(1))

      DATA BCD/4HMAIN, 6., 6HTITLE1, 6HTITLE2, 2HI1, 2HI2, 1HX, 1HY/

      CALL GINPUT (ARRAY, ARRAY, BCD, 5, 6, IERROR)
      STOP
      END
```

The above program would load data into ARRAY. The symbols TITLE1, TITLE2, I1, I2, X, and Y are available in the calling program to reference the various data items. Table IV illustrates sample data cards that will load every location of ARRAY.

Table IV

EXAMPLE OF COMPLETE SET OF DATA CARDS

```
NBCD                                  3                                                                  MAIN
TITLE1     1          1BCD WORD 1                                                                        MAIN
TITLE2     2          5BCD WORD 1BCD WORD 2BCD WORD 3BCD WORD 4BCD WORD 5                                MAIN
TITLE2     7          5BCD WORD 1BCD WORD 2BCD WORD 3BCD WORD 4BCD WORD 5                                MAIN
NFIXED                                2                                                                  MAIN
I1        12          1 I WORD 1                                                                         MAIN
I2        13          3 I WORD 1  I WORD 2  I WORD 3                                                     MAIN
NFLOAT                                3                                                                  MAIN
X         16          1 FP WORD 1                                                                        MAIN
Y         17          5 FP WORD 1 FP WORD 2 FP WORD 3 FP WORD 4 FP WORD 5                                MAIN
Y         22          1 FP WORD 1                                                                        MAIN
ENDFIL                                                                                                   MAIN
```

Table Va illustrates sample data for loading of integer and floating point data, with BCD information deleted.

Table Vb illustrates sample data to load information into TITLE1, I2(2), Y(2), and Y(5).

NOTE: In tables IV and V the words are numbered relative to each data card.

1. Symbols

Symbolic representation	Meaning or use
ARRAY IARRAY	The array where the data items are stored.
BCD(1)	A subroutine name, in BCD, for a given data set.
BCD(2)	The number of variable names for the data set. Floating point number.
BCD(I) (I - 3, BCD(2))	The variable names in left justified BCD mode, one name per location with no imbedded blanks.
INUNT	Logical input unit number. (5 for many uses.)
IOUTUT	Logical output unit. Only used if GINPUT detects data errors. (6 for many uses.)
IERT	Flag that signals if GINPUT detected data errors. 0 if no errors were detected, 1 if errors were detected.
BUFF	Input area for one floating point or BCD data card.
IBUFF	Input area for one integer data card.
F1	Characters NBCD; used to identify a BCD prefix card.
F2	Characters NFIXED; used to identify an integer prefix card.
F3	Characters NFLOAT; used to identify a floating point prefix card.
F4	Characters ENDFIL; used for "end-of-file" card.
IARG	Upper index value used when searching BCD array for variable names.

Table V

EXAMPLE OF PARTIAL SET OF DATA CARDS

A. BCD INFORMATION DELETED

```
NBCD                                0                                                  MAIN
NFIXED                              2                                                  MAIN
I1        12          1 I WORD 1                                                       MAIN
I2        13          3 I WORD 1  I WORD 2  I WORD 3                                   MAIN
NFLOAT                              3                                                  MAIN
X         16          1 FP WORD 1                                                      MAIN
Y         17          5 FP WORD 1 FP WORD 2 FP WORD 3 FP WORD 4 FP WORD 5              MAIN
Y         22          1 FP WORD 1                                                      MAIN
ENDFIL                                                                               MAIN
```

B. SELECTED INFORMATION DELETED --

```
NBCD                                1                                                  MAIN
TITLE1     1          1 BCD WORD1                                                      MAIN
NFIXED                              1                                                  MAIN
I2        14          1 I WORD 1                                                       MAIN
NFLOAT                              2                                                  MAIN
Y         18          1  Y(2)                                                         MAIN
Y         21          1  Y(5)                                                         MAIN
ENDFIL                                                                               MAIN
```

IERR	Internal error signal.
NOCTN	Maximum word count per data card, if GINPUT detects a word count error.
IGEN	Initializes error number for prototype data cards.

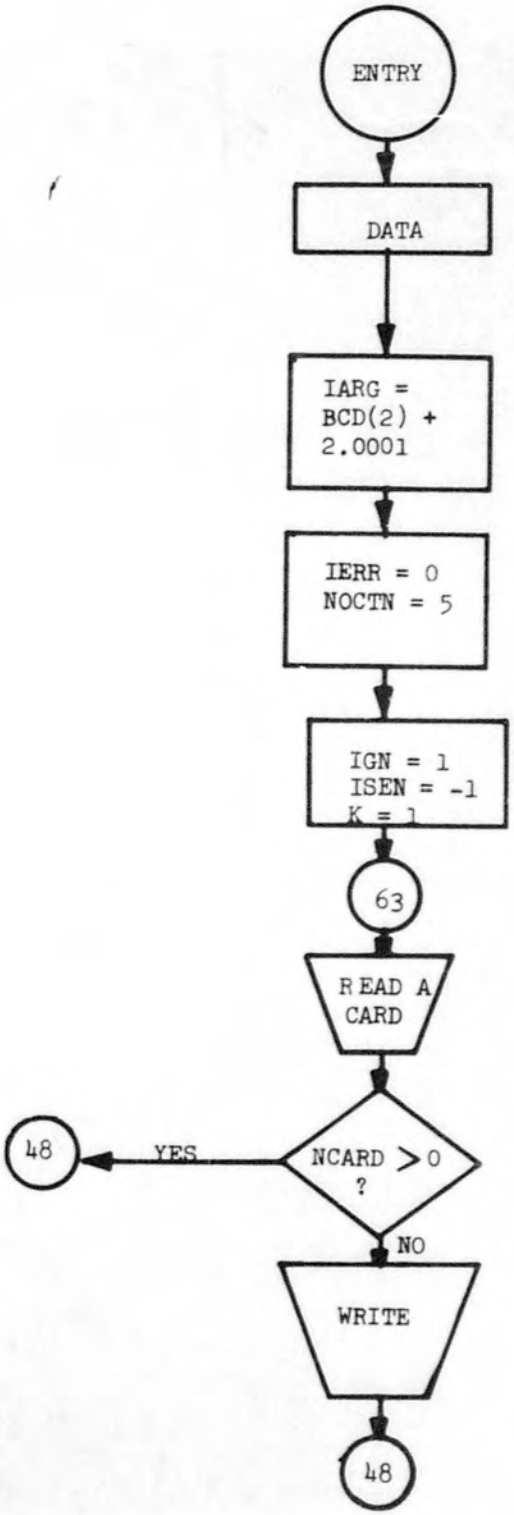
Prototype data	Value
BCD	1
Integer	8
Floating Point	15

ISEN	Used for sequence checking. (Sequence checking is performed by the GINPUT version described in this report.)
K	Integer variable that indicates which type of data is being loaded.

Prototype data	K
BCD	1
Integer	2
Floating Point	3

TYPE	Either F1, F2, F3, F4, or a variable name.
ISQ ISEQ	Card sequence number. (Unchecked by this version of the program.)
NCARD	The number of cards for a given prototype format--excluding prefix card(s).
SUB	BCD subroutine name for this data set, in columns 73-80 of each data card and the prefix cards.
IGO	Error type number. (Values from 1 to 21; explained in detail in the error checking section.)
INDX	The index of ARRAY or IARRAY where the first word on a data card is stored.
IWCNT	The number of words on a data card that are stored in ARRAY or IARRAY.
J1	Pick up index for BUFF or IBUFF.

m. Flow Chart



DEFINE INTERNAL DATA VARIABLES.

SET THE UPPER LIMIT FOR VARIABLE NAME SCAN.

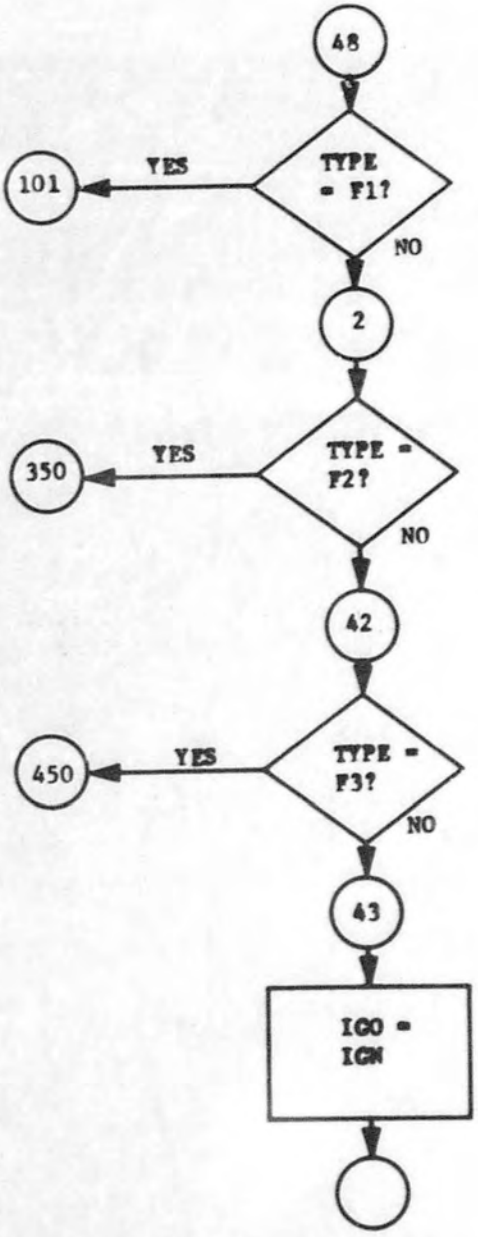
INITIALIZE ERROR INDICATOR AND MAXIMUM WORD COUNT PER CARD.

INITIALIZE ERROR FLAG, SEQUENCE NUMBER, AND CARD PROTOTYPE FLAG.

READ THE FIRST PREFIX CARD.

IS THE NUMBER OF DATA CARDS ZERO?

LIST A CARD COUNT ERROR.



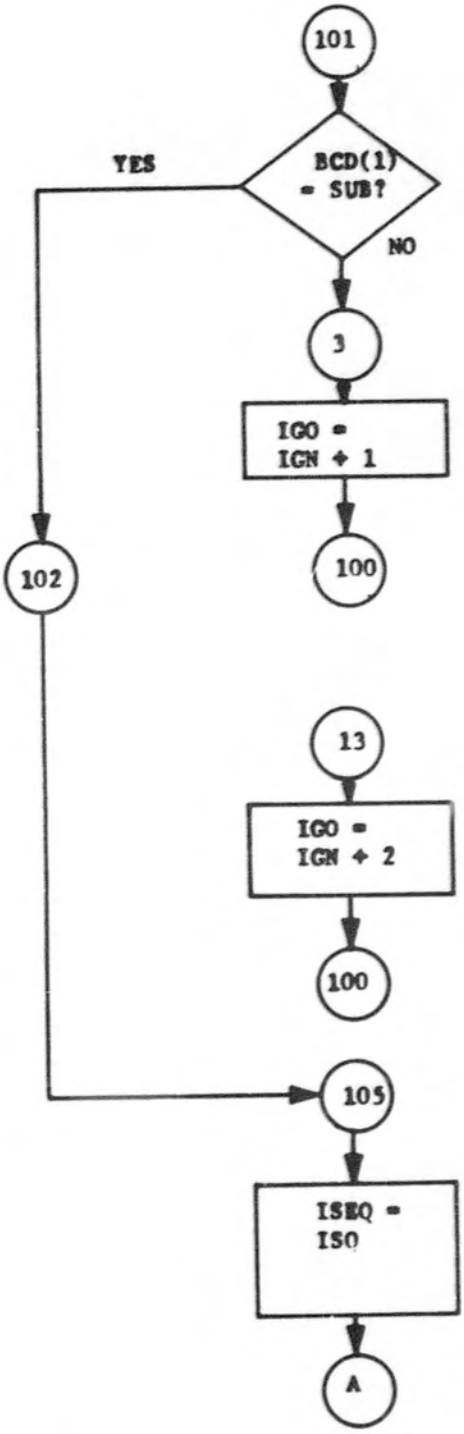
DOES THE PREFIX CARD INDICATE BCD DATA FOLLOWS?

DOES THE PREFIX CARD INDICATE INTEGER DATA FOLLOWS?

DOES THE PREFIX CARD INDICATE FLOATING POINT DATA FOLLOWS?

SET ERROR FLAG NUMBER FOR A PREFIX CARD THAT DOES NOT SPECIFY ONE OF THE THREE POSSIBLE INPUT FORMATS CORRECTLY.

FLAG VALUE	
BCD DATA	1
INTEGER DATA	8
FLOATING POINT DATA	15

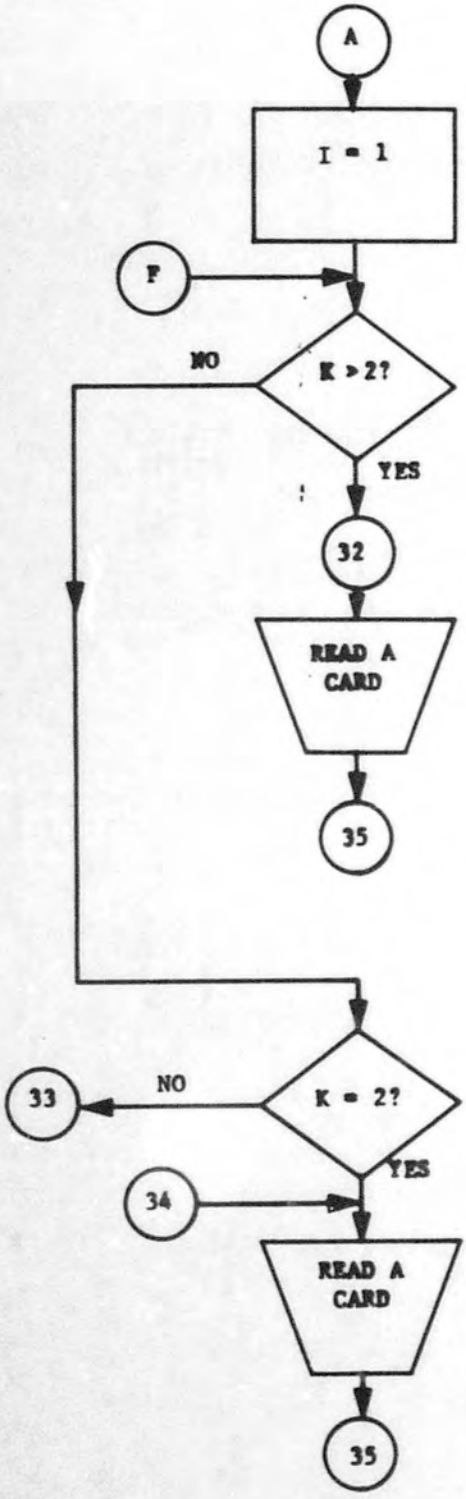


CHECK SUBROUTINE NAME.

SET ERROR FLAG FOR INCORRECT SUBROUTINE NAME ON A PREFIX CARD. FLAG VALUE

BCD	2
INTEGER DATA	
FLOATING POINT DATA	16

PICK UP THE CARD SEQUENCE NUMBER



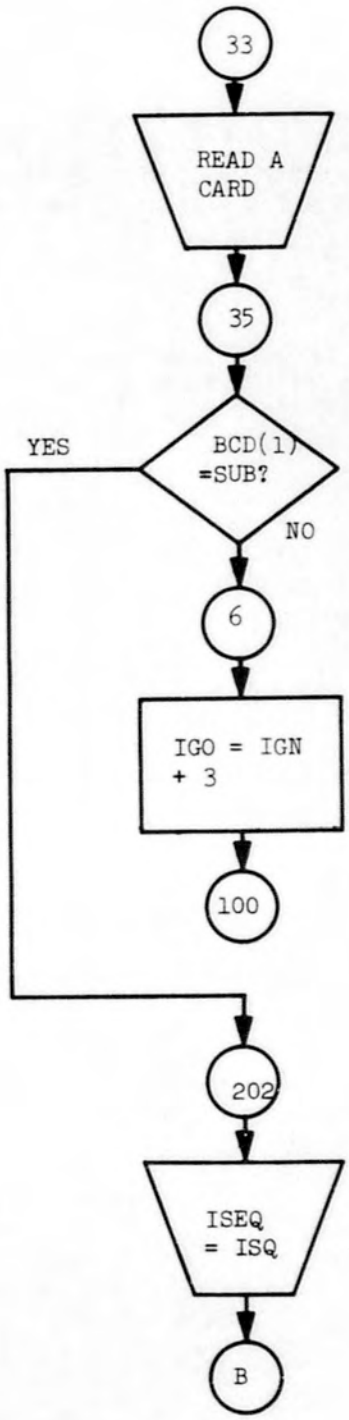
INITIALIZE THE LOOP INDEX TO READ NCARD CARDS OF PROTOTYPE TYPE.

IS THIS DATA PACK FLOATING POINT DATA?

READ ONE FLOATING POINT DATA CARD.

IS THE DATA PACK INTEGER DATA?

READ ONE INTEGER DATA CARD.



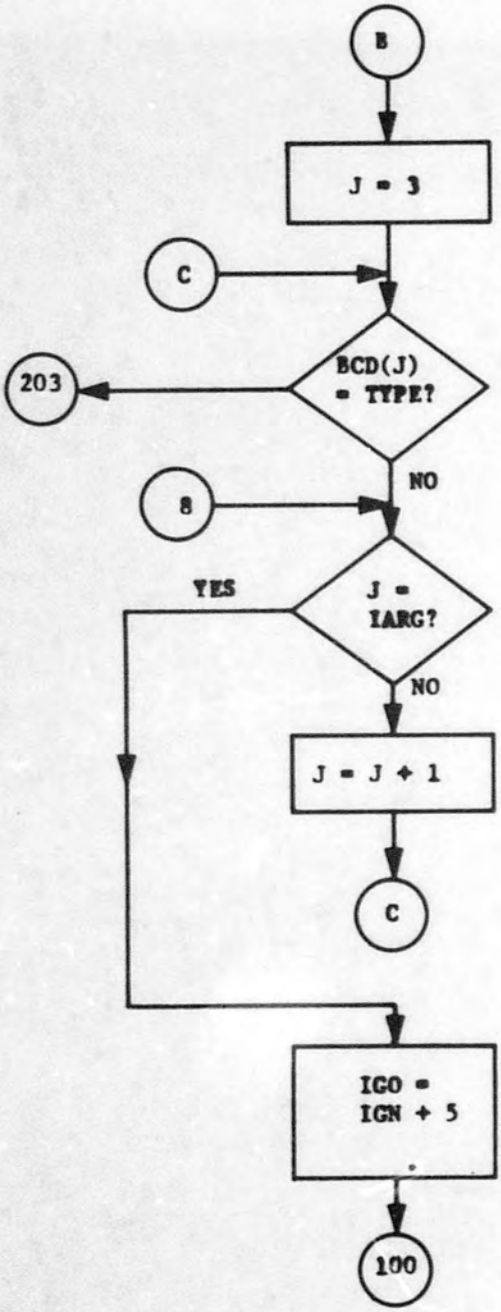
READ ONE BCD DATA CARD.

IS THE SUBROUTINE NAME ON THE DATA CARD CORRECT?

SET THE ERROR FLAG FOR AN INCORRECT SUBROUTINE NAME ON A DATA CARD. FLAG VALUE

BCD DATA	4
INTEGER DATA	11
FLOATING POINT DATA	18

PICK UP THE SEQUENCE NUMBER FROM THE DATA CARD.



INITIALIZE THE LOOP INDEX USED TO CHECK THE VARIABLE NAME

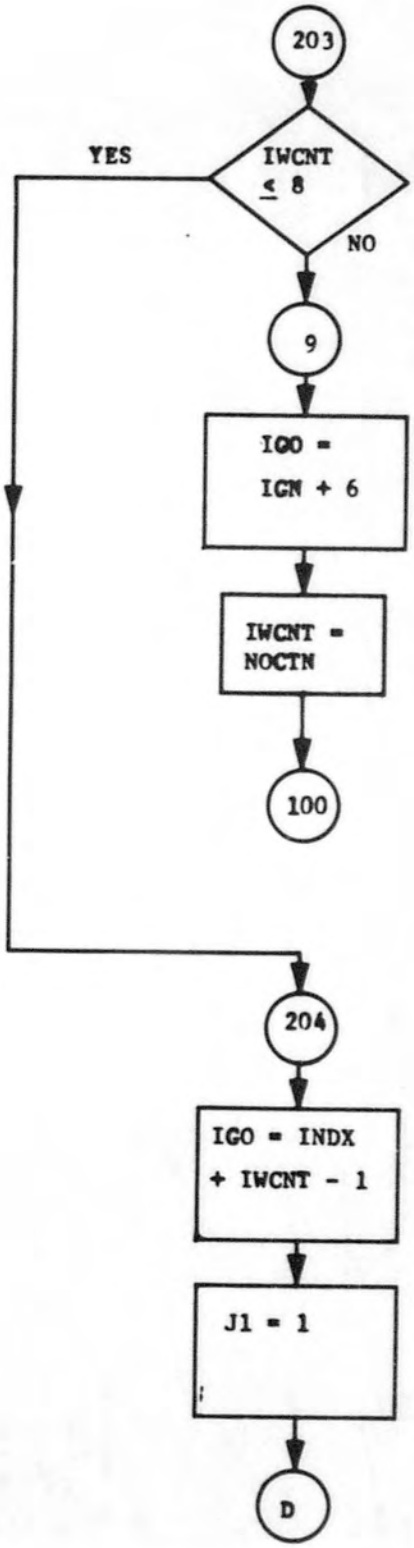
DOES THE DATA CARD VARIABLE NAME MATCH THE JTH ENTRY IN THE BCD ARRAY?

HAVE ALL POSSIBLE VARIABLE NAMES BEEN CHECKED.

INCREMENT THE LOOP SEARCH INDEX.

VARIABLE NAME COULD NOT BE MATCHED. SET ERROR FLAG. FLAG VALUE

BCD	6
INTEGER DATA	13
FLOATING POINT DATA	20



IS THE WORD COUNT GREATER THAN 8?

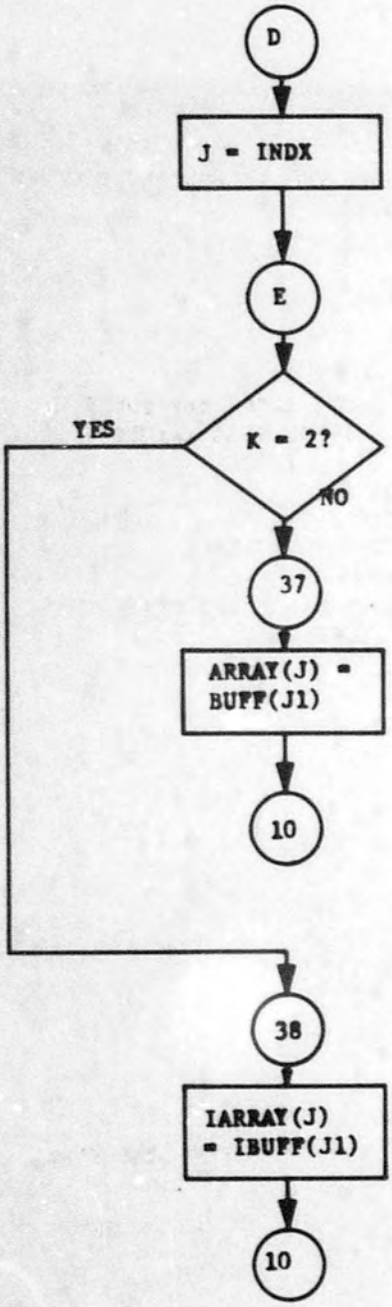
WORD COUNT ERROR SET ERROR FLAG. ERROR FLAG VALUES

BCD DATA	7
INTEGER DATA	14
FLOATING POINT DATA	21

TRY TO SALVAGE AND RESET THE WORD COUNT.

CALCULATE THE UPPER LOAD INDEX FOR STORAGE.

INITIALIZE THE BUFFER PICKUP INDEX.

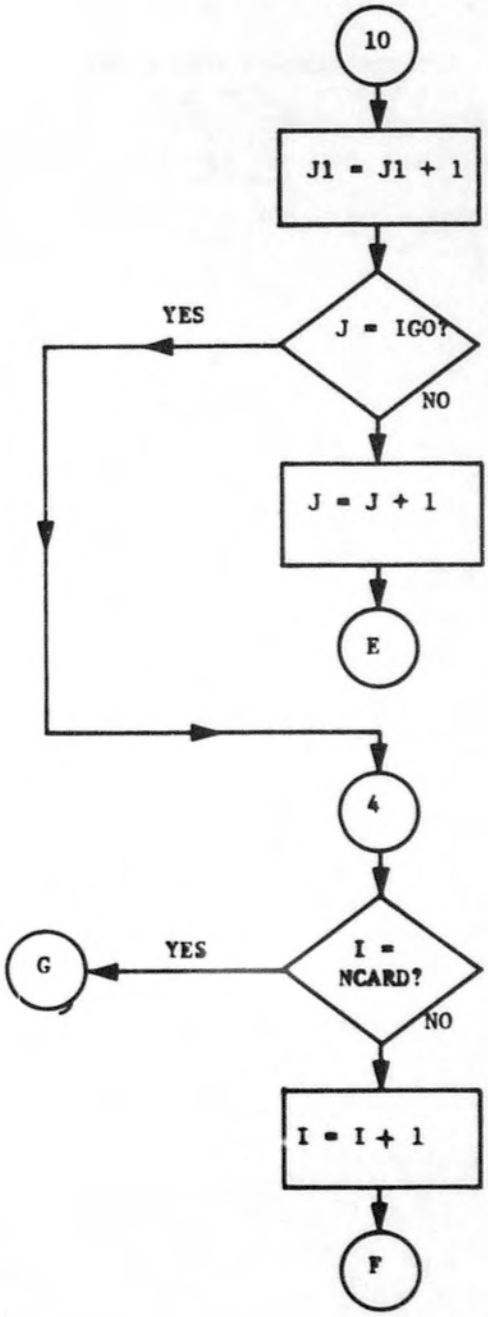


INITIALIZE THE STORE LOOP INDEX WITH THE ARRAY SUBSCRIPT FROM THE DATA CARD.

INTEGER DATA?

PICK UP FLOATING POINT OF BCD DATA ITEMS FROM THE BUFFER AND STORE THEM IN THE DATA ARRAY.

PICK UP INTEGER DATA ITEMS FROM THE BUFFER AND STORE THEM IN THE DATA ARRAY.



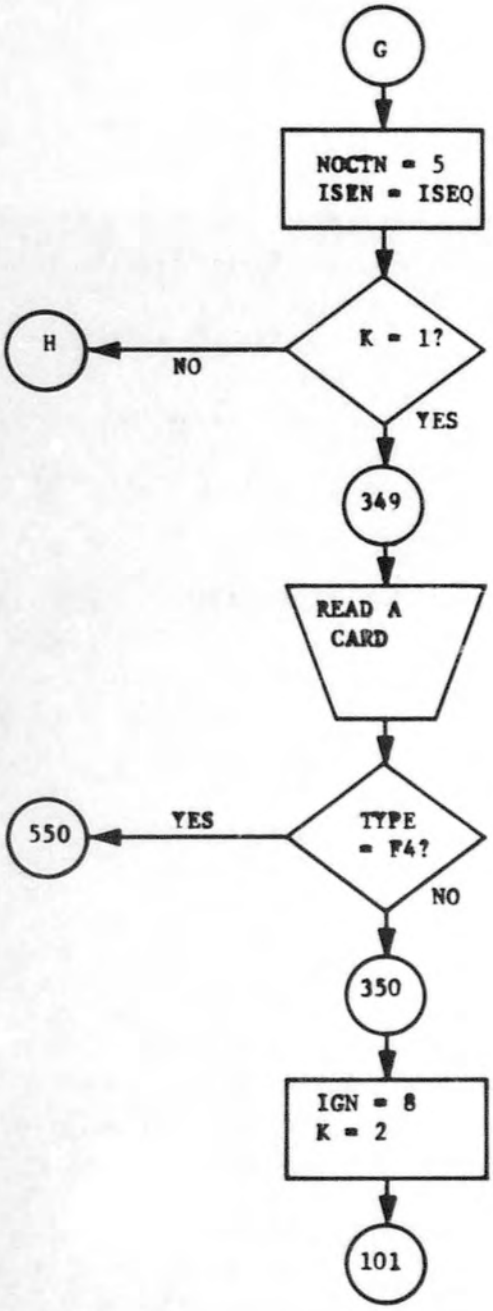
INCREMENT THE BUFFER PICK UP INDEX.

LAST DATA ITEM STORED?

INCREMENT STORE INDEX.

LAST CARD FOR THIS PACK PROCESSED?

INCREMENT LOOP INDEX.



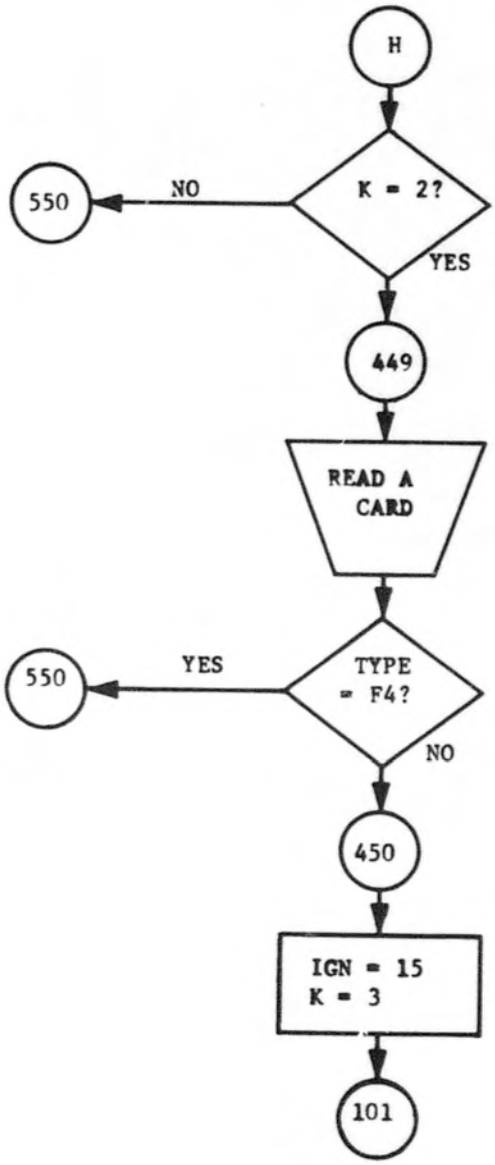
SET THE MAXIMUM WORD COUNT AND SEQUENCE NUMBER.

WAS THE LAST DATA PACK BCD?

READ INTEGER PREFIX CARD.

"END OF FILE" CARD?

SET TRIGGERS FOR INTEGER DATA.

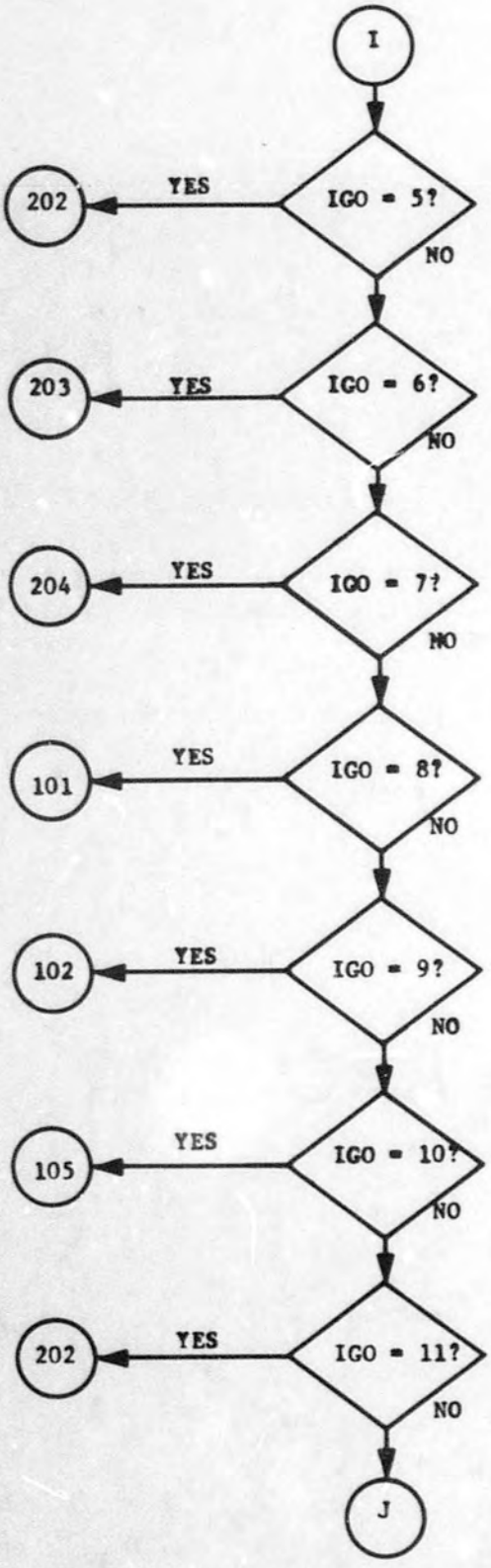


WAS THE LAST DATA PACK INTEGER DATA?

READ FLOATING POINT PREFIX CARD.

"END OF FILE" CARD?

SET TRIGGERS FOR FLOATING POINT DATA.



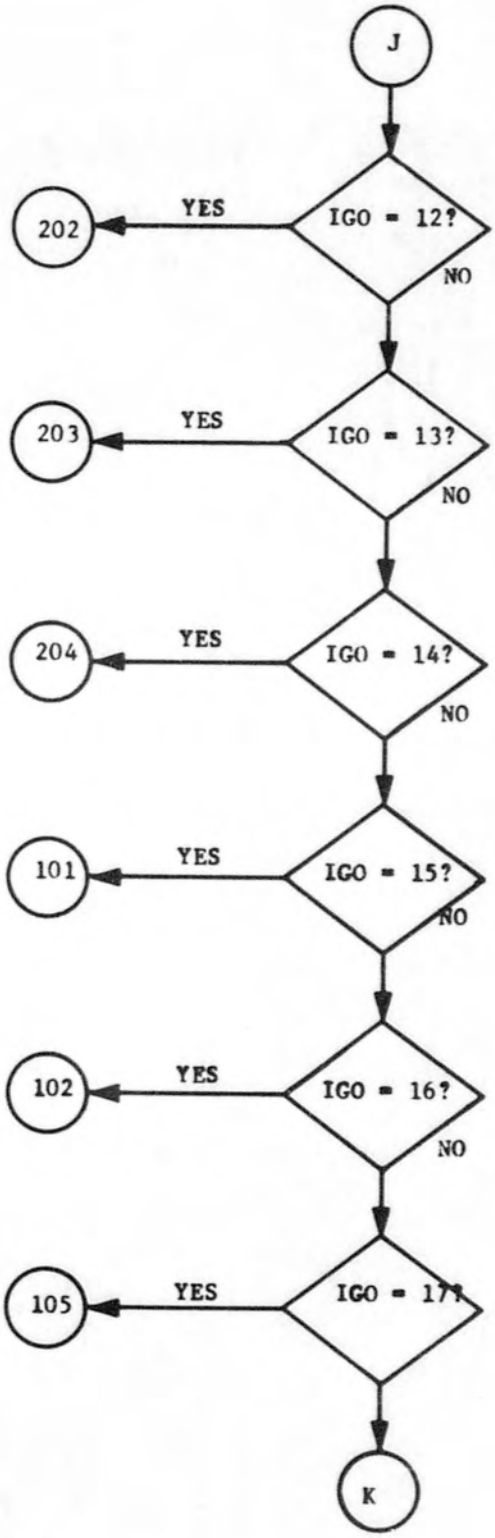
INCORRECT BCD VARIABLE NAME?

WORD COUNT ERROR FOR BCD DATA?

INTEGER CARD COUNT ERROR?

INCORRECT SUBROUTINE NAME ON INTEGER PREFIX CARD?

INCORRECT SUBROUTINE NAME ON INTEGER DATA CARD?

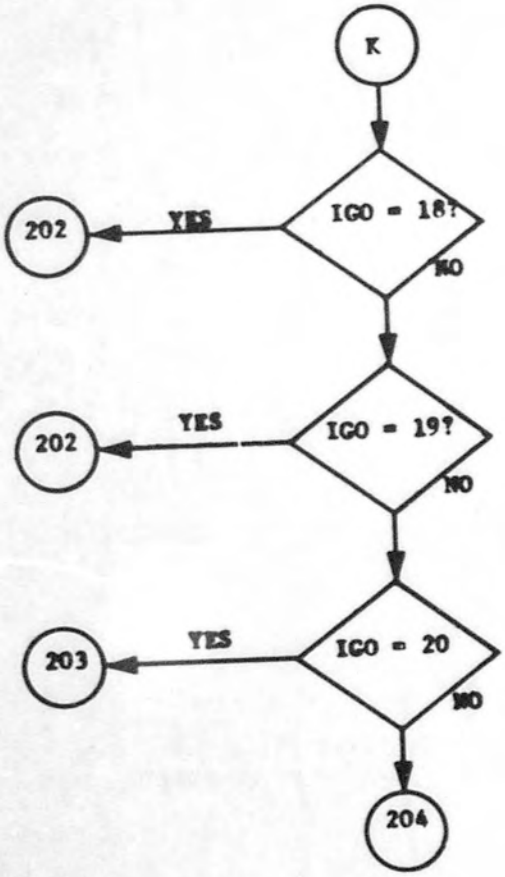


INCORRECT INTEGER VARIABLE NAME?

WORD COUNT ERROR FOR INTEGER DATA?

FLOATING POINT CARD COUNT ERROR?

INCORRECT SUBROUTINE NAME ON FLOATING POINT PREFIX CARD?



INCORRECT SUBROUTINE NAME ON FLOATING POINT DATA CARD?

INCORRECT FLOATING POINT VARIABLE NAME?

WORD COUNT ERROR FOR FLOATING POINT DATA?

```
      N.  LISTING
      SUBROUTINE GINPUT(ARRAY,IARRAY,BCD,INUNT,IOUTUT,IERT)
C****     BCD(1) = NAME OF SUBROUTINE
C****     BCD(2) = NUMBER OF DIFFERENT VARIABLE NAMES
C****     IERT   = ERROR OUTPUT SIGNAL
      DIMENSION  ARRAY(1),IARRAY(1),BCD(1),BUFF(10),IBUFF(10)
      DATA F1,F2,F3,F4/6HNBCD   ,6HNFIXED,6HNFLOAT,6HENDFIL/
      IARG = BCD(2) +  2.0001
      IERR = 0
      NOCTN = 5
C
C**** SETTING UP TO READ BCD DATA
C
      IGN = 1
      ISEN = -1
      K= 1
C
C****     INITIALIZE AND READ PREFIX CARD.
C
   63 READ(INUNT,1) TYPE,ISQ,NCARD,SUB
      IF (EOF,5) 660,165
    1 FORMAT (A6,4XI4,12XI4,42XA6)
C
C****          NOW ERROR CHECK THIS CARD.
C
  165 IF (NCARD) 46,46,48
   46 GO TO (349,449,550),K
   48 IF (K.GT.1) GO TO 101
      IF (TYPE - F1) 2,101,2
    2 IF (TYPE - F2)42,350,42
   42 IF (TYPE - F3)43,450,43
   43 IGO = IGN
      GO TO 100
101   IF(BCD(1)-SUB) 3,102,3
    3 IGO = IGN + 1
      GO TO 100
  102 GO TO 105
   13 IGO = IGN + 2
      GO TO 100
  105 ISEQ = ISQ
C
C****     PREFIX CARD IS FINISHED.NOW READ TITLE CARDS.
C
      DO 4 I = 1,NCARD
C
C****     READ A TITLE CARD INTO A BUFFER AREA.
C
      IF(K -2) 33,34,32
   32 READ(INUNT,25) TYPE, INDX,ISQ,IWCNT,(BUFF(J),J=1,5),SUB
   25 FORMAT(A6,I4,I4,I6,5E10.0,2X,A6)
      GO TO 35
   34 READ(INUNT,15) TYPE,INDX,ISQ,IWCNT,(IBUFF(J),J=1,5),SUB
   15 FORMAT(A6,I4,I4,I6,5I10,2X,A6)
      GO TO 35
   33 READ(INUNT,5)  TYPE,INDX,ISQ,IWCNT,(BUFF(J),J=1,NOCTN),SUB
    5 FORMAT (A6,I4,I4,I6,5A10,2X,A6)
C
C****          NOW ERROR CHECK THE DATA CARD.
```

```
C
   35 IF (BCD(1) - SUB) 6,202,6
    6 IGO  = IGN + 3
      GO TO 100
202   ISEQ = ISQ
      DO 8  J=3,IARG
      IF(BCD(J)- TYPE) 8,203,8
8     CONTINUE
      IGO =  IGN + 5
      GO TO 100
203   IF(IWCNT-8) 204,204,9
    9 IGO = IGN + 6
      IWCNT = NOCTN
      GO TO 100
C
C***            ERROR CHECKING DONE NOW STORE THE DATA.
  204 IGO  = INDX + IWCNT - 1
      J1= 1
      DO 10 J=INDX,IGO
      IF(K - 2) 37,38,37
   37 ARRAY(J) = BUFF(J1)
      GO TO 10
   38 IARRAY(J)= IBUFF(J1)
   10 J1 = J1 + 1
C
C****           NOW DETERMINE IF THERE ARE MORE CARDS TO BE READ.
C
4     CONTINUE
      NOCTN = 5
      ISEN = ISEQ
      GO  TO  (349,449,550),K
C
C**** SETTING UP TO READ INTEGER DATA
C
  349 READ(INUNT,1) TYPE,ISQ,NCARD,SUB
      IF(TYPE - F4) 350,550,350
  350 IGN = 8
      K=2
      GO TO 165
C
C**** SETTING UP TO READ FLOATING POINT DATA
C
  449 READ(INUNT,1) TYPE,ISQ,NCARD,SUB
      IF(TYPE - F4) 450,550,450
  450 IGN =15
      K= 3
      GO TO 165
  550 IERT  = IERR
      RETURN
  100 IERR = 1
      WRITE(IOUTUT,23) IGO
   23 FORMAT(1H0,25X,17HERROR MESSAGE NO.,I3)
      WRITE(IOUTUT,600) TYPE,SUB
  600 FORMAT(1H0,5HTYPE=,A6,9HCASE NAME,A6)
      GO  TO (101,102,105,202,202,203,204,101,102,105,202,202,203,204,
     *101,102,105,202,202,203,204),IGO
  660 CALL EXIT
      END
```

2. SUBROUTINE VALUES

a. General

Subroutine VALUES is used to calculate the functions

$$F = \frac{Y_1 - Y_0}{X_1 - X_0}$$

and

$$E = Y_1 - \frac{(Y_1 - Y_0)}{(X_1 - X_0)} X_1$$

The functions E and F are used in the boundary layer program as part of a convergence routine. The following list defines the parameters in the subroutine calling statement.

Variable	Program symbol	Definition
YO	YO	First Y value
Y1	Y1	Second Y value
XO	XO	First X value
X1	X1	Second X value
F	F	The function F
E	E	The function E

b. Listing

```
SUBROUTINE VALUES(YO,Y1,XO,X1,E,F)
F=(Y1-YO)/(X1-XO)
E=Y1-F*X1
RETURN
END
```

3. SUBROUTINE DISCOT

a. General

Subroutine DISCOT is used to linearly interpolate a function of one or two independent variables.

b. Features of Subroutine DISCOT

(1) One or two independent variables.

(2) Linear interpolation for one independent variable.

(3) Partial second order correction terms for two independent variables.

(4) Does not depend on program DIMENSION statements

(5) X and Y arrays must be in ascending order.

(6) Extrapolation is always performed if required.

(7) No two X or Y table values can be the same.

c. Theory

For a function that depends on one independent variable, the function can be approximated locally by

$$Z \approx Z\left(X_K\right) + \left.\frac{dZ}{dX}\right|_{X=X_K} \left(X_a - X_K\right) \tag{29}$$

where

$$X_{K-1} \leq X_a \leq X_K \tag{30}$$

and

$$\left.\frac{dZ}{dX}\right|_{X=X_K} \simeq \frac{Z_K - Z_{K-1}}{X_K - X_{K-1}} \tag{31}$$

Note that equations (29), (30), and (31) define a "backward" expansion, that is, the expansion is from a larger X value to a smaller X value. The one exception to the above "backward" expansion is extrapolation beyond the maximum X value where a "forward" expansion is done.

For a variable that is a function of two independent variables, four values will be found by the search part of the program. The four X and Y coordinate pairs locate a rectangle in the XY plane such that X_a and Y_a lie in or on the boundary of this rectangle--if extrapolation is not required. If extrapolation is required, the nearest known rectangle is selected.

The fact that four points are located on the surface presents an ambiguity since four points on the surface are known, and it takes only three points to determine the linear terms in a Taylor's expansion of two independent variables. Hence, a valid question is: which three surface values should be used? Rather than answer this question, an alternative approach is used. The alternative is to use all four surface points. By examining Taylor's expansion, it is noticed that it is possible to find the value of one, and only one, other term which is the first cross term. Therefore, the surface is represented locally by

$$Z = Z\left(X_K, Y_K\right) + \left.\frac{dZ}{dX}\right|_{\substack{X=Y_K \\ Y=Y_K}} \left(X_a - X_K\right) + \left.\frac{dZ}{dY}\right|_{\substack{Y=Y_K \\ X=X_K}} \left(Y_a - Y_K\right)$$

$$+ \left.\frac{d^2Z}{dYdX}\right|_{\substack{X=X_K \\ Y=Y_K}} \left(X_a - X_K\right)\left(Y_a - Y_K\right) \qquad (32)$$

When extrapolation is not required, the following inequalities hold.

$$X_{K-1} \leq X_a \leq X_K \qquad (33)$$

$$Y_{L-1} \leq Y_a \leq Y_L \qquad (34)$$

A backward expansion is done using relation (32). The following formulas are used to estimate the partial derivatives.

$$\frac{dZ}{dX} \approx \frac{Z_{K,L} - Z_{K-1,L}}{X_K - X_{K-1}} \tag{35}$$

$$\left(\frac{dZ}{dX}\right)^1 \approx \frac{Z_{K,L-1} - Z_{K-1,L-1}}{X_K - X_{K-1}} \tag{36}$$

$$\frac{d^2Z}{dYdX} \approx \frac{\frac{dZ}{dX} - \left(\frac{dZ}{dX}\right)^1}{Y_L - Y_{L-1}} \tag{37}$$

and

$$\frac{dZ}{dY} \approx \frac{Z_{K,L} - Z_{K,L-1}}{Y_L - Y_{L-1}} \tag{38}$$

One of the undesirable features of equation (32) is that it keeps only part of the second order correction terms. In defense of equation (32) the following statements can be made.

(a) Some use might as well be made of the fourth point since the trouble was taken to find this point.

(b) The requirements on the user have been reduced, since he can determine point spacing by looking at XZ and YZ projections of the surface.

(c) Only linear approximations were promised and the cross term is a bonus.

An interesting geometric illustration of the two independent variable expansion is shown in figure 5. In figure 5 the four surface coordinates are

$$\left(X_2, Y_1, Z_1\right)$$
$$\left(X_2, Y_2, Z_2\right)$$

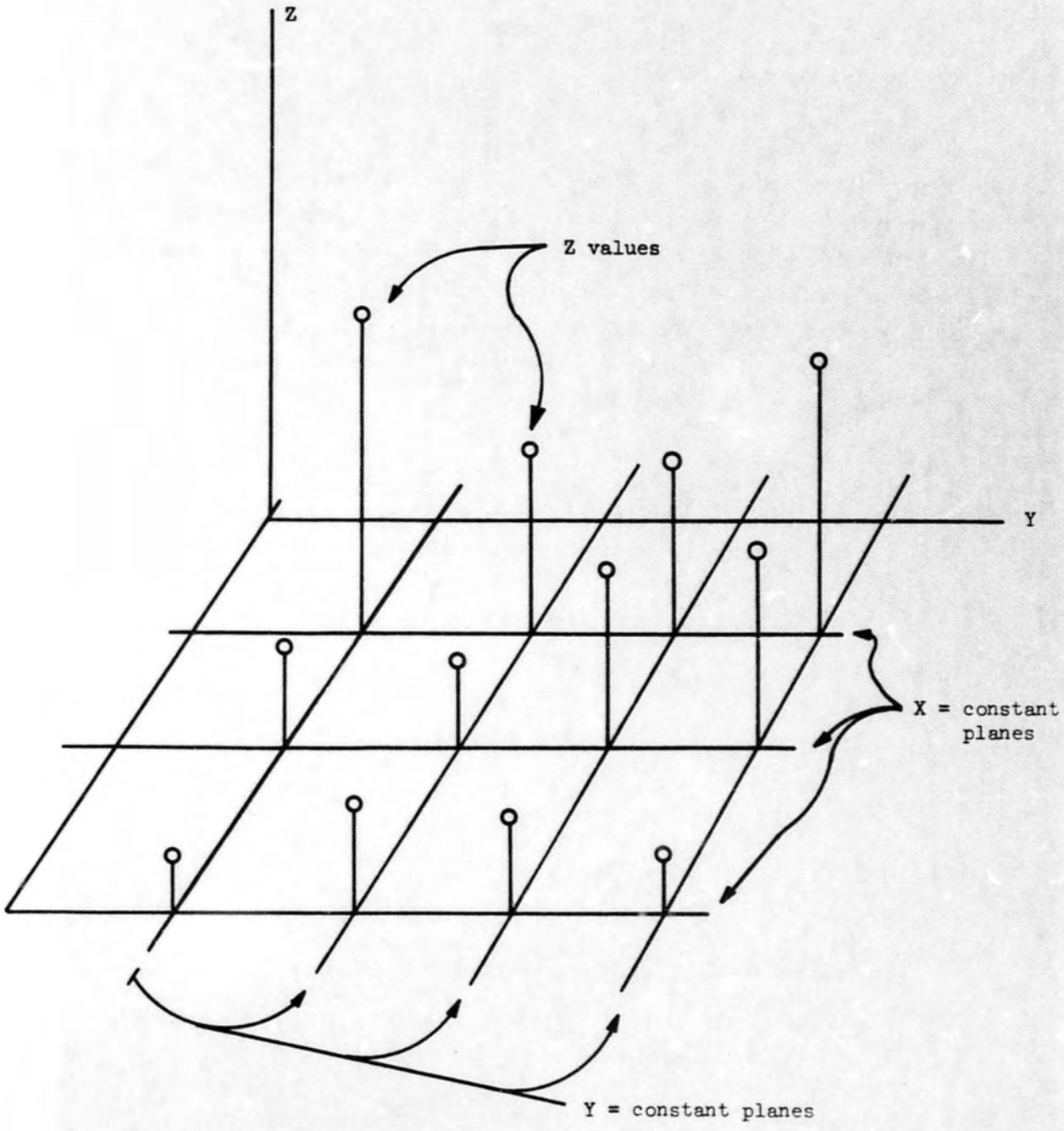


Figure 5. Illustration of Interpolation Scheme

$$(X_1, Y_2, Z_3)$$
$$(X_1, Y_1, Z_4)$$

The point to be interpolated is

$$(X_a, Y_a, Z_a)$$

If straight lines $Z_1'(X)$ and $Z_2'(X)$ are found, and the Z values at $X = X_a$ are determined, then a line in the YZ plane, $Z_3'(Y)$, can be constructed through the points determined by lines $Z_1'(X)$ and $Z_2'(X)$. This line can then be evaluated at $Y = Y_a$ to give Z_a. It can be shown that relation (32) and the above procedure give identical results. Further, since $\partial^2 Z/\partial Y \partial X = \partial^2 Z/\partial X \partial Y$, the above procedure could have started using YZ planes instead of XZ planes and the same result would have been obtained.

d. Functions

(1) One Independent Variable

The functional notation for one independent variable is

$$Z = Z(X) \tag{39}$$

Function (39) is described pointwise by

$$X_i \quad i = 1, N_x \tag{40}$$

$$Z_i \quad i = 1, N_z \tag{41}$$

$$N_x = N_z \tag{42}$$

(2) Subroutine Calling Sequence for One Independent Variable

The following list defines the parameters in the subroutine calling statement for a function of one independent variable.

Variable	Program symbol	Definition
X_a	X	Independent variable value
---	Y	Not used
X_i	TABX	An array of independent variable values in ascending sequence

Variable	Program symbol	Definition
Z_i	TABZ	Dependent variable values for X_i
---	TABY	Not used
N_z	NZ	The number of TABZ values
---	NY	Not used
Z	Z	The interpolated dependent variable at X_a
---	NDIM	Not used
---	NS	XXXO (XXX represents any number of digits)

(3) Two Independent Variables

A function of two independent variables is denoted by

$$Z = Z(X,Y) \tag{43}$$

The surface is described by giving the coordinates of the intersection of an x = constant plane, y = constant plane, and z at the intersection point. The set of YZ and XZ planes are shown in figure 6. Therefore the surface is described by

$$X_i = i = 1,N_x \tag{44}$$

$$Y_j \quad j = 1,N_y \tag{45}$$

$$Z_{ij} \quad N_z = N_x N_y \quad \text{total values} \tag{46}$$

The following list describes the subroutine calling statement parameters for a function of two independent variables.

Variable	Program symbol	Definition
X_a	X	First independent variable argument
Y_a	Y	Second independent variable argument
Z	Z	Interpolated dependent variable at X_a, Y_a
X_i	TABX	An array of values for the first independent variable. Must be in ascending sequence

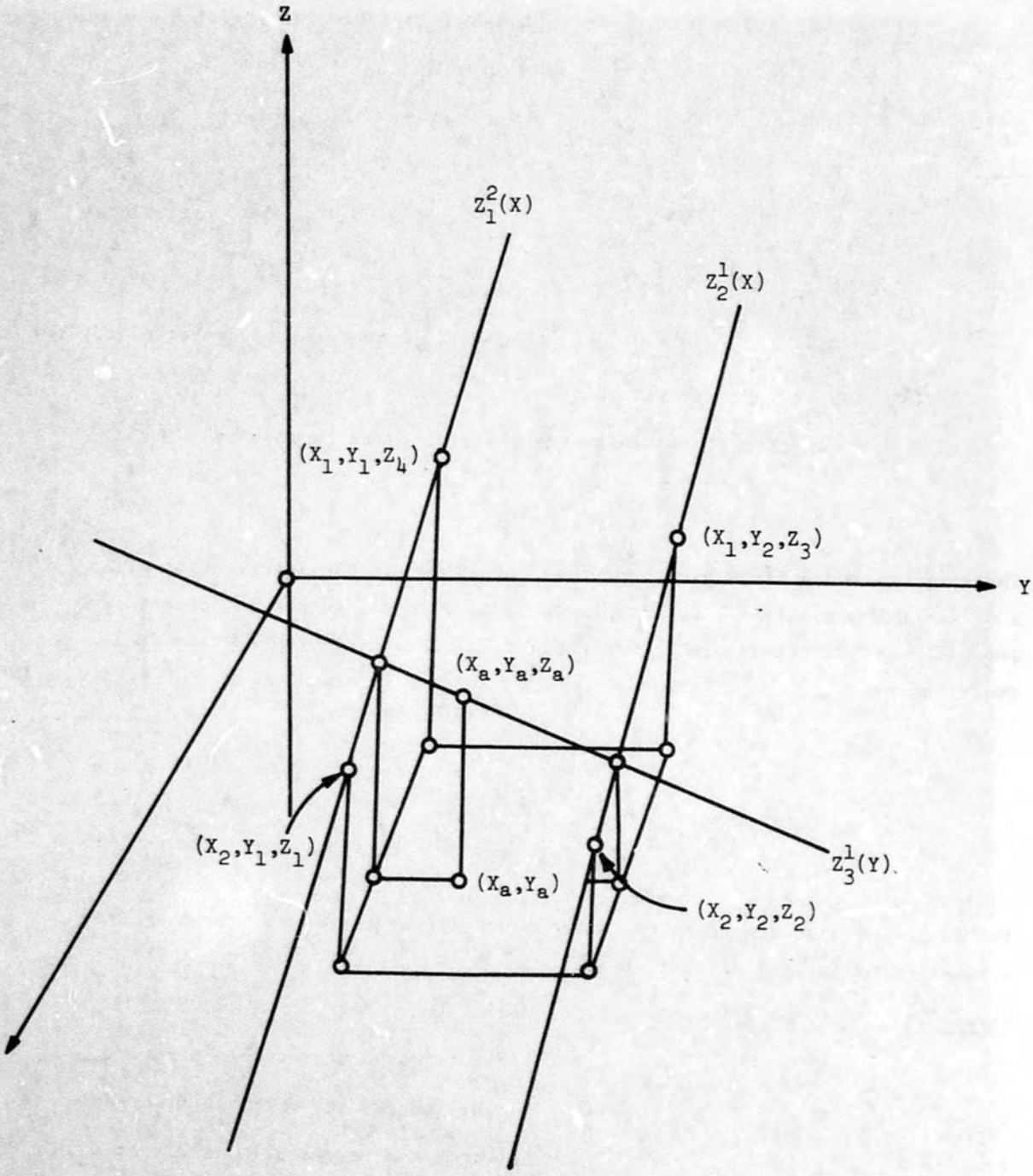


Figure 6. Illustration of Two-Dimensional Table Interpolations

Variable	Program symbol	Definition
Y_j	TABY	An array of values for the second independent variable. Must be in ascending sequence
Z_{ij}	TABZ	Dependent variable matrix
N_z	NZ	The number of elements in the Z matrix
N_y	NY	The number of elements in the Y_j vector
---	NS	XXX1 (XXX represents any number of digits to the left)
---	NDIM	The number of storage locations reserved in the Z_{ij} matrix for X_i values

The rather unusual form for NS resulted from subroutine compatibility reasons. NS is incoded in this form so programs that used a different interpolation subroutine would not have to be changed.

(4) Array Storage

Since the independent variable arrays are vectors--one-dimensional in the FORTRAN sense--there is no ambiguity in the way these arrays are stored. However, for the Z_{ij} matrix an ambiguity of storage can occur. The Z_{ij} matrix is stored in the following way.

Computer location relative to Z(1)	Value of Z_{ij}	Value of X_i for reference only (Storage layout is for Z)	Value of Y_j
1	Z_1	X_1	Y_1
2	Z_2	X_2	Y_1
.	.	.	.
.	.	.	.
.	.	.	.
N_x	Z_{N_x}	X_N	Y_1
.			
.	unused core if NDIM $\neq N_x$		
.			

Computer location relative to Z(1)	Value of Z_{ij}	Value of X_i	Value of Y_j
		for reference only (Storage layout is for Z)	
NDIM			
NDIM+1	$Z_{N_{x+1}}$	X_1	Y_2
NDIM+2	$Z_{N_{x+2}}$	X_2	Y_2
.	.	.	.
.	.	.	.
.	.		.
.	.	.	.
NDIM+N_x	Z_{2N_x}	X_{N_x}	Y_2
.			
.			
.	unused core if NDIM ≠ NX		
.			
2NDIM			
2NDIM+1	$Z_{2N_{x+1}}$	X_1	Y_3
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.

(NY blocks of storage setup as above)

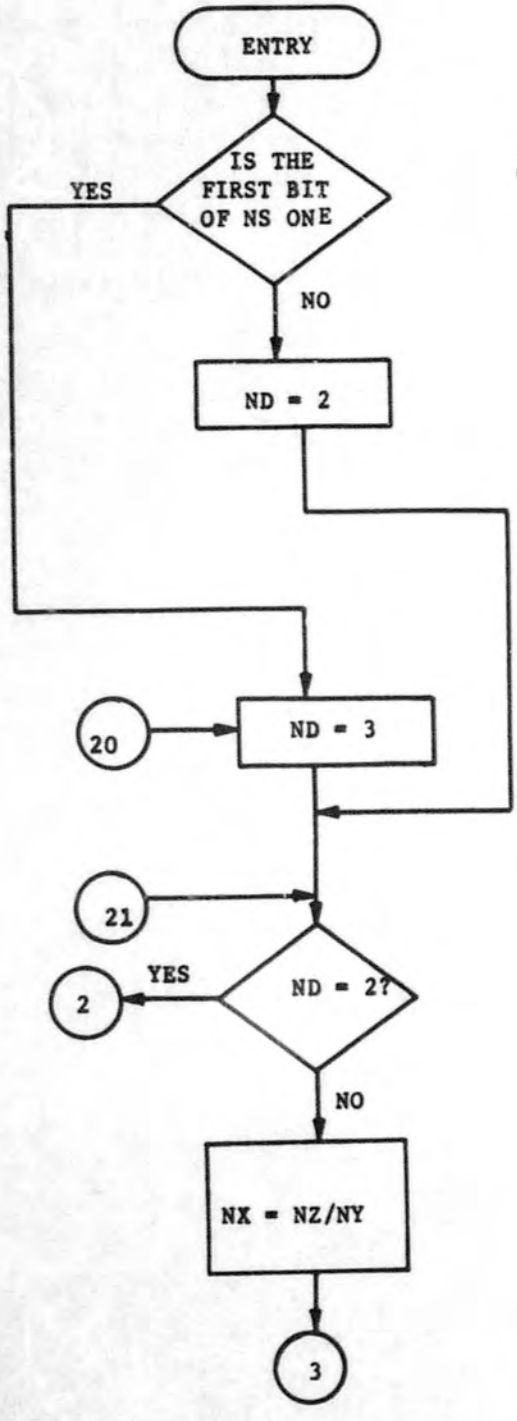
e. Program Symbols

The following list defines program symbols that do not appear in the calling sequence.

Variable	Program symbol	Definition
---	ND	Dimension of variable space
N	NX	The number of values in TABX table
$(X_a - X_X)$	DX	Distance away from the expansion point for one independent variable

Variable	Program symbol	Definition
$\left.\frac{dZ}{dX}\right\|_{X=X_X}$	DZDX	Derivative for one independent variable
$Z\left(X_K\right)$	ZO	Z at the expansion point for one independent variable
IX1	K	Expansion point TABX index
IX2	K-1	The other TABX point or plane index
IY1	L	Expansion point TABY index
IY2	L-1	The other TABY plane index
I1	---	Z matrix index for K and L
I2	---	Z matrix index for K-1 and L
I3	---	Z matrix index for K and L-1
I4	---	Z matrix index for K-1 and L-1
$\frac{\partial Z}{\partial X}$	DZDX1	Partial derivative with respect to X in the Y_L plane
$\left(\frac{\partial Z}{\partial X}\right)^1$	DZDX2	Partial derivative with respect to X in the Y_{L-1} plane
$\frac{\partial^2 Z}{\partial Y \partial X}$	DZDYDX	Cross partial
$\frac{\partial Z}{\partial Y}$	DZDY	Partial derivative with respect to Y in the X_K plane

f. Flow Chart

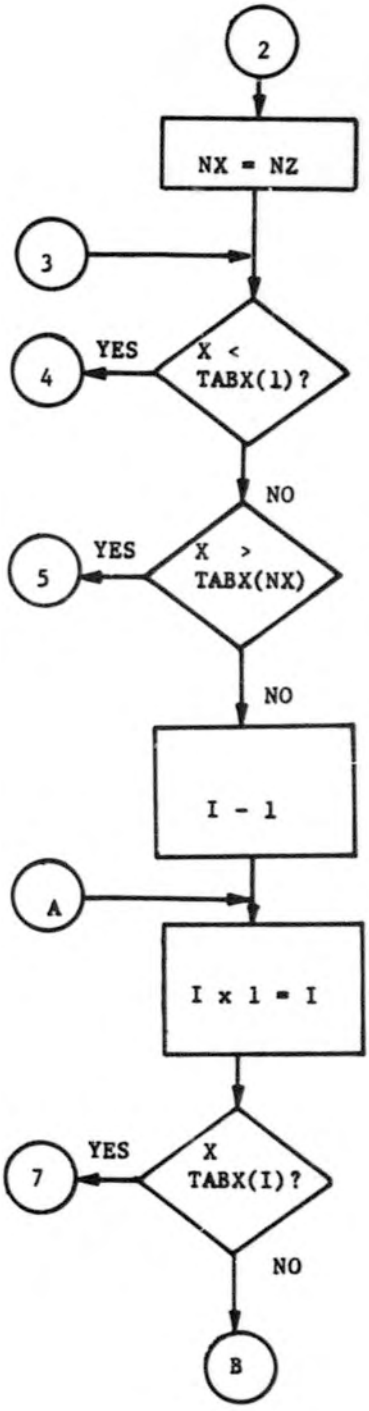


CHECK FOR ONE OR TWO INDEPENDENT VARIABLES

TWO INDEPENDENT VARIABLES
SET THE VARIABLE SPACE
DIMENSION TO THREE

ONE INDEPENDENT VARIABLE

CALCULATE THE NUMBER OF TABX
VALUES
TWO INDEPENDENT VARIABLE CASE



CALCULATE THE NUMBER OF TABX VALUES FOR ONE INDEPENDENT VARIABLE CASE

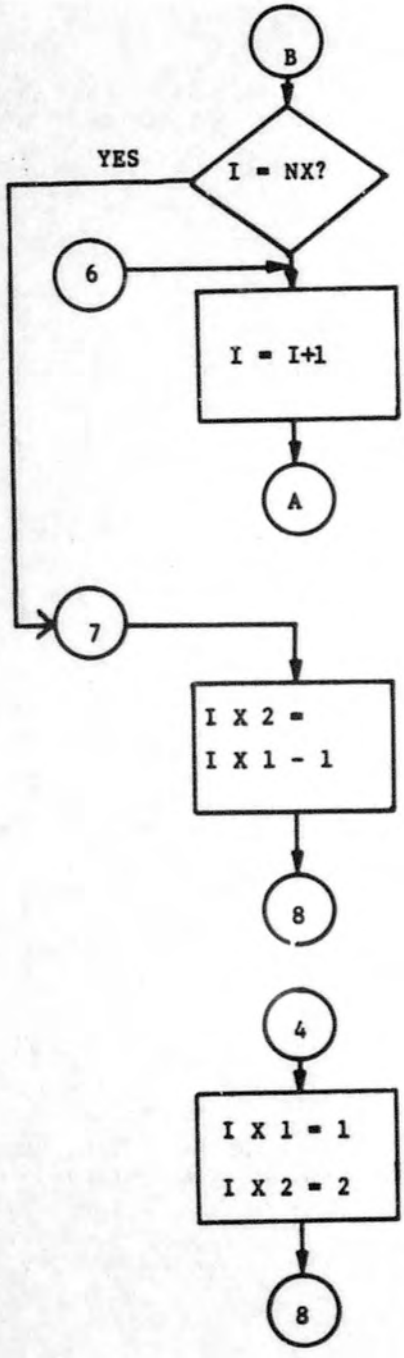
IS THE X ARGUMENT OFF THE LOWER END OF THE TABX TABLE?

IS THE x ARGUMENT OFF THE UPPER END OF THE TABX TABLE?

INITIALIZE DO-LOOP SEARCH INDEX AT 1

SAVE THE VALUE THAT WILL BE USED FOR K

HAVE WE FOUND THE FIRST TABX VALUE GREATER THAN THE X ARGUMENT?

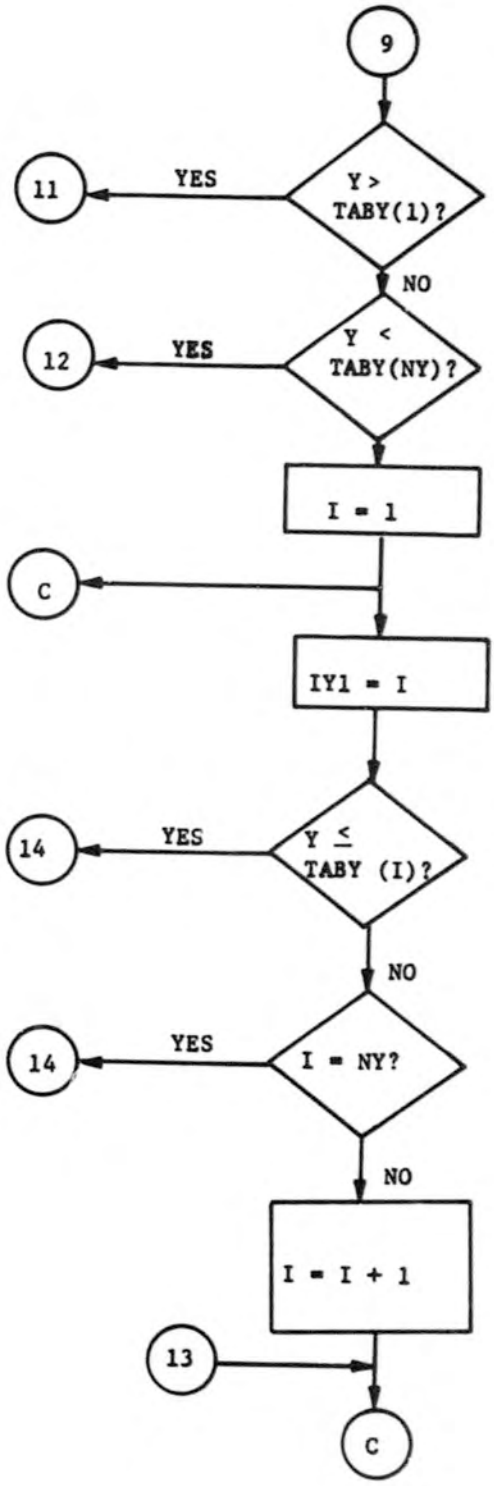


IS THE DO LOOP FINISHED?

INCREMENT THE DO LOOP INDEX

CALCULATE THE VALUE THAT WILL BE USED FOR K - 1

CALCULATE K AND K - 1 WHEN THE X TABLE MUST BE EXTRAPOLATED OFF THE LOWER END



IS THE Y ARGUMENT OFF THE LOWER END OF TABY?

IS THE Y ARGUMENT OFF THE UPPER END OF TABY?

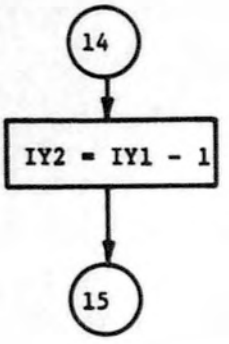
INITIALIZE THE DO LOOP INDEX FOR THE TABY TABLE SEARCH

SAVE THE VALUE THAT WILL BE USED FOR L

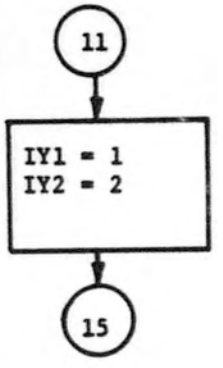
HAVE WE FOUND THE FIRST TABY VALUE GREATER OR EQUAL TO THE Y ARGUMENT?

HAVE WE CHECKED ALL TABY VALUES?

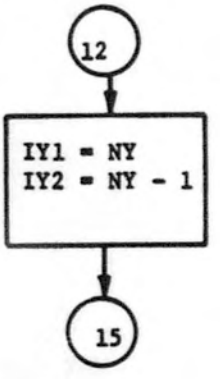
INCREMENT THE SEARCH INDEX



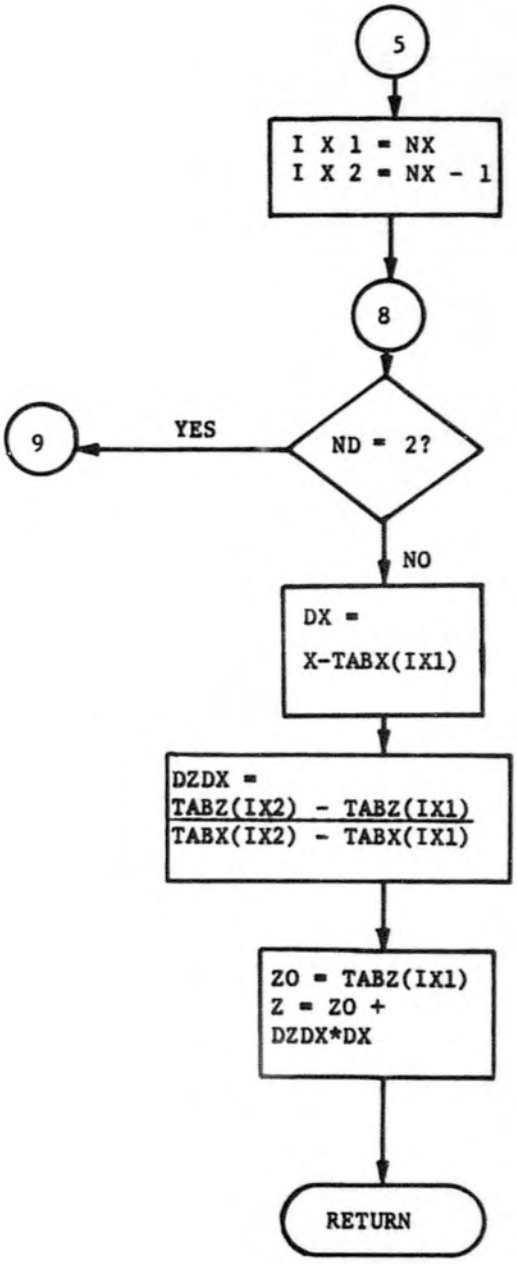
CALCULATE THE VALUE THAT
WILL BE USED FOR L - 1



SET UP THE VALUES OF L AND L - 1
WHEN THE Y TABLE, TABY, MUST BE
EXTRAPOLATED OFF THE LOWER END



SET UP THE VALUES OF L AND L - 1
WHEN TABY MUST BE EXTRAPOLATED
OFF THE UPPER END



CALCULATE K AND K - 1
WHEN THE X TABLE MUST BE
EXTRAPOLATED OFF THE UPPER END

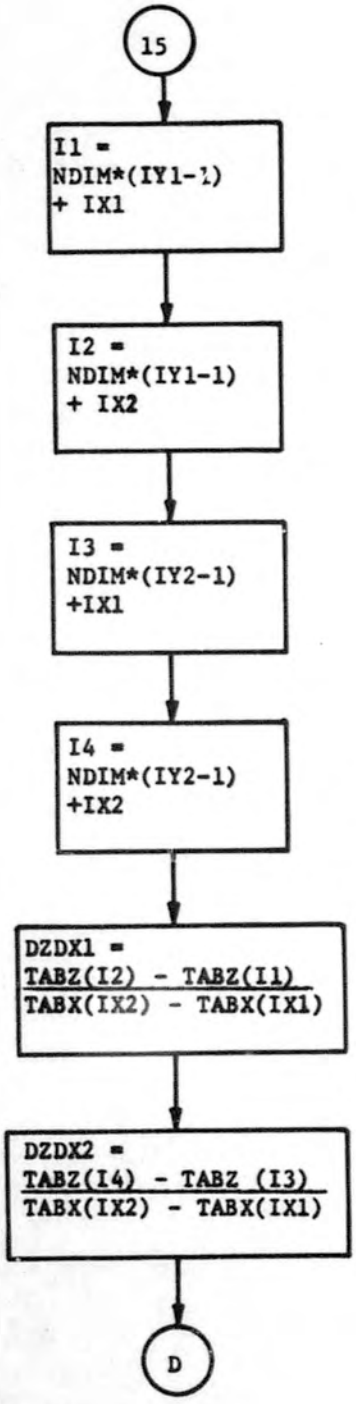
TWO INDEPENDENT VARIABLES

CALCULATE $(X_a - X_K)$ IN EQUATION (77)

ESTIMATE DZ/DX FOR EQUATION (77)

PICK UP Z AT THE EXPANSION POINT
EVALUATE Z ACCORDING TO EQUATION (77)

ONE INDEPENDENT VARIABLE CASE
FINISHED. BACK TO THE CALLING
PROGRAM



CALCULATE THE TABZ LOCATION FOR K, L

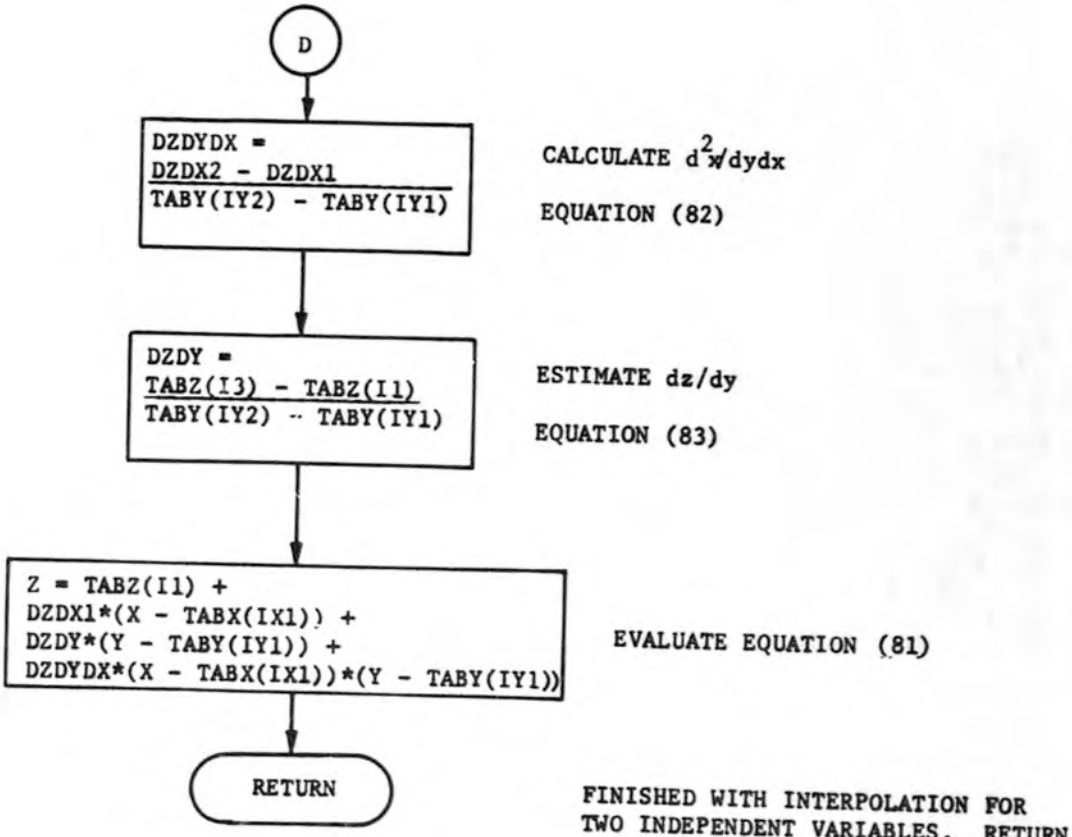
CALCULATE THE TABZ LOCATION FOR K-1, L

CALCULATE THE TABZ LOCATION FOR K, L-1

CALCULATE THE TABZ LOCATION FOR K-1, L-1

ESTIMATE DZ/DX FOR EQUATION (80)

ESTIMATE DZ/DX FOR EQUATION (81)


D
DZDYDX =
DZDX2 - DZDX1
TABY(IY2) - TABY(IY1)
CALCULATE d²z/dydx
EQUATION (82)
DZDY =
TABZ(I3) - TABZ(I1)
TABY(IY2) - TABY(IY1)
ESTIMATE dz/dy
EQUATION (83)
Z = TABZ(I1) +
DZDX1*(X - TABX(IX1)) +
DZDY*(Y - TABY(IY1)) +
DZDYDX*(X - TABX(IX1))*(Y - TABY(IY1))
EVALUATE EQUATION (81)
RETURN
FINISHED WITH INTERPOLATION FOR TWO INDEPENDENT VARIABLES. RETURN TO THE CALLING PROGRAM

g. **Listing**

```
      SUBROUTINE DISCOT (X,Y,TABX,TABZ,TABY,NS,NZ,NY,Z,NDIM)
      DIMENSION TABX(1),TABZ(1), TABY(1)
      IF(NS.AND.1) GO TO 20
      ND = 2
      GO TO 21
20    ND = 3
21    CONTINUE
C
C****          CHECK FOR 2-DIMENSIONAL CASE.
C
      IF(ND.EQ.2) GO TO 2
C
C****          CALCULATE THE NUMBER OF X VALUES FOR THE 3-DIMENSIONAL
C****          PLANE CUT CASE.
C
      NX=NZ/NY
      GO TO 3
C
C****          2-DIMENSIONAL CASE.
C
2     NX=NZ
C
C****          CHECK FOR X EXTRAPOLATION
C
3     IF(X.LE.TABX(1)) GO TO 4
      IF(X.GE.TABX(NX)) GO TO 5
C
C****          X IS IN THE TABX RANGE.
C
      DO 6 I=1,NX
      IX1=I
      IF(X.LE.TABX(I)) GO TO 7
6     CONTINUE
C
C****          WE HAVE FOUND AN X VALUE. SETUP INDICIES.
C
7     IX2=IX1-1
      GO TO 8
4     IX1=1
      IX2=2
      GO TO 8
5     IX1=NX
      IX2=NX-1
C
C****          X COORDINATE INDIFIES FOUND.
C****          CHECK FOR 2-DIMENSIONAL CASE.
C
8     IF(ND.GT.2) GO TO 9
C
C****          FINISH UP QUANTITIES FOR 2-DIMENSIONAL CASE.
C
      DX=X-TABX(IX1)
      DZDX=(TABZ(IX2)-TABZ(IX1))/(TABX(IX2)-TABX(IX1))
      Z0=TABZ(IX1)
      Z = Z0 +DZDX*DX
```

```
      RETURN
C
C****          BEGIN 3-DIMENSIONAL Y SEARCH.
9     IF(Y.LE.TABY(1)) GO TO 11
      IF(Y.GE.TABY(NY)) GO TO 12
C
C****          Y IS IN THE TABY RANGE.
C
      DO 13 I=1,NY
      IY1=I
      IF(Y.LE.TABY(I)) GO TO 14
13    CONTINUE
14    IY2=IY1-1
      GO TO 15
11    IY1=1
      IY2=2
      GO TO 15
12    IY1=NY
      IY2=NY-1
C
C****          X AND Y INDICIES FOUND AT THIS POINT.
C
C****          COMPUTE Z INDICIES.
15    CONTINUE
      I1=NDIM*(IY1-1)+IX1
      I2=NDIM*(IY1-1)+IX2
      I3=NDIM*(IY2-1)+IX1
      I4 = NDIM*(IY2-1)+IX2
C
C****          CALCULATE PARTIAL DERIVATIVES.
C
      DZDX1=(TABZ(I2)-TABZ(I1))/(TABX(IX2)-TABX(IX1))
      DZDX2=(TABZ(I4)-TABZ(I3))/(TABX(IX2)-TABX(IX1))
      DZDYDX=(DZDX2-DZDX1)/(TABY(IY2)-TABY(IY1))
      DZDY=(TABZ(I3)-TABZ(I1))/(TABY(IY2)-TABY(IY1))
C
C****          DO THE TAYLORS EXPANSION.
C
      XJ = TABZ(I1)+DZDX1*(X-TABX(IX1))+DZDY*(Y-TABY(IY1))
      XD = DZDYDX*(X-TABX(IX1))*(Y-TABY(IY1))
      Z = XJ + XD
      RETURN
      END
```

4. SUBROUTINE ARDC59

a. General

Subroutine ARDC59 produces average thermodynamic and transport properties of air as a function of altitude. The properties of air are calculated from the ARDC 1959 Model Atmosphere. The atmosphere is divided into 11 layers. Table VI gives the layer altitude bands.

The conversion of altitude to geopotential meters is

$$H_q = \frac{0.304h}{1.0 + 0.3048h/6356766} \tag{47}$$

where h is the altitude in feet and Hq is the altitude in geopotential meters.

The following models are used for layers 1, 3, 5, 7, 8, 9, 10, and 11 (see table VI).

$$T_m = \left(T_m\right)_b\left[1 + K_1\left(H_1 - H_b\right)\right], \left(H_q < 90\ K_m\right) \tag{48}$$

$$T = T_m\left[A - B\ \tan^{-1}\left(\frac{H_q - C}{D}\right)\right], \left(H_q > 90\ K_m\right) \tag{49}$$

$$P = P_b\left[1 + K_1\left(H_q - H_b\right)\right]^{-K_2} \tag{50}$$

$$\rho = \rho_b\left[1 + K_1\left(H_q - H_b\right)\right]^{-\left(1+K_2\right)} \tag{51}$$

$$V_s = 49.020576\left(T_m\right)^{1/2} \tag{52}$$

$$v = 0.0226988 \times 10^{-6}\left[\frac{T^{3/2}}{(T + 198.72)}\right] \tag{53}$$

For layers 2, 4, and 6, the following equations are substituted for equations (50) and (51).

$$P = P_b \exp\left[-K_3\left(H_q - H_b\right)\right] \tag{54}$$

$$\rho = \rho_b \exp\left[-K_3\left(H_q - H_b\right)\right] \tag{55}$$

In equations (48) through (55), T_m and T are temperature, P is pressure, ρ is density, V_s is the velocity of sound, and v is the viscosity. K_1, K_2, K_3, T_b, P_b, ρ_b, H_b, A, B, C, and D are "constants" for given altitude bands. The dependence of the above "constants" on altitude is given in tables VII and VIII.

Sample calling sequence:

```
CALL ARDC59 (ALT, RHO, VSONIC, T, P, XNU);
```

where

ALT	is the height above sea level in feet (input value)
RHO	atmospheric density in slugs/ft^3 (output value)
VSØNIC	speed of sound in ft/sec (output value)
T	atmospheric temperature in °R (output value)
P	atmospheric pressure in lb/ft^2 (output value)
XNU	atmospheric viscosity in ft^2/sec (output value)

Table VI

ALTITUDE LAYERS FOR ATMOSPHERE MODEL

Layer	Altitude range (geopotential meters)
1	0 - 11000
2	11000 - 25000
3	25000 - 47000
4	47000 - 53000
5	53000 - 79000
6	79000 - 90000
7	90000 - 105000
8	105000 - 160000
9	160000 - 170000
10	170000 - 200000
11	200000 - 700000

Table VII

CONSTANTS FOR ATMOSPHERE MODEL

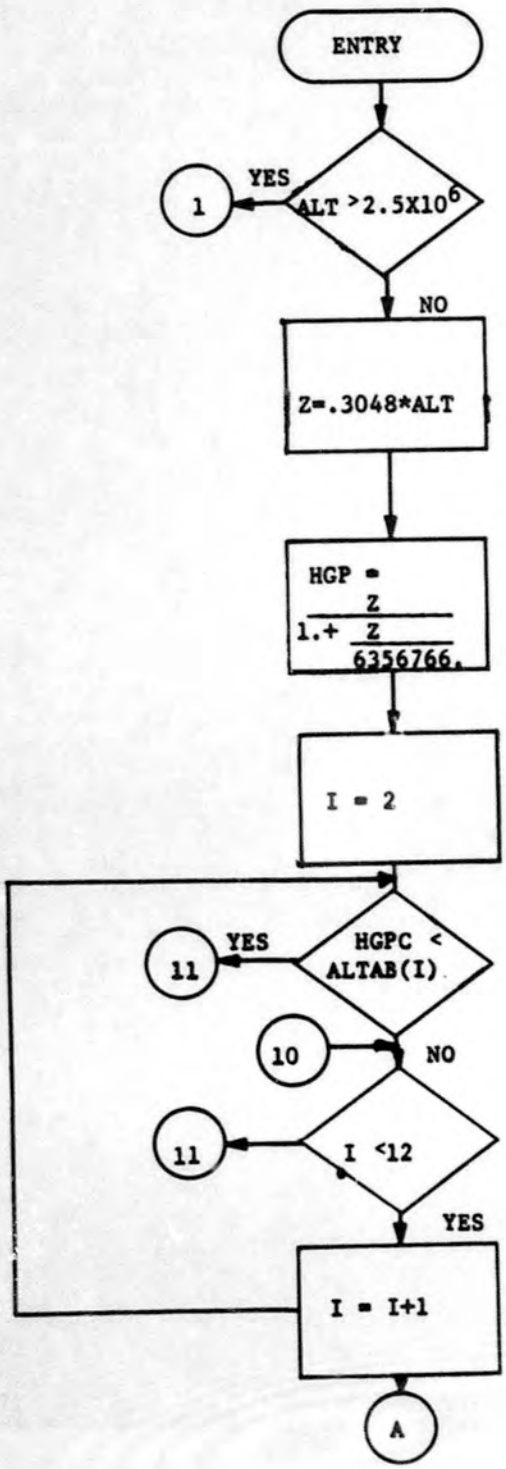
H_q (Km)	A	B	C	D
0 - 90	1.0	0.0	---	---
90 - 180	0.759511	0.174164	220.0	25.0
180 - 1200	0.935787	0.273966	180.0	140.0

Table VIII

SPECIFIC LAYER CONSTANTS

Layer	K_1	K_2	K_3	T_b	P_b	ρ_b	H_b
1	0.225569×10^{-4}	-5.25612	---	518.688	2116.22	2.37692×10^{-3}	0.0
2	0.0	---	0.157689×10^{-3}	389.988	472.73	7.0620×10^{-4}	11000.0
3	0.138466×10^{-4}	11.3883	---	389.988	51.979	7.7650×10^{-5}	25000.0
4	0.0	---	0.120869×10^{-3}	508.788	2.5155	2.8804×10^{-6}	47000.0
5	-0.159202×10^{-4}	-7.59218	---	508.788	1.2181	1.39468×10^{-6}	53000.0
6	0.0	---	0.206234×10^{-3}	298.188	2.1080×10^{-2}	4.1189×10^{-8}	79000.0
7	0.241458×10^{-4}	8.54120	---	298.188	2.1809×10^{-3}	4.261×10^{-9}	90000.0
8	0.886289×10^{-4}	1.70824	---	406.188	1.5562×10^{-4}	2.232×10^{-10}	105000.0
9	0.754341×10^{-4}	3.41648	---	2386.188	7.5578×10^{-6}	1.845×10^{-12}	160000.0
10	0.350715×10^{-5}	6.83296	---	2566.188	5.8954×10^{-6}	1.338×10^{-12}	170000.0
11	0.222129×10^{-5}	9.76137	---	2836.188	2.9759×10^{-6}	6.113×10^{-13}	200000.0

b. Flow Chart



IS THE ALTITUDE ABOVE THE UPPER CUTOFF

CONVERT THE ALTITUDE FROM FEET TO METERS

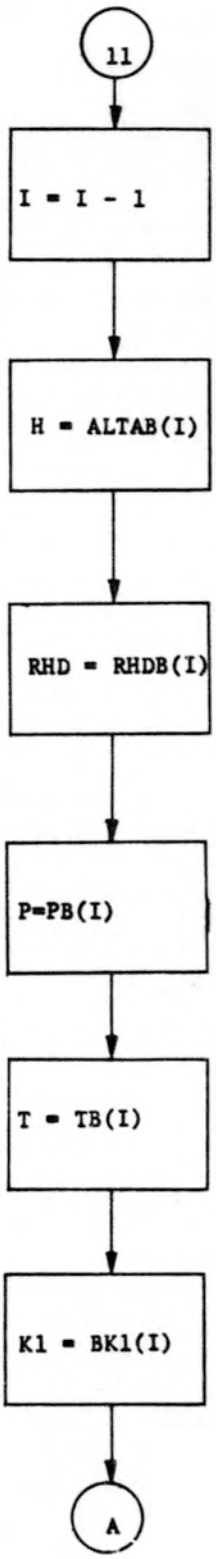
CONVERT THE ALTITUDE FROM METERS TO GEOPOTENTIAL METERS

INITIALIZE DO INDEX FOR ALTITUDE LAYER SEARCH

IS THE INPUT ALTITUDE, HGP, LESS THAN THE ITH LAYER UPPER ALTITUDE LIMIT?

ALL ALTITUDE LIMITS CHECKED

INCREMENT SEARCH INDEX



THE ALTITUDE LIMIT FOUND BY THE ABOVE SEARCH WAS ONE TOO HIGH IN THE TABLE. REDUCE THE SEARCH INDEX TO GET THE PROPER LAYER INDEX

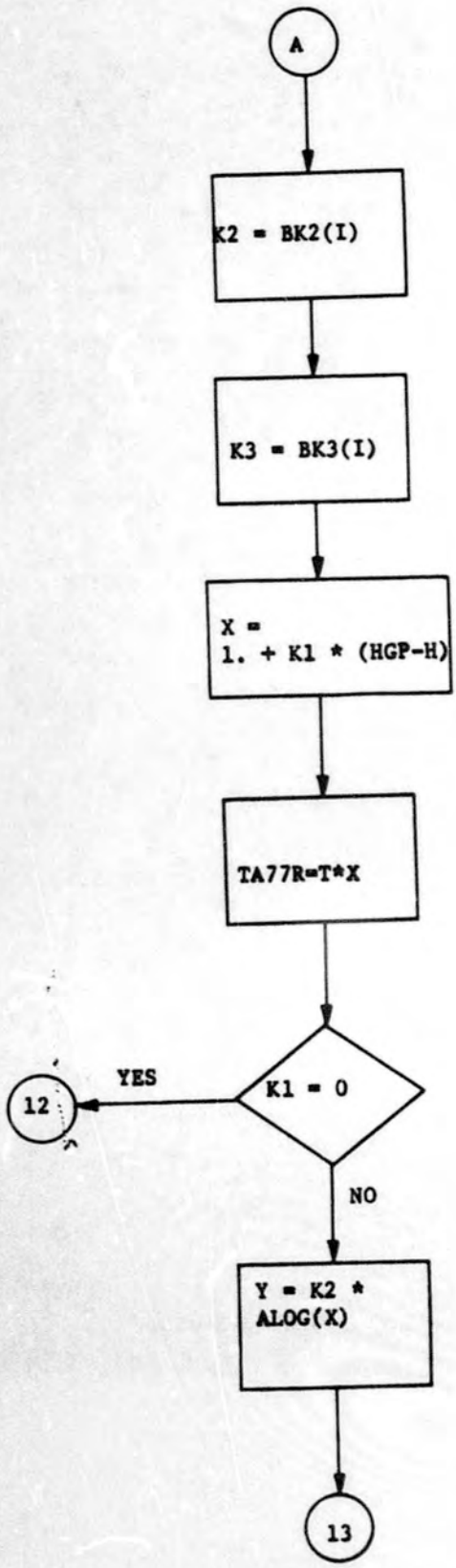
GET THE VALUE OF H_b FOR THIS LAYER

GET P_b FOR THIS LAYER

GET P_b FOR THIS LAYER

GET $(T_m)_b$ FOR THIS LAYER

GET K_1 FOR THIS LAYER



GET K_2 FOR THIS LAYER

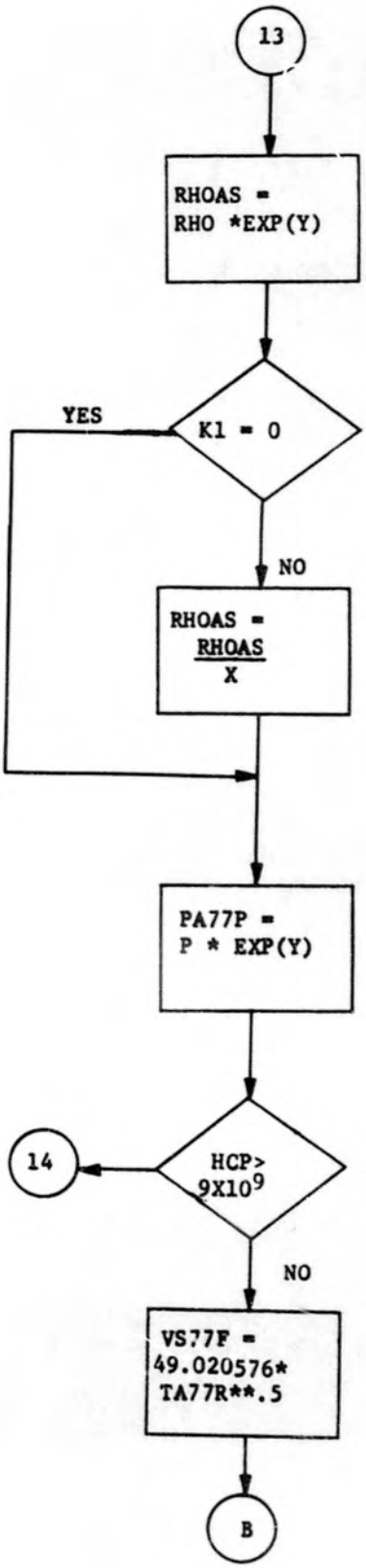
GET K_3 FOR THIS LAYER

CALCULATE T_M

CHECK TO DETERMINE IF THE INPUT ALTITUDE IS IN LAYERS 2, 4, OR 6

CALCULATE LN $\left(\frac{P}{P_b}\right)$

LAYERS 1, 3, 5, 7, 8, 9, 10, AND 11



CALCULATE THE ATMOSPHERIC DENSITY FOR LAYERS 3, 4, AND 6

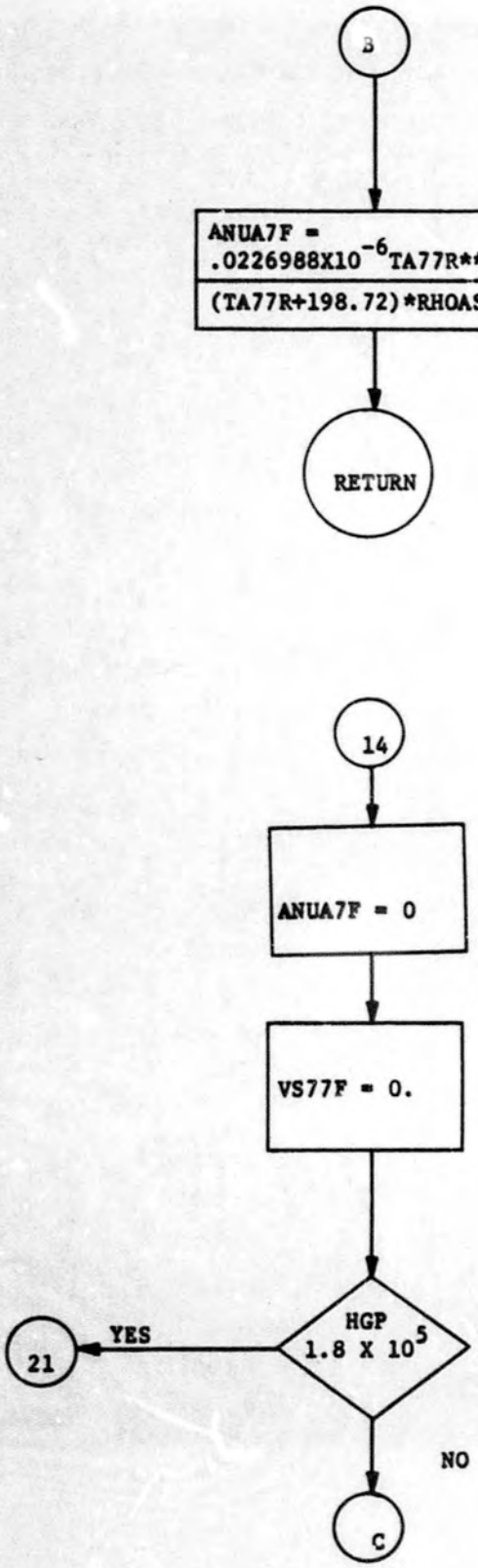
CHECK FOR LAYERS 2, 4, AND 6

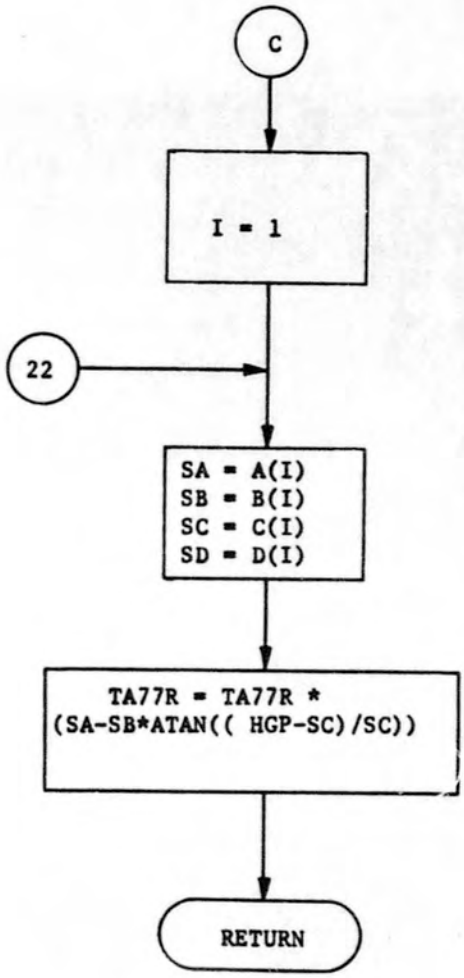
CALCULATE THE ATMOSPHERIC DENSITY FOR LAYERS 1, 3, 5, 7, 8, 9, 10, AND 11

CALCULATE THE ATMOSPHERIC PRESSURE

IS THE ALTITUDE ABOVE 90000 GEOPOTENTIAL METERS

CALCULATE THE SONIC VELOCITY

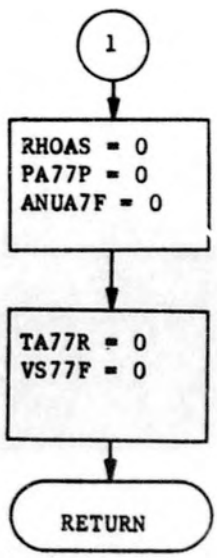

B
ANUA7F =
.0226988X10^-6TA77R**1.5
(TA77R+198.72)*RHOAS
RETURN
CALCULATE THE ATMOSPHERIC VISCOSITY
14
ANUA7F = 0
ABOVE 90000 GEOPOTENTIAL METERS THE ATMOSPHERIC VISCOSITY IS ZERO
VS77F = 0.
ABOVE 90000 GEOPOTENTIAL METERS THE SPEED OF SOUND IS ZERO
HGP
1.8 X 10^5
YES
21
IS THE GEOPOTENTIAL ALTITUDE GREATER THAN 180000 METERS?
NO
C



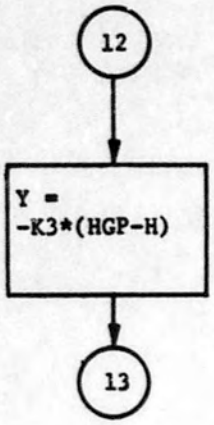
SET THE INDEX TO PICKUP A, B, C, AND D FOR THE 90000-180000 GEOPOTENTIAL METER BAND

PICKUP A, B, C, AND D VALUES GIVEN IN TABLE IV

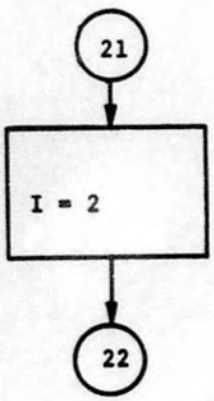
COMPUTE THE TEMPERATURE ABOVE 90000 GEOPOTENTIAL METERS



ABOVE AN ALTITUDE OF 2.5×10^6 FEET ALL PROPERTIES ARE ZERO



COMPUTE Y FOR BANDS 2, 4, and 6



SET UP THE INDEX USED TO FIND A, B, C,
AND FOR 180 - 1200 GEOPOTENTIAL KM

c. **Listing**

```
      SUBROUTINE ARDC59(ALT,RHOAS,VS77F,T/ 7R,PA77P,ANUA7F)
      REAL  K1,K2,K3
      DIMENSION ALTAB(12),BK1(11),BK2(11),BK3(11),TB(11),PB(11),
     1RHOB(11),A(2),B(2),C(2),D(2)
      DATA ALTAB/0.,11.E3,25.E3,47.E3,53.E3,79.E3,9.E4,
     1105.E3,16.E4,17.E4,2.E5,7.E5/
      DATA BK1/-.225569E-4,0.,.138466E-4,0.,-.159202E-4,
     10.,.241458E-4,.886289E-4,.754341E-5,.350715E-5,
     2.222129E-5/
      DATA BK2/-5.25612,0.,11.3883,0.,-7.59218,0.,
     18.54120,1.70824,3.41648,6.83296,9.76137/
      DATA BK3/0.,.157689E-3,0.,.120869E-3,0.,
     1.206234E-3,5*0.0/
      DATA TB/518.688,2*389.988,2*508.788,
     12*298.188,406.188,2386.188,2566.188,2836.188/
      DATA PB/2116.21695,472.73,51.979,2.5155,1.21801,
     1.10819E-2,2.1811E-3,1.5562E-4,7.5578E-6,
     25.8954E-6,2.9759E-6/
      DATA RHOB/2.37692E-3,7.0620E-4,7.7650E-5,
     1.8804E-6,1.39468E-6,4.11819E-8,4.261E-9,
     22.232E-10,1.845E-12,1.338E-12,6.113E-13/
      DATA A,B,C,D/.759511,.935787,.174164,.273966,
     1220.,.180,.25,.140./
      IF(ALT.GT.2.5E6)GO TO 1
      Z=.3048*ALT
      HGP=Z/(1.+Z/6356766.)
      DO 10 I=2,12
      IF(HGP.LT.ALTAB(I))GO TO 11
10    CONTINUE
11    I=I-1
      H=ALTAB(I)
      RHO=RHOB(I)
      P=PB(I)
      T=TB(I)
      K1=BK1(I)
      K2=BK2(I)
      K3=BK3(I)
      X=1.+K1*(HGP-H)
      TA77P=T*X
      IF(K1.EQ.0.)GO TO 12
      Y= -K2*ALOG(X)
13    RHOAS =RHO*EXP(Y)
      IF(K1.NE.0.)RHOAS=RHOAS/X
      PA77P =P* EXP(Y)
      IF(HGP.GE.9.E4)GO TO 14
      VS77F=49.020576*TA77R**.5
      ANUA7F=TA77R**1.5/((TA77R+198.72)*RHOAS)*.0226988E-6
      RETURN
14    ANUA7F=0.
      VS77F=0.
      IF(HGP.GE.18.E4)GO TO 21
      I=1
22    SA=A(I)
      SB=B(I)
      SC=C(I)
```

```
      SD=D(I)
      TA77R=TA77R*(SA-SB*ATAN((HGP/1000.-SC)/SD))
      RETURN
1     RHOAS=0.
      PA77P=0.
      ANUA7F=0.
      TA77R=0.
      VS77F=0.
      RETURN
12    Y=-K3*(HGP-H)
      GO TO 13
21    I=2
      GO TO 22
      END
```

5. SUBROUTINE HANSEN

a. General

Subroutine HANSEN is used to provide the thermodynamic and transport properties of equilibrium air. This subroutine is used to provide the atmospheric properties of the boundary layer of the vehicle. Subroutine HANSEN sets up the arguments for a linear interpolation subroutine, DISCOT. DISCOT does a "table lookup" to determine the required property. The properties are set up in five block data subprograms, BLK1, BLK2, BLK3, BLK4, and BLK5, as shown in paragraph f. The data in the block data subprograms are adapted from reference 14.

Subroutine HANSEN has nine entry points. Each entry point provides a different property, or a different state variable dependence. The entry points are described below.

IENTRY = 1 Finds the pressure ratio across a normal shock as a function of density and velocity.

IENTRY = 2 Finds the temperature ratio across a normal shock as a function of pressure and density.

IENTRY = 3 Finds the enthalpy as a function of temperature and pressure.

IENTRY = 4 Finds the temperature as a function of enthalpy and pressure.

IENTRY = 5 Compressibility as a function of temperature and pressure

IENTRY = 6 Prandtl number as a function of temperature and pressure.

IENTRY = 7 Lewis number as a function of temperature and pressure.

IENTRY = 8 Viscosity as a function of temperature and pressure.

IENTRY = 9 Dissociation enthalpy as a function of temperature.

The following are sample entries into subroutine HANSEN.

(1) CALL HANSEN (1, V', ρ', P')

$$V' = \mathrm{Log}_{10}\left(V/V_o\right) \qquad V_o = 1090 \text{ ft/sec} \qquad (56)$$

V has units ft/sec

$$\rho' = \mathrm{Log}_{10}\left(\frac{P}{T}\,\frac{491.69}{2116.22}\right) \qquad (57)$$

P is pressure with units lb/ft^2

T is temperature in degrees Rankine

Hence, V' and P' are the logs of the velocity and density normalized to standard conditions. P' is the output argument and is the log of pressure ratio across a normal shock.

$$P' = \text{Log}_{10}\left(\frac{P_2}{P_1}\right) \tag{58}$$

where

P_2 is the pressure behind a normal shock

P_1 is the free-stream pressure

(2) CALL HANSEN (2, P', ρ', T')

P' and ρ' are defined the same as in (1).

T' is the $\log_{10}$ of the temperature ratio across a normal shock.

$$T' = \text{Log}_{10}\left(\frac{T_2}{T_o}\right) \qquad T_o = 491.69°R \tag{59}$$

P' and ρ' are input arguments and T' is the output argument.

(3) CALL HANSEN (3, T", P', H")

T' is the temperature in °R.

P' is the log of the normalized pressure $\text{Log}_{10}\left(\frac{P}{P_o}\right)$

H" enthalpy in BTU/lb.

T" and P' are input arguments and H" is an output argument.

(4) CALL HANSEN (4, H", P", T")

H" and P' are input arguments and T" is the output argument.

H", T", and P' are defined in (3).

(5) CALL HANSEN (5, T", P', K')

T" is the temperature in degrees R.

P' is defined in (3) and K' is the compressibility.

T" and P' are input arguments.

(6) CALL HANSEN (6, T", P', ψ)

ψ is the Prandtl number and is the output argument.

T" is the input temperature °R.

P' is input and is defined in (3).

(7) CALL HANSEN (7, T", P', L)

T" input temperature °R.

P' input and defined by (3).

L output Lewis number.

(8) CALL HANSEN (8, T", P', η')

T" is the temperature in °R.

P' is defined by (3).

η' the normalized viscosity coefficient.

$$\eta = \eta' \left(2.28 \times 10^{-8} \, T^{1/2} \left(1 + \frac{202}{T} \right) \right) \qquad (60)$$

(9) CALL HANSEN (9, T", P', H')

T" is the input temperature in °R.

P' is input and defined in (3).

H' is output and is dimensionless enthalpy.

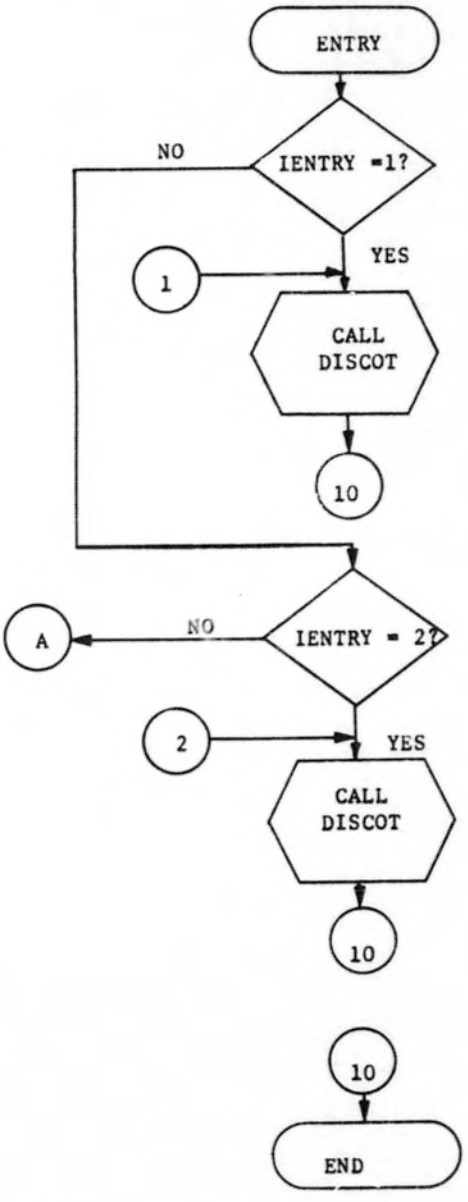
b. Symbols

Symbolic representation	Meaning or use
XA1	Normalized velocity
YA1	Normalized pressure
ZA1	Normalized density
XA2	Normalized pressure
YA2	Normalized temperature
ZA2	Normalized density
XA3	Temperature in °R
YA3	Enthalpy BTU/lb
ZA3	Normalized Pressure
XA11	Enthalpy BTU/lb
YA13	Temperature °R
ZA11	Normalized pressure
YA4	Compressibility
XA5	Temperature °R
YA5	Prandtl number
ZA3	Normalized pressure
YA6	Lewis number
YA7	Normalized viscosity
XA8	Temperature °R
YA8	Dimensionless enthalpy
ZA8	Normalized pressure

c. Versions of HANSEN

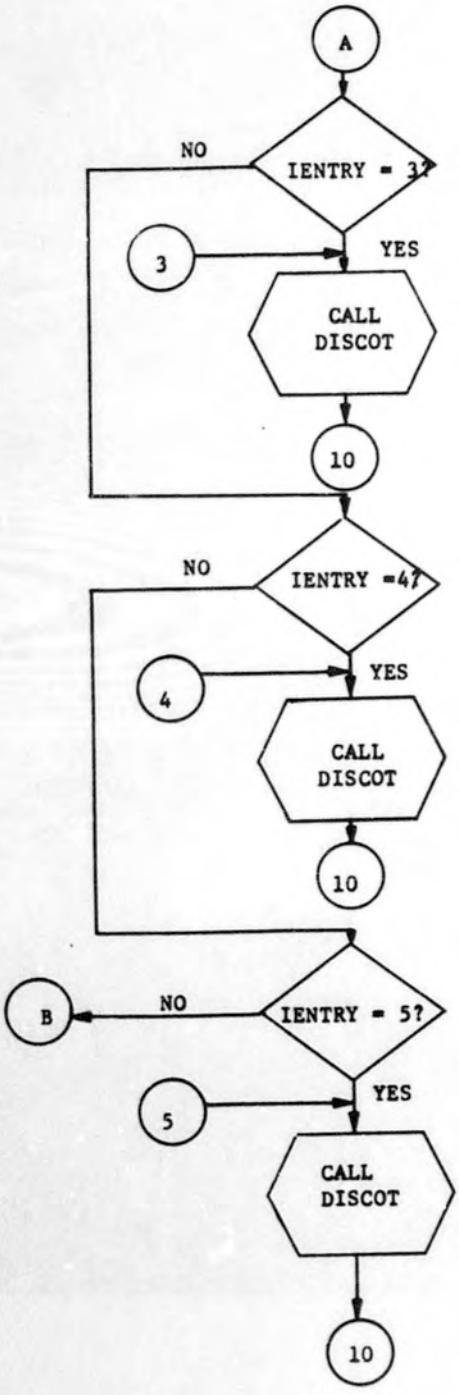
To conserve computer storage space, two versions of the basic logic are used in the boundary layer program. These two versions are identified as the long version and the short version. The long version is used in Segment FOUR and the short version is used in Segments ONE, THREE, and FIVE. While the long version can calculate all of the parameters described in subsection a, the short version calculates only the enthalpy and compressibility as functions of pressure and temperature. The block data for each version is given in subsection f. A listing for each version is given in subsection e.

d. Flow Chart (Long Version)



PERFORM A LINEAR INTERPOLATION TO FIND THE PRESSURE RATIO AS A FUNCTION OF VELOCITY AND DENSITY

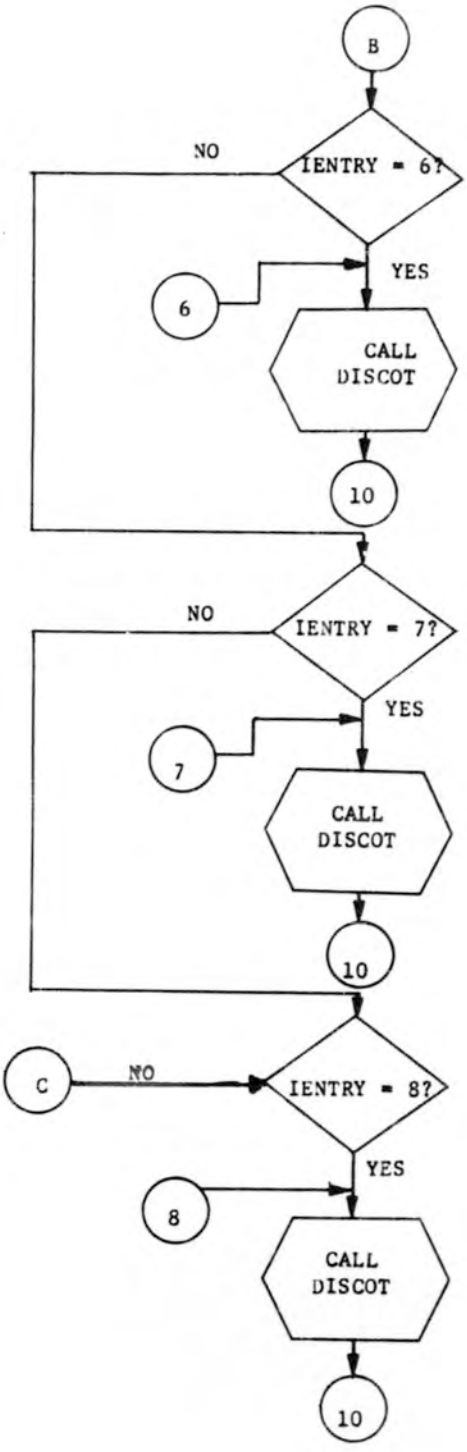
PERFORM A LINEAR INTERPOLATION TO FIND THE TEMPERATURE RATIO AS A FUNCTION OF PRESSURE AND DENSITY



PERFORM A LINEAR INTERPOLATION TO FIND ENTHALPY AS A FUNCTION OF TEMPERATURE AND PRESSURE

PERFORM A LINEAR INTERPOLATION TO FIND TEMPERATURE AS A FUNCTION OF ENTHALPY AND PRESSURE

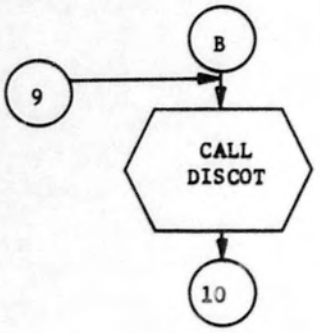
PERFORM A LINEAR INTERPOLATION TO FIND COMPRESSIBILITY AS A FUNCTION OF TEMPERATURE AND PRESSURE



PERFORM A LINEAR INTERPOLATION TO FIND THE PRANDTL NUMBER AS A FUNCTION OF TEMPERATURE AND PRESSURE

PERFORM A LINEAR INTERPOLATION TO FIND THE LEWIS NUMBER AS A FUNCTION OF TEMPERATURE AND PRESSURE

PERFORM A LINEAR INTERPOLATION TO FIND THE VISCOSITY AS A FUNCTION OF TEMPERATURE AND PRESSURE



PERFORM A LINEAR INTERPOLATION TO FIND THE DISSOCIATION ENTHALPY AS A FUNCTION OF TEMPERATURE AND DENSITY

E. LISTING

(1) LONG VERSION

```
      SUBROUTINE HANSEN(IENTRY,A1,A2,A3)
      DIMENSION  ZA8(5),XA11(47),ZA11(7),YA13(329)
      DIMENSION XA1(6),YA1(12),ZA1(2),XA2(40),YA2(320),ZA2(8),XA3(31),
     1 YA3(217),ZA3(7),YA4(217),XA5(21),YA5(147),YA6(147),YA7(217),
     2 XA8(40),YA8(200)
      COMMON /BLK005/ ZA8,XA11,ZA11,YA13
      COMMON /BLK001/XA1,YA1,ZA1,XA2,YA2,ZA2/BLK002/XA3,YA3,ZA3,YA4
     * /BLK003/ XA5,YA5,YA6/BLK004/ YA7,XA8,YA8
      GO TO (1,2,3,4,5,6,7,8,9), IENTRY
C
C****          P2PIP ENTRY
C
1     CALL DISCOT(A1,A2,XA1,YA1,ZA1,11,12,2,A3,6)
      GO TO 10
C
C****          T2OTO ENTRY
C
2     CALL DISCOT(A1,A2,XA2,YA2,ZA2,11,320,8,A3,40)
      GO TO 10
C
C****          HVALU ENTRY
C
3     CALL DISCOT(A1,A2,XA3,YA3,ZA3,11,217,7,A3,31)
      GO TO 10
C
C****          TEMPVL ENTRY
C
4     CALL DISCOT(A1,A2,XA11,YA13,ZA11,11,329,7,A3,47)
      GO TO 10
C
C****          ZVALU ENTRY
C
5     CALL DISCOT(A1,A2,XA3,YA4,ZA3,11,217,7,A3,31)
      GO TO 10
C
C****          PRVAL ENTRY
C
6     CALL DISCOT(A1,A2,XA5,YA5,ZA3,11,147,7,A3,21)
      GO TO 10
C
C****          XLVAL  ENTRY
C
7     CALL DISCOT(A1,A2,XA5,YA6,ZA3,11,147,7,A3,21)
      GO TO 10
C
C****          SMVAL ENTRY
C
8     CALL DISCOT(A1,A2,XA3,YA7,ZA3,11,217,7,A3,31)
      GO TO 10
C
C****          HOMVL ENTRY
C
9     CALL DISCOT(A1,A2,XA8,YA8,ZA8,11,200,5,A3,40)
10    RETURN
      END
```

(2) SHORT VERSION

```
      SUBROUTINE HANSEN(IENTRY,A1,A2,A3)
      DIMENSION  ZA8(5),XA11(47),ZA11(7),YA13(329)
      DIMENSION XA3(31),YA3(217),ZA3(7),YA4(217)
      COMMON/BLK002/XA3,YA3,ZA3,YA4
  100 FORMAT(1H0/10X,8HIENTRY =I3,
     *31H, OUT OF RANGE FOR SHORT HANSEN)
      COMMON /BLK005/ ZA8,XA11,ZA11,YA13
      GO TO(1,1,3,4,5),IENTRY
C
    1 WRITE(6,100)IENTRY
      GO TO 10
C
C     ENTHALPY FUNCTION OF P AND T
C
    3 CALL DISCOT(A1,A2,XA3,YA3,ZA3,11,217,7,A3,31)
      GO TO 10
C
C****          TEMPVL ENTRY
C
4     CALL DISCOT(A1,A2,XA11,YA13,ZA11,11,329,7,A3,47)
      GO TO 10
C     COMPRESSIBILITY FUNCTION OF P AND T
C
    5 CALL DISCOT(A1,A2,XA3,YA4,ZA3,11,217,7,A3,31)
10    RETURN
      END
```

F. BLOCK DATA

(1) BLOCK 001

```
 BLOCK DATA
 DIMENSION  XA1(6),YA1(12),ZA1(2),XA2(40),YA2(76),AX1(76),AX2(76),
*AX3(76),AX4(16),ZA2(8)
 COMMON/BLK001/ XA1,YA1,ZA1,XA2,YA2,AX1,AX2,AX3,AX4,ZA2
 DATA XA1/
* 4.71291720E-01, 7.78151270E-01, 1.00000000E 00, 1.17609130E 00,
* 1.47712130E 00, 1.69897000E 00/
 DATA YA1/
* 1.00000000E 00, 1.64345270E 00, 2.11394340E 00, 2.47712130E 00,
* 3.07918130E 00, 3.52504490E 00, 1.00000000E 00, 1.63346850E 00,
* 2.09691000E 00, 2.45484490E 00, 3.06069790E 00, 3.49136170E 00/
 DATA ZA1/
*-7.00000000E 00, 0.00000000E-39/
 DATA XA2/
* 1.00000000E 00, 1.13033380E 00, 1.36172780E 00, 1.47712130E 00,
* 1.60206000E 00, 1.69897000E 00, 1.77815130E 00, 1.84509810E 00,
* 1.95424250E 00, 2.02118930E 00, 2.07918130E 00, 2.14612810E 00,
* 2.17609130E 00, 2.21748400E 00, 2.24303810E 00, 2.26717180E 00,
* 2.30103000E 00, 2.34242270E 00, 2.39794000E 00, 2.47712130E 00,
* 2.54406810E 00, 2.60206000E 00, 2.77815130E 00, 2.82607480E 00,
* 2.86923180E 00, 2.89762710E 00, 2.91907810E 00, 2.93449850E 00,
* 2.95424250E 00, 2.97312790E 00, 3.00000000E 00, 3.02118930E 00,
* 3.06069790E 00, 3.09691000E 00, 3.13033380E 00, 3.17609130E 00,
* 3.23044900E 00, 3.27875360E 00, 3.32221930E 00, 3.39794000E 00/
 DATA YA2/
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.28353790E-01,
* 7.97959650E-01, 8.36324120E-01, 8.57332500E-01, 8.69231720E-01,
* 8.89301710E-01, 9.05795890E-01, 9.21686490E-01, 9.44482680E-01,
* 9.63787840E-01, 1.02530590E 00, 1.06445800E 00, 1.08635980E 00,
* 1.10721000E 00, 1.12385170E 00, 1.13987910E 00, 1.16136800E 00,
* 1.17609130E 00, 1.19033170E 00, 1.21748400E 00, 1.23044890E 00,
* 1.24551270E 00, 1.26007140E 00, 1.27184160E 00, 1.28330120E 00,
* 1.30535140E 00, 1.35024800E 00, 1.38738980E 00, 1.40312050E 00,
* 1.42813480E 00, 1.44404480E 00, 1.45636600E 00, 1.46982200E 00,
* 1.48429990E 00, 1.49554440E 00, 1.50515000E 00, 1.52762990E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.32393760E-01,
* 8.09559730E-01, 8.53089530E-01, 8.77946960E-01, 8.94869660E-01,
* 9.21686490E-01, 9.37016110E-01, 9.51823040E-01, 9.77723610E-01,
* 9.97823090E-01, 1.04139270E 00, 1.07188200E 00, 1.10037060E 00,
* 1.13033380E 00, 1.15228830E 00, 1.17026170E 00, 1.19033170E 00,
* 1.20682590E 00, 1.22010810E 00, 1.25527250E 00, 1.26717180E 00,
* 1.27875360E 00, 1.29003460E 00, 1.30319610E 00, 1.32014630E 00,
* 1.34242270E 00, 1.36735590E 00, 1.40824000E 00, 1.43136380E 00,
* 1.46239800E 00, 1.47856650E 00, 1.49276040E 00, 1.50920250E 00/
 DATA AX1/
* 1.52762990E 00, 1.54157920E 00, 1.55388300E 00, 1.57170880E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.40362700E-01,
* 8.22821650E-01, 8.75061270E-01, 9.05795890E-01, 9.24279290E-01,
* 9.54242520E-01, 9.68482960E-01, 9.84527330E-01, 1.01283720E 00,
* 1.02938380E 00, 1.06069790E 00, 1.08990510E 00, 1.11727130E 00,
* 1.15228830E 00, 1.17609130E 00, 1.20139710E 00, 1.22788670E 00,
* 1.24797330E 00, 1.26007140E 00, 1.30103000E 00, 1.31386720E 00,
* 1.32428250E 00, 1.33845650E 00, 1.34830490E 00, 1.35983550E 00,
* 1.37657700E 00, 1.39967370E 00, 1.44560420E 00, 1.47129170E 00,
* 1.50242710E 00, 1.52244420E 00, 1.53781910E 00, 1.55630250E 00,
```

```
* 1.57287160E 00, 1.58883170E 00, 1.60206000E 00, 1.62634040E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.40362700E-01,
* 8.29303780E-01, 8.87054380E-01, 9.24279290E-01, 9.49390020E-01,
* 9.86771740E-01, 1.00432140E 00, 1.02118930E 00, 1.04921800E 00,
* 1.06818590E 00, 1.09691000E 00, 1.11727130E 00, 1.13672060E 00,
* 1.17026170E 00, 1.20139710E 00, 1.23044890E 00, 1.26245110E 00,
* 1.28330120E 00, 1.30103000E 00, 1.34830490E 00, 1.36172780E 00,
* 1.37474830E 00, 1.38560630E 00, 1.39445170E 00, 1.40140060E 00,
* 1.41497330E 00, 1.43136380E 00, 1.46686760E 00, 1.49136170E 00/
 DATA AX2/
* 1.53147890E 00, 1.55630250E 00, 1.57749180E 00, 1.60206000E 00,
* 1.62838890E 00, 1.64738300E 00, 1.66275780E 00, 1.68930890E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.40362700E-01,
* 8.29303780E-01, 8.94869660E-01, 9.39519260E-01, 9.70811610E-01,
* 1.01283720E 00, 1.04139270E 00, 1.06818590E 00, 1.10037060E 00,
* 1.11394340E 00, 1.13353890E 00, 1.14612800E 00, 1.16136800E 00,
* 1.18752070E 00, 1.22271650E 00, 1.26245110E 00, 1.30103000E 00,
* 1.32428250E 00, 1.34242270E 00, 1.39445170E 00, 1.40993310E 00,
* 1.42813480E 00, 1.44247980E 00, 1.44870630E 00, 1.45939250E 00,
* 1.46982200E 00, 1.48000700E 00, 1.49831060E 00, 1.51587390E 00,
* 1.55630250E 00, 1.59879050E 00, 1.62428210E 00, 1.65513840E 00,
* 1.68124120E 00, 1.70243050E 00, 1.71933130E 00, 1.75127910E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.40362700E-01,
* 8.29303780E-01, 8.95974740E-01, 9.44482680E-01, 9.82271240E-01,
* 1.04921800E 00, 1.08635980E 00, 1.11058970E 00, 1.13672060E 00,
* 1.14921910E 00, 1.16435290E 00, 1.17609130E 00, 1.19033170E 00,
* 1.21484390E 00, 1.25042000E 00, 1.29446620E 00, 1.33845650E 00,
* 1.36548800E 00, 1.38916610E 00, 1.44715810E 00, 1.46389300E 00,
* 1.48000700E 00, 1.49554440E 00, 1.50242710E 00, 1.50785590E 00/
 DATA AX3/
* 1.51851400E 00, 1.52891670E 00, 1.54654270E 00, 1.56702640E 00,
* 1.60422610E 00, 1.64345270E 00, 1.67209790E 00, 1.70329140E 00,
* 1.73639650E 00, 1.76342800E 00, 1.78532980E 00, 1.82607480E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.40362700E-01,
* 8.29303780E-01, 8.97077010E-01, 9.51823040E-01, 1.00000000E 00,
* 1.07188200E 00, 1.11058970E 00, 1.13672060E 00, 1.16731730E 00,
* 1.18184360E 00, 1.20412000E 00, 1.22010810E 00, 1.23552850E 00,
* 1.25767860E 00, 1.29003460E 00, 1.32837960E 00, 1.38021120E 00,
* 1.41161970E 00, 1.43775060E 00, 1.50515000E 00, 1.52113810E 00,
* 1.53907610E 00, 1.55388300E 00, 1.55990660E 00, 1.56820180E 00,
* 1.57516790E 00, 1.58433120E 00, 1.59988310E 00, 1.61489720E 00,
* 1.64836000E 00, 1.68394710E 00, 1.71600340E 00, 1.75587490E 00,
* 1.79726760E 00, 1.83250890E 00, 1.85973860E 00, 1.90309000E 00,
* 4.16640510E-01, 4.94154600E-01, 6.57055850E-01, 7.40362700E-01,
* 8.29303780E-01, 9.00367130E-01, 9.59041400E-01, 1.00860020E 00,
* 1.08635980E 00, 1.13033380E 00, 1.16136800E 00, 1.20412000E 00,
* 1.22010810E 00, 1.24303810E 00, 1.26007140E 00, 1.27875360E 00,
* 1.30103000E 00, 1.32221930E 00, 1.36921590E 00, 1.41664050E 00,
* 1.45939250E 00, 1.48713840E 00, 1.56466610E 00, 1.58658730E 00/
 DATA AX4/
* 1.60638140E 00, 1.62324930E 00, 1.63042790E 00, 1.63648790E 00,
* 1.64542230E 00, 1.65417660E 00, 1.66745300E 00, 1.68124120E 00,
* 1.70757020E 00, 1.73399930E 00, 1.76342800E 00, 1.80277380E 00,
* 1.85733250E 00, 1.89762710E 00, 1.92582760E 00, 1.96378790E 00/
 DATA ZA2/
*-7.00000000E 00,-6.00000000E 00,-5.00000000E 00,-4.00000000E 00,
*-3.00000000E 00,-2.00000000E 00,-1.00000000E 00, 0.00000000E-39/
 END
```

(2) BLOCK 002

```
      BLOCK DATA
      DIMENSION XA3(31),YA3(76),AX5(76),AX6(65),ZA3(7),YA4(76),AX7(76),
     *  AX8(65)
      COMMON /BLK002/ XA3,YA3,AX5,AX6,ZA3,YA4,AX7,AX8
      DATA XA3/
     * 0.00000000E-39, 9.00000000E 02, 1.80000000E 03, 2.70000000E 03,
     * 3.60000000E 03, 4.50000000E 03, 5.40000000E 03, 6.30000000E 03,
     * 7.20000000E 03, 8.10000000E 03, 9.00000000E 03, 9.90000000E 03,
     * 1.08000000E 04, 1.17000000E 04, 1.26000000E 04, 1.35000000E 04,
     * 1.44000000E 04, 1.53000000E 04, 1.62000000E 04, 1.71000000E 04,
     * 1.80000000E 04, 1.89000000E 04, 1.98000000E 04, 2.07000000E 04,
     * 2.16000000E 04, 2.25000000E 04, 2.34000000E 04, 2.43000000E 04,
     * 2.52000000E 04, 2.61000000E 04, 2.70000000E 04/
      DATA YA3/
     * 0.00000000E-39, 2.17323000E 02, 4.50702000E 02, 7.03829990E 02,
     * 1.08910800E 03, 2.47577000E 03, 3.03388000E 03, 3.47042000E 03,
     * 4.85028000E 03, 9.33509000E 03, 1.44842400E 04, 1.60141400E 04,
     * 1.66993900E 04, 1.76014800E 04, 1.92665300E 04, 2.24672400E 04,
     * 2.83016200E 04, 3.75224900E 04, 4.86090700E 04, 5.76559000E 04,
     * 6.29254800E 04, 6.56567100E 04, 6.72077290E 04, 6.82600290E 04,
     * 6.91091990E 04, 6.98588990E 04, 7.05823590E 04, 7.12801590E 04,
     * 7.19838000E 04, 7.26569200E 04, 7.33471190E 04, 0.00000000E-39,
     * 2.17323000E 02, 4.50702000E 02, 7.03836000E 02, 1.00512000E 03,
     * 1.90161000E 03, 2.97092000E 03, 3.35803000E 03, 3.99578000E 03,
     * 5.86221000E 03, 1.02982500E 04, 1.46558600E 04, 1.62771100E 04,
     * 1.70476000E 04, 1.78835600E 04, 1.91887700E 04, 2.13867400E 04,
     * 2.50324800E 04, 3.07391800E 04, 3.87109800E 04, 4.78978200E 04,
     * 5.61142900E 04, 6.18967800E 04, 6.54486400E 04, 6.76274390E 04,
     * 6.90561000E 04, 7.01328400E 04, 7.10298690E 04, 7.18283190E 04,
     * 7.25673990E 04, 7.32915000E 04, 0.00000000E-39, 2.17323000E 02,
     * 4.50702000E 02, 7.03836000E 02, 9.80424000E 02, 1.48487000E 03,
     * 2.64125000E 03, 3.29320000E 03, 3.71923000E 03, 4.52304000E 03,
     * 6.47073000E 03, 1.02821400E 04, 1.42989800E 04, 1.63333200E 04/
      DATA AX5/
     * 1.72958900E 04, 1.80959400E 04, 1.91047700E 04, 2.05717700E 04,
     * 2.27708800E 04, 2.60067100E 04, 3.05983800E 04, 3.66661900E 04,
     * 4.38861100E 04, 5.13053600E 04, 5.78035400E 04, 6.27588000E 04,
     * 6.62323000E 04, 6.86295200E 04, 7.03415200E 04, 7.16364090E 04,
     * 7.26802200E 04, 0.00000000E-39, 2.17323000E 02, 4.50702000E 02,
     * 7.03836000E 02, 9.70560000E 02, 1.31814000E 03, 2.05594000E 03,
     * 3.05985000E 03, 3.59575000E 03, 4.07300000E 03, 4.91554000E 03,
     * 6.60805000E 03, 9.57960000E 03, 1.32111700E 04, 1.58523100E 04,
     * 1.72810800E 04, 1.82059200E 04, 1.90708400E 04, 2.01037100E 04,
     * 2.14552000E 04, 2.32759800E 04, 2.57232800E 04, 2.89450300E 04,
     * 3.30581100E 04, 3.80663300E 04, 4.37737500E 04, 4.97462900E 04,
     * 5.54603800E 04, 6.04532900E 04, 6.44923200E 04, 6.76609200E 04,
     * 0.00000000E-39, 2.17323000E 02, 4.50702000E 02, 7.03836000E 02,
     * 9.68076000E 02, 1.26256500E 03, 1.70775000E 03, 2.48504000E 03,
     * 3.32899000E 03, 3.87852000E 03, 4.38356000E 03, 5.14790000E 03,
     * 6.44566000E 03, 8.53983000E 03, 1.14095500E 04, 1.43360600E 04,
     * 1.65265900E 04, 1.79372600E 04, 1.89368300E 04, 1.98365100E 04,
     * 2.07939600E 04, 2.19115300E 04, 2.32673800E 04, 2.49497100E 04,
     * 2.70272200E 04, 2.95735500E 04, 3.26184300E 04, 3.62067600E 04/
      DATA AX6/
     * 4.02617900E 04, 4.47257400E 04, 4.93797600E 04, 0.00000000E-39,
     * 2.17323000E 02, 4.50702000E 02, 7.03836000E 02, 9.68076000E 02,
     * 1.24407000E 03, 1.57437000E 03, 2.05286000E 03, 2.74622000E 03,
```

```
* 3.49507000E 03, 4.08717000E 03, 4.61815000E 03, 5.26770000E 03,
* 6.15362000E 03, 7.57184000E 03, 9.48321000E 03, 1.18441400E 04,
* 1.43058100E 04, 1.64363600E 04, 1.80651200E 04, 1.92753000E 04,
* 2.02778100E 04, 2.11891700E 04, 2.21235300E 04, 2.31450500E 04,
* 2.42946000E 04, 2.56197200E 04, 2.71384800E 04, 2.88870100E 04,
* 3.09034400E 04, 3.31913700E 04, 0.00000000E-39, 2.17323000E 02,
* 4.50702000E 02, 7.03836000E 02, 9.68076000E 02, 1.23790500E 03,
* 1.52993000E 03, 1.87564000E 03, 2.32142000E 03, 2.88943000E 03,
* 3.53772000E 03, 4.16315000E 03, 4.72684000E 03, 5.31331990E 03,
* 6.00730000E 03, 6.89012990E 03, 8.06083000E 03, 9.55117990E 03,
* 1.13354600E 04, 1.33260300E 04, 1.53361800E 04, 1.71532600E 04,
* 1.87034800E 04, 1.99939200E 04, 2.10705800E 04, 2.20412300E 04,
* 2.29549300E 04, 2.38545800E 04, 2.47897400E 04, 2.57826200E 04,
* 2.68382700E 04/
 DATA ZA3/
*-4.00000000E 00,-3.00000000E 00,-2.00000000E 00,-1.00000000E 00,
* 0.00000000E-39, 1.00000000E 00, 2.00000000E 00/
 DATA YA4/
* 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.01600000E 00, 1.16300000E 00, 1.20000000E 00, 1.21100000E 00,
* 1.28700000E 00, 1.57700000E 00, 1.91000000E 00, 1.99000000E 00,
* 2.00800000E 00, 2.03200000E 00, 2.08800000E 00, 2.21000000E 00,
* 2.44600000E 00, 2.82600000E 00, 3.28200000E 00, 3.64500000E 00,
* 3.84300000E 00, 3.93200000E 00, 3.96900000E 00, 3.98500000E 00,
* 3.99300000E 00, 3.99600000E 00, 3.99800000E 00, 3.99900000E 00,
* 3.99900000E 00, 4.00000000E 00, 4.00000000E 00, 1.00000000E 00,
* 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00500000E 00,
* 1.08800000E 00, 1.19200000E 00, 1.20300000E 00, 1.22800000E 00,
* 1.33700000E 00, 1.62200000E 00, 1.89800000E 00, 1.98300000E 00,
* 2.00600000E 00, 2.02700000E 00, 2.06700000E 00, 2.14400000E 00,
* 2.28400000E 00, 2.51000000E 00, 2.83200000E 00, 3.20200000E 00,
* 3.52600000E 00, 3.74500000E 00, 3.86700000E 00, 3.93100000E 00,
* 3.96300000E 00, 3.97900000E 00, 3.98800000E 00, 3.99300000E 00,
* 3.99600000E 00, 3.99700000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.00000000E 00, 1.00000000E 00, 1.00200000E 00, 1.03300000E 00,
* 1.14900000E 00, 1.19700000E 00, 1.20800000E 00, 1.24500000E 00,
* 1.35900000E 00, 1.59900000E 00, 1.84900000E 00, 1.96100000E 00/
 DATA AX7/
* 1.99700000E 00, 2.01700000E 00, 2.04400000E 00, 2.09000000E 00,
* 2.16600000E 00, 2.28600000E 00, 2.46200000E 00, 2.70000000E 00,
* 2.98300000E 00, 3.27200000E 00, 3.52000000E 00, 3.70000000E 00,
* 3.81800000E 00, 3.88900000E 00, 3.93200000E 00, 3.95700000E 00,
* 3.97300000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.00000000E 00, 1.00100000E 00, 1.01100000E 00, 1.07200000E 00,
* 1.16700000E 00, 1.19800000E 00, 1.21300000E 00, 1.25200000E 00,
* 1.34800000E 00, 1.52900000E 00, 1.75200000E 00, 1.90400000E 00,
* 1.97100000E 00, 2.00100000E 00, 2.02300000E 00, 2.05000000E 00,
* 2.09000000E 00, 2.14900000E 00, 2.23400000E 00, 2.35100000E 00,
* 2.50500000E 00, 2.69400000E 00, 2.91000000E 00, 3.13500000E 00,
* 3.34700000E 00, 3.52700000E 00, 3.66700000E 00, 3.76900000E 00,
* 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.00000000E 00, 1.00400000E 00, 1.02600000E 00, 1.09200000E 00,
* 1.16500000E 00, 1.19600000E 00, 1.21400000E 00, 1.24800000E 00,
* 1.31600000E 00, 1.43700000E 00, 1.60700000E 00, 1.77800000E 00,
* 1.89600000E 00, 1.95900000E 00, 1.99300000E 00, 2.01800000E 00,
* 2.04200000E 00, 2.07100000E 00, 2.11100000E 00, 2.16300000E 00,
* 2.23200000E 00, 2.31800000E 00, 2.42600000E 00, 2.55300000E 00/
 DATA AX8/
* 2.70000000E 00, 2.86100000E 00, 3.02800000E 00, 1.00000000E 00,
* 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.00100000E 00, 1.00900000E 00, 1.03500000E 00, 1.08900000E 00,
* 1.14900000E 00, 1.18600000E 00, 1.20800000E 00, 1.23500000E 00,
```

```
* 1.27900000E 00, 1.35100000E 00, 1.45700000E 00, 1.59000000E 00,
* 1.72700000E 00, 1.83800000E 00, 1.91400000E 00, 1.96200000E 00,
* 1.99300000E 00, 2.01800000E 00, 2.04200000E 00, 2.06700000E 00,
* 2.09800000E 00, 2.13500000E 00, 2.18000000E 00, 2.23300000E 00,
* 2.29700000E 00, 2.37200000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
* 1.00300000E 00, 1.01200000E 00, 1.03300000E 00, 1.07100000E 00,
* 1.11800000E 00, 1.15900000E 00, 1.18900000E 00, 1.21400000E 00,
* 1.24300000E 00, 1.28400000E 00, 1.34100000E 00, 1.41800000E 00,
* 1.51200000E 00, 1.61600000E 00, 1.71800000E 00, 1.80700000E 00,
* 1.87600000E 00, 1.92700000E 00, 1.96500000E 00, 1.99300000E 00,
* 2.01700000E 00, 2.03900000E 00, 2.06200000E 00, 2.08600000E 00,
* 2.11300000E 00/
 END
```

(3) BLOCK 003

```
      BLOCK DATA
      DIMENSION  XA5(21), YA5(76),AX9(71),YA6(76),AX10(71)
      COMMON /BLK003/ XA5,YA5,AX9,YA6,AX10
      DATA XA5/
     * 0.00000000E-39, 9.00000000E 02, 1.80000000E 03, 2.70000000E 03,
     * 3.60000000E 03, 4.50000000E 03, 5.40000000E 03, 6.30000000E 03,
     * 7.20000000E 03, 8.10000000E 03, 9.00000000E 03, 9.90000000E 03,
     * 1.08000000E 04, 1.17000000E 04, 1.26000000E 04, 1.35000000E 04,
     * 1.44000000E 04, 1.53000000E 04, 1.62000000E 04, 1.71000000E 04,
     * 1.80000000E 04/
      DATA YA5/
     * 7.30000000E-01, 7.42000000E-01, 7.53000000E-01, 7.63000000E-01,
     * 7.72000000E-01, 7.50000000E-01, 7.38000000E-01, 7.34000000E-01,
     * 7.16999990E-01, 6.83000000E-01, 6.83000000E-01, 6.94000000E-01,
     * 7.02000000E-01, 7.09999990E-01, 7.16999990E-01, 7.24000000E-01,
     * 7.31000000E-01, 7.37000000E-01, 7.44000000E-01, 7.49000000E-01,
     * 7.55000000E-01, 7.30000000E-01, 7.42000000E-01, 7.53000000E-01,
     * 7.63000000E-01, 7.72000000E-01, 7.63000000E-01, 7.38999990E-01,
     * 7.34000000E-01, 7.28999990E-01, 7.08000000E-01, 6.82000000E-01,
     * 6.86000000E-01, 6.97000000E-01, 7.05000000E-01, 7.13000000E-01,
     * 7.20999990E-01, 7.27999990E-01, 7.35000000E-01, 7.42000000E-01,
     * 7.48000000E-01, 7.53999990E-01, 7.30000000E-01, 7.42000000E-01,
     * 7.53000000E-01, 7.63000000E-01, 7.72000000E-01, 7.70000000E-01,
     * 7.46999990E-01, 7.38000000E-01, 7.26000000E-01, 7.24999990E-01,
     * 7.04000000E-01, 6.83999990E-01, 6.91000000E-01, 7.02999990E-01,
     * 7.11000000E-01, 7.17999990E-01, 7.24999990E-01, 7.31999990E-01,
     * 7.38999990E-01, 7.46000000E-01, 7.53000000E-01, 7.30000000E-01,
     * 7.42000000E-01, 7.53000000E-01, 7.63000000E-01, 7.72999990E-01,
     * 7.72999990E-01, 7.61999990E-01, 7.44000000E-01, 7.38000000E-01,
     * 7.31999990E-01, 7.23000000E-01, 7.05000000E-01, 6.87000000E-01/
      DATA AX9/
     * 6.89000000E-01, 7.02999990E-01, 7.09999990E-01, 7.20999990E-01,
     * 7.30000000E-01, 7.37000000E-01, 7.44000000E-01, 7.50999990E-01,
     * 7.30000000E-01, 7.42000000E-01, 7.53000000E-01, 7.63000000E-01,
     * 7.72999990E-01, 7.74000000E-01, 7.72999990E-01, 7.56000000E-01,
     * 7.42999990E-01, 7.38999990E-01, 7.31999990E-01, 7.24000000E-01,
     * 7.09000000E-01, 6.93000000E-01, 6.83999990E-01, 6.95999990E-01,
     * 7.09999990E-01, 7.19000000E-01, 7.34000000E-01, 7.37000000E-01,
     * 7.45000000E-01, 7.30000000E-01, 7.42000000E-01, 7.53000000E-01,
     * 7.63000000E-01, 7.72999990E-01, 7.75000000E-01, 7.77000000E-01,
     * 7.72000000E-01, 7.57999990E-01, 7.46999990E-01, 7.38999990E-01,
     * 7.31999990E-01, 7.24000000E-01, 7.15000000E-01, 7.00000000E-01,
     * 6.90000000E-01, 6.90000000E-01, 6.93000000E-01, 7.06999990E-01,
     * 7.13999990E-01, 7.24999990E-01, 7.30000000E-01, 7.42000000E-01,
     * 7.53000000E-01, 7.64000000E-01, 7.74000000E-01, 7.75000000E-01,
     * 7.77000000E-01, 7.75999990E-01, 7.71000000E-01, 7.64000000E-01,
     * 7.53000000E-01, 7.44000000E-01, 7.39999990E-01, 7.27999990E-01,
     * 7.20000000E-01, 7.11000000E-01, 6.98000000E-01, 6.91000000E-01,
     * 6.87000000E-01, 6.87000000E-01, 6.94999990E-01/
      DATA YA6/
     * 1.00000000E 00, 1.10000000E 00, 1.20000000E 00, 1.28000000E 00,
     * 1.33500000E 00, 1.17200000E 00, 1.14700000E 00, 1.14300000E 00,
     * 1.05100000E 00, 7.86000000E-01, 6.09000000E-01, 5.80000000E-01,
     * 5.75000000E-01, 5.69999990E-01, 5.65000000E-01, 5.59999990E-01,
     * 5.55000000E-01, 5.50000000E-01, 5.44999990E-01, 5.40000000E-01,
     * 5.35000000E-01, 1.00000000E 00, 1.10000000E 00, 1.20000000E 00,
     * 1.28000000E 00, 1.34600000E 00, 1.23300000E 00, 1.15700000E 00,
```

```
     * 1.15400000E 00, 1.13000000E 00, 9.98000000E-01, 7.50000000E-01,
     * 6.10000000E-01, 5.77999990E-01, 5.69999990E-01, 5.65000000E-01,
     * 5.59999990E-01, 5.55000000E-01, 5.50000000E-01, 5.44999990E-01,
     * 5.40000000E-01, 5.35000000E-01, 1.00000000E 00, 1.10000000E 00,
     * 1.20000000E 00, 1.28000000E 00, 1.35500000E 00, 1.35500000E 00,
     * 1.21300000E 00, 1.16700000E 00, 1.14400000E 00, 1.11400000E 00,
     * 9.70000000E-01, 7.61999990E-01, 6.28000000E-01, 5.88000000E-01,
     * 5.73999990E-01, 5.59999990E-01, 5.55000000E-01, 5.50000000E-01,
     * 5.44999990E-01, 5.40000000E-01, 5.35000000E-01, 1.00000000E 00,
     * 1.10000000E 00, 1.20000000E 00, 1.28000000E 00, 1.35500000E 00,
     * 1.39100000E 00, 1.32700000E 00, 1.22700000E 00, 1.18000000E 00,
     * 1.16000000E 00, 1.10500000E 00, 9.81999990E-01, 8.08000000E-01/
      DATA AX10/
     * 6.69000000E-01, 6.05000000E-01, 5.87000000E-01, 5.69999990E-01,
     * 5.65000000E-01, 5.59999990E-01, 5.55000000E-01, 5.50000000E-01,
     * 1.00000000E 00, 1.10000000E 00, 1.20000000E 00, 1.28000000E 00,
     * 1.35500000E 00, 1.40400000E 00, 1.40500000E 00, 1.32000000E 00,
     * 1.22500000E 00, 1.19600000E 00, 1.15800000E 00, 1.11500000E 00,
     * 1.01900000E 00, 8.85000000E-01, 7.46000000E-01, 6.52000000E-01,
     * 6.01000000E-01, 5.69999990E-01, 5.66999990E-01, 5.59999990E-01,
     * 5.55000000E-01, 1.00000000E 00, 1.10000000E 00, 1.20000000E 00,
     * 1.28000000E 00, 1.35600000E 00, 1.40700000E 00, 1.43700000E 00,
     * 1.42400000E 00, 1.34700000E 00, 1.26500000E 00, 1.21500000E 00,
     * 1.16700000E 00, 1.13100000E 00, 1.07100000E 00, 9.74999990E-01,
     * 8.60000000E-01, 7.56000000E-01, 6.69000000E-01, 6.17999990E-01,
     * 5.84000000E-01, 5.58000000E-01, 1.00000000E 00, 1.10000000E 00,
     * 1.20000000E 00, 1.28000000E 00, 1.35700000E 00, 1.40800000E 00,
     * 1.44500000E 00, 1.46400000E 00, 1.44900000E 00, 1.40400000E 00,
     * 1.32700000E 00, 1.26800000E 00, 1.23500000E 00, 1.15800000E 00,
     * 1.11800000E 00, 1.05900000E 00, 9.78999990E-01, 8.93000000E-01,
     * 8.05000000E-01, 7.26000000E-01, 6.61999990E-01/
      END
```

(4) BLOCK 004

```
      BLOCK DATA
      DIMENSION YA7(76),AX11(76),AX12(65),XA8(40),YA8(76),AX13(76),
     * AX14(48)
      COMMON /BLK004/ YA7,AX11,AX12,XA8,YA8,AX13,AX14
      DATA YA7/
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.01100000E 00,
     * 1.03200000E 00, 1.09600000E 00, 1.18100000E 00, 1.22700000E 00,
     * 1.25600000E 00, 1.27100000E 00, 1.26400000E 00, 1.20100000E 00,
     * 1.07200000E 00, 8.26000000E-01, 5.17000000E-01, 2.61000000E-01,
     * 1.18000000E-01, 5.50000000E-02, 2.90000000E-02, 1.80000000E-02,
     * 1.20000000E-02, 9.00000000E-03, 7.99999990E-03, 7.00000000E-03,
     * 7.00000000E-03, 7.99999990E-03, 7.99999990E-03, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.01000000E 00, 1.02400000E 00,
     * 1.05500000E 00, 1.12800000E 00, 1.20900000E 00, 1.25700000E 00,
     * 1.28600000E 00, 1.30300000E 00, 1.30700000E 00, 1.28000000E 00,
     * 1.20700000E 00, 1.06800000E 00, 8.52999990E-01, 5.95000000E-01,
     * 3.61000000E-01, 2.00000000E-01, 1.08000000E-01, 6.29999990E-02,
     * 3.60000000E-02, 2.40000000E-02, 1.80000000E-02, 1.50000000E-02,
     * 1.30000000E-02, 1.20000000E-02, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.01000000E 00, 1.02200000E 00, 1.03800000E 00,
     * 1.07400000E 00, 1.14600000E 00, 1.22800000E 00, 1.27600000E 00/
      DATA AX11/
     * 1.31700000E 00, 1.33700000E 00, 1.34700000E 00, 1.34300000E 00,
     * 1.31400000E 00, 1.25100000E 00, 1.14300000E 00, 9.83000000E-01,
     * 7.82000000E-01, 5.70999990E-01, 3.87000000E-01, 2.48000000E-01,
     * 1.58000000E-01, 1.00000000E-01, 6.70000000E-02, 4.20000000E-02,
     * 1.60000000E-02, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00600000E 00, 1.02000000E 00, 1.03300000E 00, 1.05100000E 00,
     * 1.08600000E 00, 1.14800000E 00, 1.22900000E 00, 1.29400000E 00,
     * 1.33200000E 00, 1.37100000E 00, 1.38600000E 00, 1.39600000E 00,
     * 1.39300000E 00, 1.37500000E 00, 1.33500000E 00, 1.26700000E 00,
     * 1.16800000E 00, 1.04000000E 00, 8.81000000E-01, 7.11000000E-01,
     * 5.47000000E-01, 4.08000000E-01, 2.68000000E-01, 2.12000000E-01,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00300000E 00,
     * 1.01600000E 00, 1.02900000E 00, 1.04300000E 00, 1.06000000E 00,
     * 1.09000000E 00, 1.13900000E 00, 1.20800000E 00, 1.28300000E 00,
     * 1.34200000E 00, 1.38600000E 00, 1.42500000E 00, 1.43800000E 00,
     * 1.44500000E 00, 1.44800000E 00, 1.44200000E 00, 1.42400000E 00,
     * 1.39400000E 00, 1.34200000E 00, 1.27400000E 00, 1.18700000E 00/
      DATA AX12/
     * 1.08200000E 00, 9.40000000E-01, 8.27999990E-01, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00100000E 00, 1.00800000E 00,
     * 1.02200000E 00, 1.03600000E 00, 1.05200000E 00, 1.06700000E 00,
     * 1.09000000E 00, 1.12400000E 00, 1.17500000E 00, 1.23800000E 00,
     * 1.30700000E 00, 1.36800000E 00, 1.41800000E 00, 1.46800000E 00,
     * 1.49600000E 00, 1.50100000E 00, 1.51100000E 00, 1.52000000E 00,
     * 1.51600000E 00, 1.50800000E 00, 1.49200000E 00, 1.46800000E 00,
     * 1.41500000E 00, 1.38700000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00000000E 00, 1.00000000E 00,
     * 1.00000000E 00, 1.00000000E 00, 1.00300000E 00, 1.01000000E 00,
     * 1.02200000E 00, 1.03600000E 00, 1.05000000E 00, 1.07200000E 00,
```

```
     * 1.08900000E 00, 1.11200000E 00, 1.14300000E 00, 1.18500000E 00,
     * 1.23800000E 00, 1.29800000E 00, 1.36100000E 00, 1.41800000E 00,
     * 1.46700000E 00, 1.50900000E 00, 1.54900000E 00, 1.57700000E 00,
     * 1.58100000E 00, 1.59400000E 00, 1.59900000E 00, 1.60100000E 00,
     * 1.60400000E 00/
      DATA XA8/
     * 1.80000000E 03, 2.02500000E 03, 2.47500000E 03, 2.70000000E 03,
     * 2.92500000E 03, 3.15000000E 03, 3.37500000E 03, 3.60000000E 03,
     * 3.82500000E 03, 4.05000000E 03, 4.27499990E 03, 4.49999990E 03,
     * 4.72499990E 03, 4.94999990E 03, 5.17499990E 03, 5.39999990E 03,
     * 5.62499990E 03, 5.84999990E 03, 6.07499990E 03, 6.29999990E 03,
     * 6.52499990E 03, 6.74999990E 03, 6.97499990E 03, 7.19999990E 03,
     * 7.42499990E 03, 7.64999990E 03, 7.87499990E 03, 8.09999990E 03,
     * 8.32499990E 03, 8.54999990E 03, 8.77499990E 03, 8.99999990E 03,
     * 9.22499990E 03, 9.44999990E 03, 9.67499990E 03, 9.89999990E 03,
     * 1.01250000E 04, 1.03500000E 04, 1.05750000E 04, 1.08000000E 04/
      DATA YA8/
     * 0.00000000E-39, 0.00000000E-39, 4.99999990E-03, 6.00000000E-03,
     * 7.99999990E-03, 1.60000000E-02, 2.30000000E-02, 5.70000000E-02,
     * 1.25000000E-01, 1.90000000E-01, 2.57000000E-01, 3.22000000E-01,
     * 3.79000000E-01, 4.31000000E-01, 4.63000000E-01, 4.75000000E-01,
     * 4.76000000E-01, 4.73000000E-01, 4.65000000E-01, 4.57000000E-01,
     * 4.55000000E-01, 4.55000000E-01, 4.56000000E-01, 4.60000000E-01,
     * 4.70000000E-01, 4.88000000E-01, 5.14999990E-01, 5.44000000E-01,
     * 5.75000000E-01, 6.03999990E-01, 6.30000000E-01, 6.57000000E-01,
     * 6.75000000E-01, 6.91999990E-01, 7.09000000E-01, 7.20999990E-01,
     * 7.31000000E-01, 7.38999990E-01, 7.45000000E-01, 7.50000000E-01,
     * 0.00000000E-39, 0.00000000E-39, 4.99999990E-03, 6.00000000E-03,
     * 7.00000000E-03, 1.30000000E-02, 1.60000000E-02, 2.60000000E-02,
     * 5.30000000E-02, 9.09999990E-02, 1.33000000E-01, 1.82000000E-01,
     * 2.36000000E-01, 2.92000000E-01, 3.47000000E-01, 4.00000000E-01,
     * 4.25000000E-01, 4.37000000E-01, 4.42000000E-01, 4.43000000E-01,
     * 4.38000000E-01, 4.34000000E-01, 4.29000000E-01, 4.27000000E-01,
     * 4.32000000E-01, 4.37000000E-01, 4.45000000E-01, 4.57000000E-01,
     * 4.70000000E-01, 4.85000000E-01, 5.02000000E-01, 5.18000000E-01,
     * 5.38000000E-01, 5.58000000E-01, 5.77000000E-01, 5.99999990E-01/
      DATA AX13/
     * 6.20000000E-01, 6.42000000E-01, 6.65000000E-01, 6.84999990E-01,
     * 0.00000000E-39, 0.00000000E-39, 5.00000000E-02, 6.00000000E-03,
     * 7.00000000E-03, 1.20000000E-02, 1.30000000E-02, 1.70000000E-02,
     * 2.50000000E-02, 4.00000000E-02, 5.70000000E-02, 8.50000000E-02,
     * 1.26000000E-01, 1.67000000E-01, 2.08000000E-01, 2.50000000E-01,
     * 2.83000000E-01, 3.20000000E-01, 3.52000000E-01, 3.82000000E-01,
     * 3.95000000E-01, 4.02000000E-01, 4.05000000E-01, 4.06000000E-01,
     * 4.05000000E-01, 4.04000000E-01, 4.03000000E-01, 4.02000000E-01,
     * 4.04000000E-01, 4.07000000E-01, 4.15000000E-01, 4.23000000E-01,
     * 4.34000000E-01, 4.49000000E-01, 4.62000000E-01, 4.80000000E-01,
     * 4.97000000E-01, 5.14999990E-01, 5.35000000E-01, 5.55999990E-01,
     * 0.00000000E-39, 0.00000000E-39, 4.99999990E-03, 6.00000000E-03,
     * 7.00000000E-03, 1.10000000E-02, 1.20000000E-02, 1.50000000E-02,
     * 2.00000000E-02, 2.60000000E-02, 3.50000000E-02, 4.50000000E-02,
     * 6.20000000E-02, 8.09999990E-02, 1.04000000E-01, 1.27000000E-01,
     * 1.53000000E-01, 1.78000000E-01, 2.05000000E-01, 2.31000000E-01,
     * 2.57000000E-01, 2.85000000E-01, 3.11000000E-01, 3.37000000E-01,
     * 3.50000000E-01, 3.56000000E-01, 3.59000000E-01, 3.62000000E-01,
     * 3.63000000E-01, 3.64000000E-01, 3.65000000E-01, 3.69000000E-01/
      DATA AX14/
     * 3.75000000E-01, 3.80000000E-01, 3.87000000E-01, 3.94000000E-01,
     * 4.01000000E-01, 4.09000000E-01, 4.17000000E-01, 4.25000000E-01,
     * 0.00000000E-39, 0.00000000E-39, 4.99999990E-03, 6.00000000E-03,
     * 7.00000000E-03, 9.99999990E-03, 1.10000000E-02, 1.20000000E-02,
     * 1.30000000E-02, 1.40000000E-02, 1.50000000E-02, 1.90000000E-02,
```

```
     * 2.60000000E-02, 3.70000000E-02, 5.00000000E-02, 6.99999990E-02,
     * 8.80000000E-02, 1.08000000E-01, 1.26000000E-01, 1.45000000E-01,
     * 1.65000000E-01, 1.83000000E-01, 2.00000000E-01, 2.20000000E-01,
     * 2.35000000E-01, 2.50000000E-01, 2.65000000E-01, 2.77000000E-01,
     * 2.90000000E-01, 2.97000000E-01, 3.05000000E-01, 3.10000000E-01,
     * 3.15000000E-01, 3.18000000E-01, 3.20000000E-01, 3.22000000E-01,
     * 3.25000000E-01, 3.28000000E-01, 3.32000000E-01, 3.38000000E-01/
      END
```

(5) BLOCK 005

```
 BLOCK DATA
 DIMENSION ZA8(5),XA11(47),ZA11(7),YA13(76),AX15(76),AX16(76),
*  AX17(76),AX18(25)
 COMMON /BLK005/ ZA8,XA11,ZA11,YA13,AX15,AX16,AX17,AX18
 DATA ZA8/
*-4.00000000E 00,-3.00000000E 00,-2.00000000E 00,-1.00000000E 00,
*  0.00000000E-39/
 DATA XA11/
*  0.00000000E-39,  3.00000000E 02,  5.00000000 E02,  1.00000000E 03,
*  1.30000000E 03,  1.50000000E 03,  2.00000000 E03,  2.50000000E 03,
*  3.00000000E 03,  3.50000000E 03,  4.00000000 E03,  4.50000000E 03,
*  5.00000000E 03,  6.00000000E 03,  7.00000000 E03,  8.00000000E 03,
*  1.00000000E 04,  1.20000000E 04,  1.40000000 E04,  1.50000000E 04,
*  1.60000000E 04,  1.70000000E 04,  1.80000000 E04,  1.90000000E 04,
*  2.00000000E 04,  2.10000000E 04,  2.20000000 E04,  2.40000000E 04,
*  2.50000000E 04,  2.60000000E 04,  3.00000000 E04,  3.50000000E 04,
*  4.00000000E 04,  5.00000000E 04,  5.50000000 E04,  6.00000000E 04,
*  6.20000000E 04,  6.40000000E 04,  6.50000000 E04,  6.60000000E 04,
*  6.70000000E 04,  6.80000000E 04,  6.90000000 E04,  7.00000000E 04,
*  7.10000000E 04,  7.20000000E 04,  7.35000000 E04/
 DATA ZA11/
*-4.00000000E 00,-3.00000000E 00,-2.00000000E 00,-1.00000000E 00,
*  0.00000000E-39,  1.00000000E 00,  2.00000000E 00/
 DATA YA13/
*  0.00000000E-39,  1.24000000E 03,  2.00000000E 03,  3.44000000E 03,
*  3.86000000E 03,  4.09999900E 03,  4.29999900E 03,  4.54999900E 03,
*  5.29999900E 03,  6.34999900E 03,  6.79999900E 03,  7.09999900E 03,
*  7.25000000E 03,  7.50000000E 03,  7.69999900E 03,  7.89999900E 03,
*  8.19999900E 03,  8.50000000E 03,  8.89999900E 03,  9.25000000E 03,
*  9.89999900E 03,  1.11000000E 04,  1.19500000E 04,  1.25000000E 04,
*  1.28500000E 04,  1.31500000E 04,  1.34000000E 04,  1.38000000E 04,
*  1.40000000E 04,  1.42800000E 04,  1.46000000E 04,  1.50500000E 04,
*  1.55000000E 04,  1.63500000E 04,  1.68000000E 04,  1.74800000E 04,
*  1.78000000E 04,  1.83000000E 04,  1.86500000E 04,  1.91000000E 04,
*  1.97000000E 04,  2.05000000E 04,  2.14500000E 04,  2.26500000E 04,
*  2.39000000E 04,  2.52000000E 04,  2.71000000E 04,  0.00000000E-39,
*  1.24000000E 03,  2.00000000E 03,  3.60000000E 03,  3.90000000E 03,
*  4.27999900E 03,  4.50000000E 03,  4.79999900E 03,  5.50000000E 03,
*  6.59999900E 03,  7.19999900E 03,  7.51999900E 03,  7.79999900E 03,
*  8.14999900E 03,  8.39999900E 03,  8.59999900E 03,  8.94999900E 03,
*  9.29999900E 03,  9.69999900E 03,  1.00000000E 04,  1.05500000E 04,
*  1.16500000E 04,  1.27000000E 04,  1.34000000E 04,  1.39000000E 04,
*  1.42500000E 04,  1.45800000E 04,  1.50500000E 04,  1.52500000E 04/
 DATA AX15/
*  1.56000000E 04,  1.61000000E 04,  1.67000000E 04,  1.72000000E 04,
*  1.82200000E 04,  1.87500000E 04,  1.94000000E 04,  1.98000000E 04,
*  2.03000000E 04,  2.06000000E 04,  2.09000000E 04,  2.13000000E 04,
*  2.18500000E 04,  2.25000000E 04,  2.33000000E 04,  2.43000000E 04,
*  2.54000000E 04,  2.73000000E 04,  0.00000000E-39,  1.24000000E 03,
*  2.00000000E 03,  3.66000000E 03,  4.25000000E 03,  4.50000000E 03,
*  4.94999900E 03,  5.25000000E 03,  5.84999900E 03,  6.79999900E 03,
*  7.59999900E 03,  8.09999900E 03,  8.39999900E 03,  8.84999900E 03,
*  9.14999900E 03,  9.59999900E 03,  9.84999900E 03,  1.02800000E 04,
*  1.07500000E 04,  1.10500000E 04,  1.15000000E 04,  1.23000000E 04,
*  1.33500000E 04,  1.43000000E 04,  1.50000000E 04,  1.55000000E 04,
*  1.59500000E 04,  1.66000000E 04,  1.68500000E 04,  1.73500000E 04,
*  1.79000000E 04,  1.86800000E 04,  1.93000000E 04,  2.05500000E 04,
```

```
* 2.11800000E 04, 2.19800000E 04, 2.23500000E 04, 2.28000000E 04,
* 2.30800000E 04, 2.33500000E 04, 2.37000000E 04, 2.40500000E 04,
* 2.44500000E 04, 2.50000000E 04, 2.56500000E 04, 2.64000000E 04,
* 2.76500000E 04, 0.00000000E-39, 1.24000000E 03, 2.00000000E 03,
* 3.70000000E 03, 4.50000000E 03, 4.79999900E 03, 5.34999900E 03,
* 5.67999900E 03, 6.19999900E 03, 7.17999900E 03, 8.00000000E 03/
 DATA AX16/
* 8.59999900E 03, 9.04999900E 03, 9.59999900E 03, 1.00200000E 04,
* 1.03500000E 04, 1.09000000E 04, 1.14000000E 04, 1.19500000E 04,
* 1.22500000E 04, 1.26500000E 04, 1.33000000E 04, 1.42000000E 04,
* 1.52500000E 04, 1.61000000E 04, 1.68000000E 04, 1.74000000E 04,
* 1.83000000E 04, 1.86800000E 04, 1.93000000E 04, 2.00500000E 04,
* 2.11000000E 04, 2.19000000E 04, 2.34500000E 04, 2.42500000E 04,
* 2.51500000E 04, 2.55500000E 04, 2.60000000E 04, 2.62500000E 04,
* 2.65000000E 04, 2.68000000E 04, 2.71000000E 04, 2.71000000E 04,
* 2.71000000E 04, 2.71000000E 04, 2.71000000E 04, 2.71000000E 04,
* 0.00000000E-39, 1.24000000E 03, 2.00000000E 03, 3.70000000E 03,
* 4.64999900E 03, 5.00000000E 03, 5.79999900E 03, 6.39999900E 03,
* 6.77999900E 03, 7.44999900E 03, 8.34999900E 03, 9.19999900E 03,
* 9.75000000E 03, 1.00500000E 04, 1.11000000E 04, 1.15500000E 04,
* 1.22000000E 04, 1.28000000E 04, 1.34000000E 04, 1.37200000E 04,
* 1.41500000E 04, 1.47000000E 04, 1.53500000E 04, 1.62000000E 04,
* 1.72500000E 04, 1.82000000E 04, 1.90000000E 04, 2.02200000E 04,
* 2.07200000E 04, 2.16000000E 04, 2.26000000E 04, 2.40000000E 04,
* 2.51500000E 04, 2.71200000E 04, 2.71200000E 04, 2.71200000E 04,
* 2.71200000E 04, 2.71200000E 04, 2.71200000E 04, 2.71200000E 04/
 DATA AX17/
* 2.71200000E 04, 2.71200000E 04, 2.71200000E 04, 2.71200000E 04,
* 2.71200000E 04, 2.71200000E 04, 2.71200000E 04, 0.00000000E-39,
* 1.24000000E 03, 2.00000000E 03, 3.70000000E 03, 4.79999900E 03,
* 5.25000000E 03, 6.19999900E 03, 6.89999900E 03, 7.44999900E 03,
* 8.09999900E 03, 8.84999900E 03, 8.75000000E 03, 1.04500000E 04,
* 1.15800000E 04, 1.22800000E 04, 1.28200000E 04, 1.37000000E 04,
* 1.45000000E 04, 1.53000000E 04, 1.56000000E 04, 1.59500000E 04,
* 1.64000000E 04, 1.70000000E 04, 1.77500000E 04, 1.86500000E 04,
* 1.96000000E 04, 2.06000000E 04, 2.22500000E 04, 2.29500000E 04,
* 2.41800000E 04, 2.56500000E 04, 2.77500000E 04, 2.77500000E 04,
* 2.77500000E 04, 2.77500000E 04, 2.77500000E 04, 2.77500000E 04,
* 2.77500000E 04, 2.77500000E 04, 2.77500000E 04, 2.77500000E 04,
* 2.77500000E 04, 2.77500000E 04, 2.77500000E 04, 2.77500000E 04,
* 2.77500000E 04, 2.77500000E 04, 0.00000000E-39, 1.24000000E 03,
* 2.00000000E 03, 3.70000000E 03, 4.89999900E 03, 5.44999900E 03,
* 6.59999900E 03, 7.50000000E 03, 8.25000000E 03, 8.94999900E 03,
* 9.69999900E 03, 1.04500000E 04, 1.13000000E 04, 1.26000000E 04,
* 1.36000000E 04, 1.43500000E 04, 1.55500000E 04, 1.65000000E 04,
* 1.74000000E 04, 1.78000000E 04, 1.83000000E 04, 1.88000000E 04/
 DATA AX18/
* 1.94000000E 04, 2.00000000E 04, 2.07000000E 04, 2.15500000E 04,
* 2.24500000E 04, 2.44000000E 04, 2.54000000E 04, 2.72000000E 04,
* 2.72000000E 04, 2.72000000E 04, 2.72000000E 04, 2.72000000E 04,
* 2.72000000E 04, 2.72000000E 04, 2.72000000E 04, 2.72000000E 04,
* 2.72000000E 04, 2.72000000E 04, 2.72000000E 04, 2.72000000E 04,
* 2.72000000E 04, 2.72000000E 04, 2.72000000E 04, 2.72000000E 04,
* 2.72000000E 04/
   END
```

6. SUBROUTINE DISCT

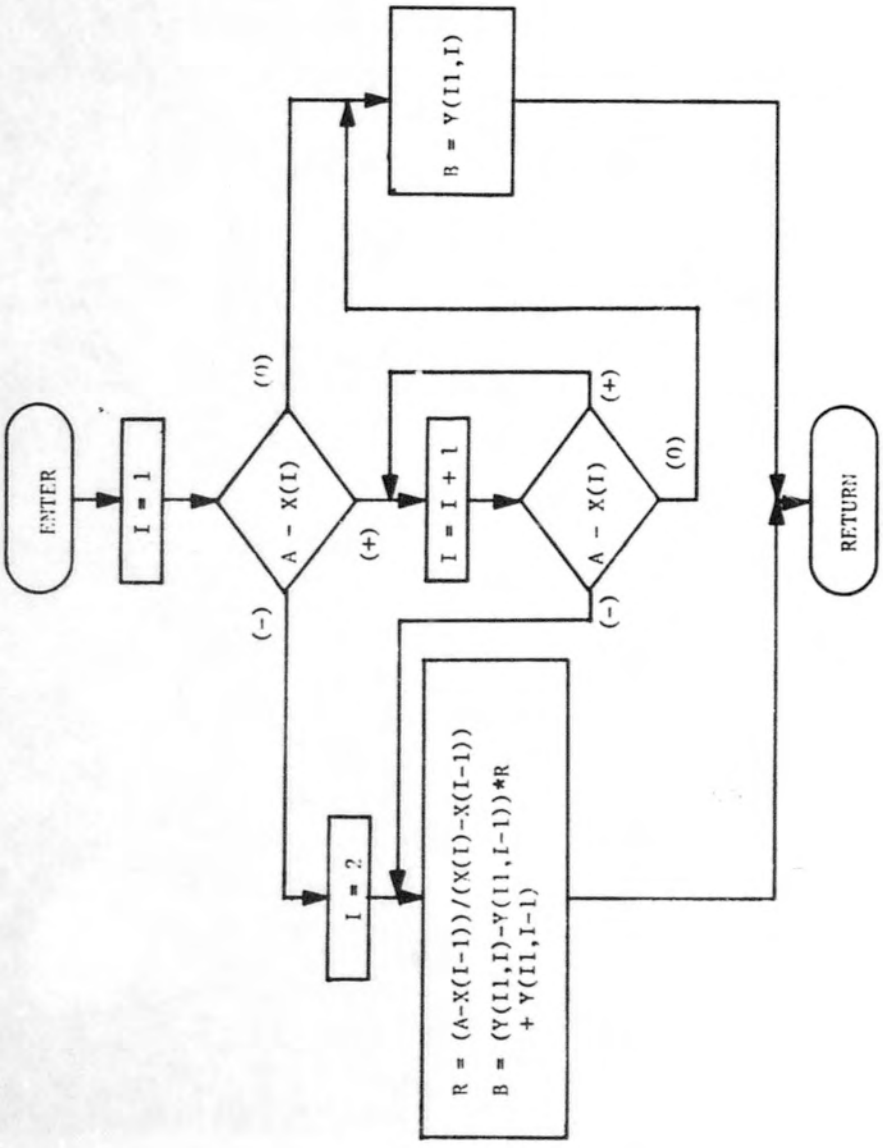
a. Function

Subroutine DISCT performs a linear interpolation in one plane of a two-dimensional array. If Y is a function of Z and X, interpolation is performed for intermediate values of X while Z is held constant. This routine is used to interpolate in the chemistry property tables where the tables are stored as a two-dimensional array. In this case, Z represents the particular specie identification number and X represents the temperatures for which the enthalpy or free energy, Y, are tabulated.

b. Symbols

Symbolic representation	Meaning or use
A	Value of independent variable in iteration
B	Value of dependent variable after iteration
I1	Index of X values in Y array
I2	Index of Z values in Y array
X	Linear array of independent variables
Y	Two-dimensional array of dependent variables

c. Flow Chart


ENTER
I = 1
A - X(I)
(0)
(+)
(-)
B = Y(I1,I)
I = I + 1
A - X(I)
(+)
(0)
(-)
I = 2
R = (A-X(I-1))/(X(I)-X(I-1))
B = (Y(I1,I)-Y(I1,I-1))*R
+ Y(I1,I-1)
RETURN

D. LISTING

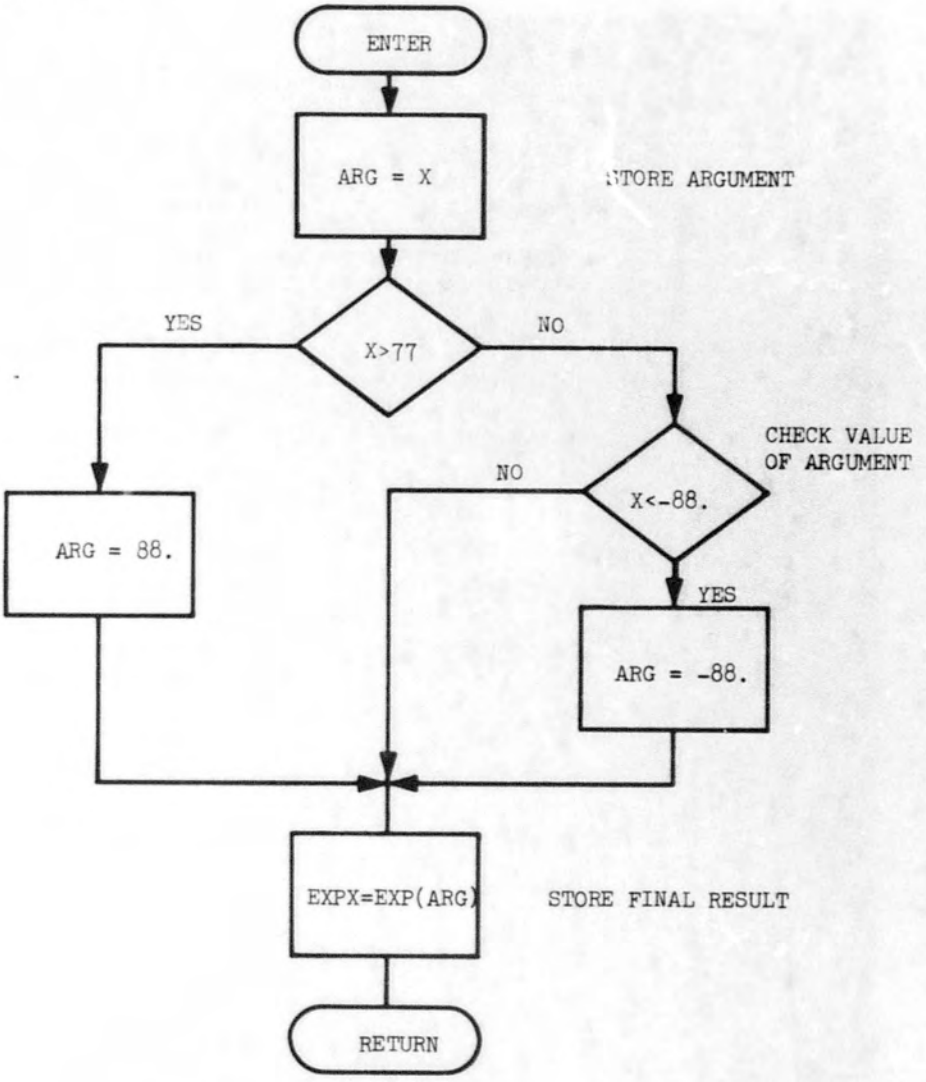
```
      SUBROUTINE DISCT (A,X,Y,I1,I2,B)
C
C***** FOR SEGMENT NINE, DIMENSION Y AS (17,39)
C*****FOR SEGMENT SIX, DIMENSION Y AS (70,39)
C
      DIMENSION X(25),Y(25,39)
      I=1
      IF(A-X(1))200,150,5
    5 DO 100 I=2,I2
      IF(A-X(I))10,150,100
  100 CONTINUE
      I=I2
      GO TO 150
  150 B=Y(I1,I)
      RETURN
  200 I=2
   10 R = (A-X(I-1))/(X(I)-X(I-1))
      B = (Y(I1,I)-Y(I1,I-1))*R + Y(I1,I-1)
      RETURN
      END
```

7. SUBROUTINE EXPX

a. Function

This subroutine is used to simulate the exponential routine of the IBM-7094 computer. The argument for the exponentiation process is limited to ± 88 to prevent an arithmetic error being generated by the machine (the CDC 6600). The chemical calculation procedure, which uses this routine, will generate large (small) arguments if the initial guesses for the iteration procedure are bad. The arguments are automatically restored to acceptable limits if a bad value is generated. The chemistry calculations will be forced toward convergence by this procedure if the computer is not permitted to overflow this limit on the argument.

b. Flow Chart


ENTER
ARG = X
STORE ARGUMENT
YES
X>77
NO
CHECK VALUE
OF ARGUMENT
NO
X<-88.
ARG = 88.
YES
ARG = -88.
EXPX=EXP(ARG)
STORE FINAL RESULT
RETURN

C. LISTING

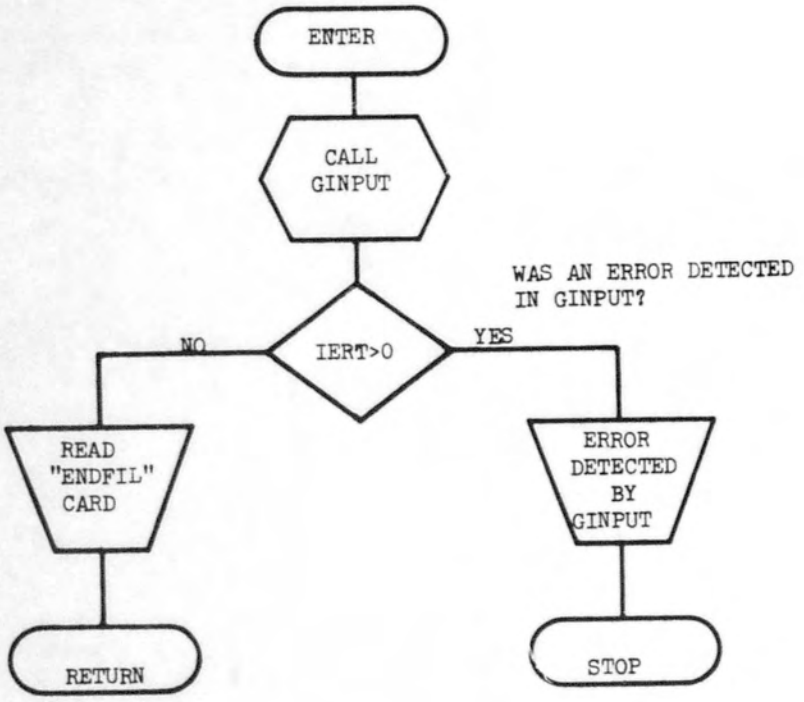
```
      FUNCTION EXPX(I,X)
C
C****          SUBPROGRAM TO SIMULATE THE IBM-7094 EXPONENTIAL ROUTINE.
C
C****          I    PROGRAM LOCATION FLAG.  USED TO ISOLATE WHERE AN
C****               EXPONENTIAL ERROR OCCURRED.
C
C****          X    THE ARGUMENT.
C
C****          SAMPLE CALLING SEQUENCE.    Y = EXPX(I,X).
C
      ARG =  X
      IF(X-88.) 1,1,2
    2 CONTINUE
      ARG = 88.
      GO TO 3
1     IF(X+88.) 4,3,3
    4 CONTINUE
      ARG= -88.
3     EXPX = EXP(ARG)
      RETURN
      END
```

8. SUBROUTINE DIRECT

a. Function

This subroutine is a short routine that performs no calculations. The subroutine calls subroutine GINPUT which reads the symbolic input data. If no errors are detected by GINPUT, DIRECT reads the "ENDFIL" card remaining in the symbolic data, then returns control of the program to Segment TWO. However, if an error is detected by GINPUT, subroutine DIRECT will terminate program execution via a STOP statement. A statement explaining the stop is written on the output file.

b. Flow Chart


ENTER
CALL GINPUT
WAS AN ERROR DETECTED IN GINPUT?
IERT>0
NO
YES
READ "ENDFIL" CARD
ERROR DETECTED BY GINPUT
RETURN
STOP

C. LISTING

```
      SUBROUTINE DIRECT(ARRAY,BCD)
      DIMENSION  ARRAY(1),BCD(1)
      COMMON /456 / ITAPE,JTAPE,INT,NMN,M,NGR
      CALL GINPUT(ARRAY,ARRAY,BCD,ITAPE,JTAPE,IERT)
      IF(IERT.GT.0) GO TO 11
      READ(ITAPE,50) NTYPE
   50 FORMAT(A6)
   60 RETURN
   11 PRINT 1
    1 FORMAT(*   ERROR DETECTED BY GINPUT, EXECUTION TERMINATED BY STOP
     **)
      STOP
      END
```

9. SUBROUTINE CHEMS

a. Function

This subroutine is a small version of the chemical equilibrium composition program used in Segment SIX. While the basic equilibrium calculation procedure is unchanged, the number of species considered, data input, and calling sequence are different from Segment SIX. These differences are minor; hence only a listing of the routine is provided and the flow chart is omitted.

B. LISTING

```
      SUBROUTINE CHEMS (TTEMP,JTR,VRTO,KIP,HCALC,LIB,T2,JP)
C      T.E. HORTON  SHOCK AND CHEMICAL EQUILIBRIUM PROGRAM                    00000020
C        THIS SUBROUTINE IS A SCALED DOWN VERSION OF THE CHEMICAL
C        COMPOSITION PROGRAM. ITERATION IS ON BOTH TEMPERATURE AND
C        DENSITY WITH DENSITY AS OUTPUT.
C
      COMMON /234/ C(   39),   CON(   25),CONOLD(   25),  CONR(   15),
     1            CPP(   25),     D(   25), DELDH(    2),   DSO(   25),
     2           EFR(25,39),    EK(25,39),   ENT(25,39),  FORM(   25),
     3            FR(   15),   IID(   15),    NA(   10),   PHI(   25),
     4            SB(   25), SFCON(   25),  STSB(   10), TTEMP(    4),
     5           WTM(   25),   WTR(   25),     Y(25,25),     Z(   25)
      COMMON /666/ MAXE,MAXS,JT,TE,STY
      COMMON /222/DY(101),FS,RHOOW(101),TRS(101),CFS(101),ELS(101),HZ,
     *HAW,HE,RHOW,TITL(5),C5,C6,UE,B,CIR
C
      DIMENSION STY(25,10), TE(39),MAXE(1),MAXS(1),JT(1)
C
C
C
C
C
C
      EQUIVALENCE(MAXEX,MAXE(1)),(MAXSX,MAXS(1)),(JTX,JT(1))
  501 FORMAT(3I2)
 4804 FORMAT(15H THE VALUE JTR=,I4,2X,12HIS NOT VALID)                        00000130
 6600 FORMAT(4X,1E12.6)                                                       00000160
 7002 FORMAT (I2,A4,E12.6)
 7004 FORMAT (12XA4,10I2,10X,F10.6)
 7008 FORMAT(20X,5F7.1)                                                       00000220
 7009 FORMAT(20X,5E12.6)                                                      00000230
 7110 FORMAT(20I4)                                                            00000250
 9601 FORMAT(3X,A4,2X,2E13.6)                                                 00000430
 9690 FORMAT(3X,A4,2X,5E14.6)                                                 00000480
10009 FORMAT (I3,5E12.6)
      INTAP = 5
      IOTAP = 6
      KIT = 1
      IF(LIB.NE.1) GO TO 6000
      LIB = 0
    8 READ (INTAP,501) MAXE,MAXS
 7155 READ (INTAP,501) JT
      READ (INTAP,7008) (TE(I),I=1,JTX)
      JFIVEL =0
      DO 7013 I=1,MAXSX                                                       00000660
      READ (INTAP,7004) FORM(I),NA,WTM(I)
      DO 7005 J=1,MAXEX                                                       00000690
 7005 STY(I,J)=NA(J)                                                          00000700
      READ (INTAP,7009) ( EK(I,J),J=1,JTX)
      READ (INTAP,7009)( ENT(I,J),J=1,JTX)
 7013 READ (INTAP,7009) EFR(I,1)
      WRITE (6,3033) (CONR(I),I=1,15), (IID(I),I=1,15)
 3033 FORMAT (1H110X6HCONR =5E16.7/, 2(16X5E16.7/),
     1 1H010X6HIID  =5I8/,2(17X5I8/))
 6000 M = MAXE
      L = MAXS
      ICOUNT = 4
```

```
      NUM=L-M                                                          00000790
      MIN=M                                                            00000800
      LARG=L                                                           00000810
      NUP=L-M+1                                                        00000820
      ITER=0                                                           00000870
      IF(LIB.GT.1)  GO TO 6001                                         00000890
      DO 6601 I=NUP,LARG
 6601 READ(5,6600) CON(I)
 6001 CONTINUE
C
 6003 CONTINUE
 6006 IST = 0
 6400 SCON=(1.0)/(0.0820575*273.15)                                    00000930
      DO 7112 I=1,MAXEX                                                00000940
 7112 STSB(I)=0.0                                                      00000950
      DO 7114 I=1,KIP                                                  00000960
      K=IID(I)                                                         00000970
      DSST=CONR(I)*SCON                                                00000980
      DO 7114 J=1,MAXEX                                                00000990
 7114 STSB(J)=STSB(J)+(DSST*STY(K,J))                                  00001000
      N=1                                                              00001010
      K=NUP                                                            00001020
      DO 6405 M=1,MAXEX                                                00001030
      LOPP=MAXS-MAXE+M                                                 00001040
      IF(N+NUM-LOPP)6405,6403,6405
 6403 I=M                                                              00001060
      DO 6404 L=1,LARG                                                 00001070
      J=L
 6404 Y(L,K)=STY(J,I)                                                  00001090
      SB(K)=STSB(M)                                                    00001100
      N=N+1                                                            00001110
      K=K+1                                                            00001120
 6405 CONTINUE                                                         00001150
 7602 DS0=0.0                                                          00001380
      DO6407 I=NUP,LARG                                                00001390
 6407 DS0=DS0+(SB(I)*WTM(I))                                           00001400
      WWTSTP=DS0*273.15*0.0820575                                      00001410
      DSAIR=(28.966348)/(273.15*0.0820575)                             00001420
      GO TO(9000,9070),KIT                                             00001430
 9070 IF (IVY) 4808,9000,4808                                          00001440
 4808 VOL=DS0/(DS1*VOLBA)                                              00001450
 9000 IF (ICOUNT-3) 10005,10005,10200
10000 CONTINUE
10200 CONTINUE
10300 LCOUNT = 1
      IF(JTR)10001,10001,10002
10001 TEMP = TTEMP(1)
      NTRIG=1
      ICOUNT = 1
      GO TO 202
10002 IF (JTR - 100) 262,6000,4801
 5001 WRITE (IOTAP,4804) JTR
      CALL EXIT
  262 IF (JTR-90) 263,265,5001
  263 IF (JTR-80) 5001,200,5001
  200 NTRIG = 3
      DELRO = .0001 * VRTO
      GO TO 201
  265 NTRIG = 2
  201 TEMP=TTEMP(1)
      DELT = 0.1 * TTEMP(1)
       HZERO=TTEMP(3)
```

```
      KNTRL=1
  202 PORT = TTEMP(2)/TEMP/0.0820575
      JITER = 0
      GO TO 6055
10005 IF(TTEMP(ICOUNT+1))10000,10000,10006
10006 ICOUNT = ICOUNT+1
      TEMP = TTEMP(ICOUNT)
      GO TO 6055
 4801 IF(IST) 4806,4805,4806                                          00001500
 4805 IVY=0                                                           00001510
      GO TO  4807                                                     00001520
 4806 VOLBA=DSO/(DS1*VOL)                                             00001530
      IVY=1                                                           00001540
 4807 IF (JTR-101)5001,4802,5001                                      00001560
 4802 GO TO 6001                                                      00001570
 6055 IF (IST)5001,5500,5001                                          00001610
 5500 AIR=28.966348                                                   00001620
C        SET INITIAL VALUES                                          00001750
 5511 JAC = 0                                                         00001760
      DO 7012 I=1,LARG                                                00001770
      PHI(I)=0.0                                                      00001780
      D(I)=0.0                                                        00001790
C        CALCULATION OF SPECIFIC ENTHALPIES AND EQUILIBRIUM CONSTANTS 00001800
      CALL DISCT (TEMP,TE,ENT,I,39,QQQ)
7012  CPP(I) = QQQ*TEMP + 1000.*EFR(I,1)
      DO 7007 I=1,NUM                                                 00001820
      CALL DISCT (TEMP,TE,EK,I,39,QQQ)
      PHI(I) = -(QQQ-EFR(I,1)*1000./TEMP)/1.98726
      DO 7007 K=NUP,LARG                                              00001840
      CALL DISCT (TEMP,TE,EK,K,39,QQQ)
      BBR = (QQQ-EFR(K,1)*1000./TEMP)/1.98726
 7007 PHI(I)=PHI(I) +Y(I,K)*BBR                                       00001860
C        TEST FOR FIRST RUN KIT=1 FOR FIRST RUN,KIT=2 FOR ALL OTHERS 00001870
      GO TO(7901,9200),KIT                                            00001880
C        CALCULATION OF INITIAL GUESS FOR FIRST RUN                  00001890
 7901 TF=ALOG(TEMP*0.0820575)                                         00001900
      VRTO = PORT * WWTSTP / DSAIR
      KIT=2                                                           00001910
      DO 5777 K=1,NUM                                                 00001920
      D(K) = -(PHI(K)+TF)
      DO 5797 I=NUP,LARG                                              00001940
 5797 D(K) = D(K) + ((TF+ALOG(CON(I)))*Y(K,I))
5777  CON(K)=EXPX(5777,D(K))
C        BEGINNING OF EQUILIBRIUM ITERATION LOOP                     00001980
 9200 JAC=JAC+1                                                       00001990
      VOL = WWTSTP / (AIR*VRTO)
C        TEST FOR NUMBER OF ITERATIONS                               00002000
      IF(JAC-150)5003,5003,5991
C        EQUILIBRIUM BLOCK NOT CONVERGENT BLOCK                      00002020
 5002 FORMAT(48H EQUILIBRIUM STATE NOT REACHED IN 150 ITERATIONS)
 5991 WRITE (IOTAP,5002)
      DO 5004 I=1,LARG                                                00002050
      WRITE (IOTAP,9601) FORM(I),CON(I),D(I)
 5004 WRITE (IOTAP,9601) FORM(I),CONOLD(I),DSO(I)
      LIM = 5
      GO TO 299
C        EQUILIBRIUM BLOCK                                           00002090
 5003 COLD=CS2                                                        00002100
      HOLD=H2                                                         00002110
      DOLD=DS2                                                        00002120
      DO 9514 I=1,LARG                                                00002130
      DSO(I)=D(I)                                                     00002140
```

```
 9514 CONOLD(I)=CON(I)                                                  00002150
      DO 9203 I=1,MIN                                                   00002160
      DO 9203 J=1,MIN                                                   00002170
      Y(I,J)=0.0                                                        00002180
      M=NUM+I                                                           00002190
      N=NUM+J                                                           00002200
      DO 9202 K=1,NUM                                                   00002210
 9202 Y(I,J)=Y(I,J)+Y(K,M)*Y(K,N)*CON(K)                                00002220
      IF(I-J) 9203,9201,9203                                            00002230
 9201 Y(I,J)=Y(I,J)+CON(M)                                              00002240
 9203 CONTINUE                                                          00002250
      TF=ALOG(TEMP*0.0820575)                                           00002260
      DO 9205 K=1,NUM                                                   00002270
      IF(CON(K)) 9313,9312,9313                                         00002280
 9312 ZIGG=.0                                                           00002290
      GO TO 9215                                                        00002300
 9313 ZIGG=ALOG(CON(K))                                                 00002310
 9215 D(K)=PHI(K)+TF+ZIGG                                               00002320
      DO 9205 I=NUP,LARG                                                00002330
      IF(CON(I)) 9311,9310,9311                                         00002340
 9310 ZAGG=0.0                                                          00002350
      GO TO 9205                                                        00002360
 9311 ZAGG=ALOG(CON(I))                                                 00002370
 9205 D(K)=D(K)-((TF+  ZAGG       )*Y(K,I))                            00002380
      DO 9206 I=NUP,LARG                                                00002390
      D(I)=CON(I)-(SB(I)/VOL)                                           00002400
      DO 9206 K=1,NUM                                                   00002410
 9206 D(I)=D(I)+(CON(K)*Y(K,I))                                         00002420
      DO 9207 I=1,MIN                                                   00002430
      K=NUM+I                                                           00002440
      C(I)=D(K)                                                         00002450
      DO 9207 J=1,NUM                                                   00002460
 9207 C(I)=C(I)-D(J)*CON(J)*Y(J,K)                                      00002470
      DO 9301 K=2,MIN                                                   00002480
 9301 Y(1,K)=Y(1,K)/Y(1,1)                                              00002490
      C(1)=C(1)/Y(1,1)                                                  00002500
      DO 9306 L=2,MIN                                                   00002510
      N=L-1                                                             00002520
      DO 9302 J=L,MIN                                                   00002530
      DO 9302 K=1,N                                                     00002540
 9302 Y(J,L)=Y(J,L)-Y(J,K)*Y(K,L)                                       00002550
      LA=L+1                                                            00002560
      DO 9304 I=LA,MIN                                                  00002570
      DO 9303 K=1,N                                                     00002580
      Y(L,I) = Y(L,I) - (Y(L,K)*Y(K,I))
 9303 CONTINUE
C
      Y(L,I) = Y(L,I)/Y(L,L)
 9304 CONTINUE
      DO 9305 K=1,N                                                     00002610
 9305 C(L)=C(L)-(Y(L,K)*C(K))                                           00002620
 9306 C(L)=C(L)/Y(L,L)                                                  00002630
      M=MIN                                                             00002640
      Z(M)=-C(M)                                                        00002650
      DO 9307 I=2,MIN                                                   00002660
      N=MIN+1-I                                                         00002670
      NET=N+1                                                           00002680
      Z(N)=-C(N)                                                        00002690
      DO 9307 K=NET,MIN                                                 00002700
 9307 Z(N)=Z(N)-Y(N,K)*Z(K)                                             00002710
      DO 9501 I=1,MIN                                                   00002720
      K=LARG-MIN+I                                                      00002730
```

```
 9501 Z(K)=Z(I)                                                          00002740
      DO 9503 I=1,NUM                                                    00002750
      Z(I)=-D(I)                                                         00002760
      DO 9503 J=NUP,LARG                                                 00002770
 9503 Z(I)=Z(I)+Y(I,J)*Z(J)                                              00002780
      DO 9602 I=1,LARG                                                   00002790
      CON(I)=EXPX(9503,Z(I))*CON(I)
      IF(CON(I))9602,9506,9507                                           00002810
 9506 CON(I) = 1.000E-35                                                 00002812
      GO TO 9602                                                         00002814
 9507 IF(CON(I) - .1)9602,9602,9508                                      00002816
 9508 CON(I) = .1                                                        00002818
 9602 CONTINUE                                                           00002830
C        CALCULATION OF THERMODYNAMIC PROPERTIES                        00002840
      CS2=0.0                                                            00002850
      H2=0.0                                                             00002860
      DS2=0.0                                                            00002870
      DO 9072 I=1,LARG                                                   00002880
      CS2=CS2+CON(I)                                                     00002890
      H2=H2+(CPP(I)*CON(I))                                              00002900
 9072 DS2=DS2+(WTM(I)*CON(I))                                            00002910
      VRTO = PORT * DS2 / (CS2*DSAIR)
C        TEST FOR CONVERGENCE OF EQUILIBRIUM PROPERTIES                 00002920
 4066 TES1=1.0-(COLD/CS2)                                                00002930
      ZT1=ABS(TES1)                                                      00002940
      IF(ZT1-0.000001) 9901, 9901, 9211                                  00002950
 9901 TES2=1.0-(DOLD/DS2)                                                00002960
      ZT2=ABS(TES2)                                                      00002970
      IF(ZT2-0.000001) 9902, 9902, 9211                                  00002980
 9902 TES3=1.0-(HOLD/H2)                                                 00002990
      ZT3=ABS(TES3)                                                      00003000
      IF(ZT3-0.000001) 9903, 9903, 9211                                  00003010
 9211 IF(JAC-4) 9221, 9221, 9200                                         00003020
 9221 DO 9220 I=1,LARG                                                   00003030
 9220 CON(I)=(CONOLD(I)+CON(I))/2.0                                      00003040
      GO TO 9200                                                         00003050
 9903 IF (JITER) 9904,9904,9905
 9904 JITER = 1
      GO TO 9211
 9905 CONTINUE
      H2 = H2/DS2
      HCALC = H2 * 1.8
      T2=TEMP                                                            00003070
      P2=(1.98726*T2*CS2)                                                00003080
      PRES2=(T2*CS2*0.0820575)                                           00003090
      GO TO (299,500,299), NTRIG
C
  300 LCOUNT =LCOUNT +1
      IF (LCOUNT-150) 202,202,301
  301 WRITE (IOTAP,302)
      LIB = 5
      JFIVEL = JFIVEL + 1
      IF (JFIVEL .GT. 3) STOP
  302 FORMAT(41H CASE DID NOT CONVERGE IN 150 ITERATIONS.)
      GO TO 299
  500 GO TO (520,530), KNTRL
C
  520 KNTRL = 2
      H12 = HCALC
      TA = TEMP
      TEMP = TEMP + DELT
      IF (TEMP)1263,1263,300
```

```
 1263 TEMP = 0.9 * TA
      DELT=TEMP-TA
      GO TO 300
  530 DELH = (HCALC-H12)/ DELT
      DELT = (HZERO-HCALC) /DELH
      TTEST = 1.0 - (HZERO/HCALC)
      IF (ABS(TTEST) - 0.000001) 299,299,520
  299 VRTO = VRTO * 1.29233 / 16.018432
      IF (JP.EQ.0) RETURN
      ZSA = (PRES2*273.15*VOL)/T2
      ENTHA = H2/(1.98726*273.15) * WWTSTP
      ENTRO = - ALOG(PRES2)
      DO 9695  I = 1,LARG
      AFR = CON(I)/CS2
      IF (AFR) 9694,9696,9694
 9696 TOM = 0.0
      GO TO 19695
 9694 TOM = ALOG(AFR)
19695 CALL DISCT(TTEMP,TE,ENT,I,39,QQQ)
      CALL DISCT(TTEMP,TE,EK ,I,39,QQ1)
 9695 ENTRO = ENTRO + AFR*((QQQ+QQ1)/1.98726 - TOM)
      ENTRO = ENTRO * (CS2/DS2) * WWTSTP
      WRITE (6,9620) TITL,DSAIR,JAC
      DO 9095 I= 1,LARG
      CONFR = CON(I) / CS2
      CPMOL = CON(I) *.6023382E+21
 9095 WRITE (6,9690) FORM(I),CONOLD(I),CON(I),CPMOL,CONFR,PHI(I)
      VPP = VRTO * 16.018432 / 1.29233
      CF = 5.814E12 * PRES2 /SQRT(T2)
      ENTHAL = 2.93641 * ENTHA * VPP / WWTSTP
      RH = VPP / RHOW * 1.29233 / 16.018432
      WRITE (6,9604) T2,VPP,PRES2,ZSA,CS2,ENTHA,ENTRO,WWTSTP,CF,ENTHAL,
     * HCALC, RH
      DO 9699 I=1,KIP
      ABCE = CONR(I) * 100.0
 9699 WRITE (6,9693) FR(I), ABCE
 9604 FORMAT(20H   FINAL CONDITIONS /                                    00000450
     *       7X, 70H TEMP DEG-K       RHO/RHOO        P/PO          Z    00000520
     1    SUM CON MOL/L /                                               00000530
     *       6X,5E14.6////                                              00000550
     *       10X,19H HMO/RTO       SMO/R/                               00000540
     *       6X,2E14.6////                                              00000560
     *       26H   COLD MIXTURE MOLE WT. =F10.6////                     00000570
     *       26H     COLLISION FREQUENCY =1PE13.6////
     *       16X10HENTHALPY =1PE13.6,24HJOULES / CC      HCALC =E13.6//
     *16X,9HDENSITY =E14.6)
 9620 FORMAT (1H130X24HTURBULENT BOUNDARY LAYER/1H030X5A10/
     * 1H09X70H CHEMICAL EQUILIBRIUM COMPOSITION AND PROPERTIES OF A COM
     *PLEX MIXTURE //
     *       9H   RHOO =1E14.6,5H GM/L 7X,22HNUMBER OF ITERATIONS =1I2/ 00000420
     *       10X,70HCONCENTRATION - MOLE/LITER  CONCENTRATION  MOLE FRAC00000490
     1TION  -LOGE(KP)      /                                            00000500
     *       11X,40H(FINAL-1)        (FINAL)    PARTICLES/CC     )     00000510
 9693 FORMAT(3X,23HTOTAL CONCENTRATION OF A4,23HAT STANDARD CONDITIONS  00000580
     1  E13.5, 12H PER CENT    )
      RETURN
      END
```

10. SUBROUTINE DDUMP

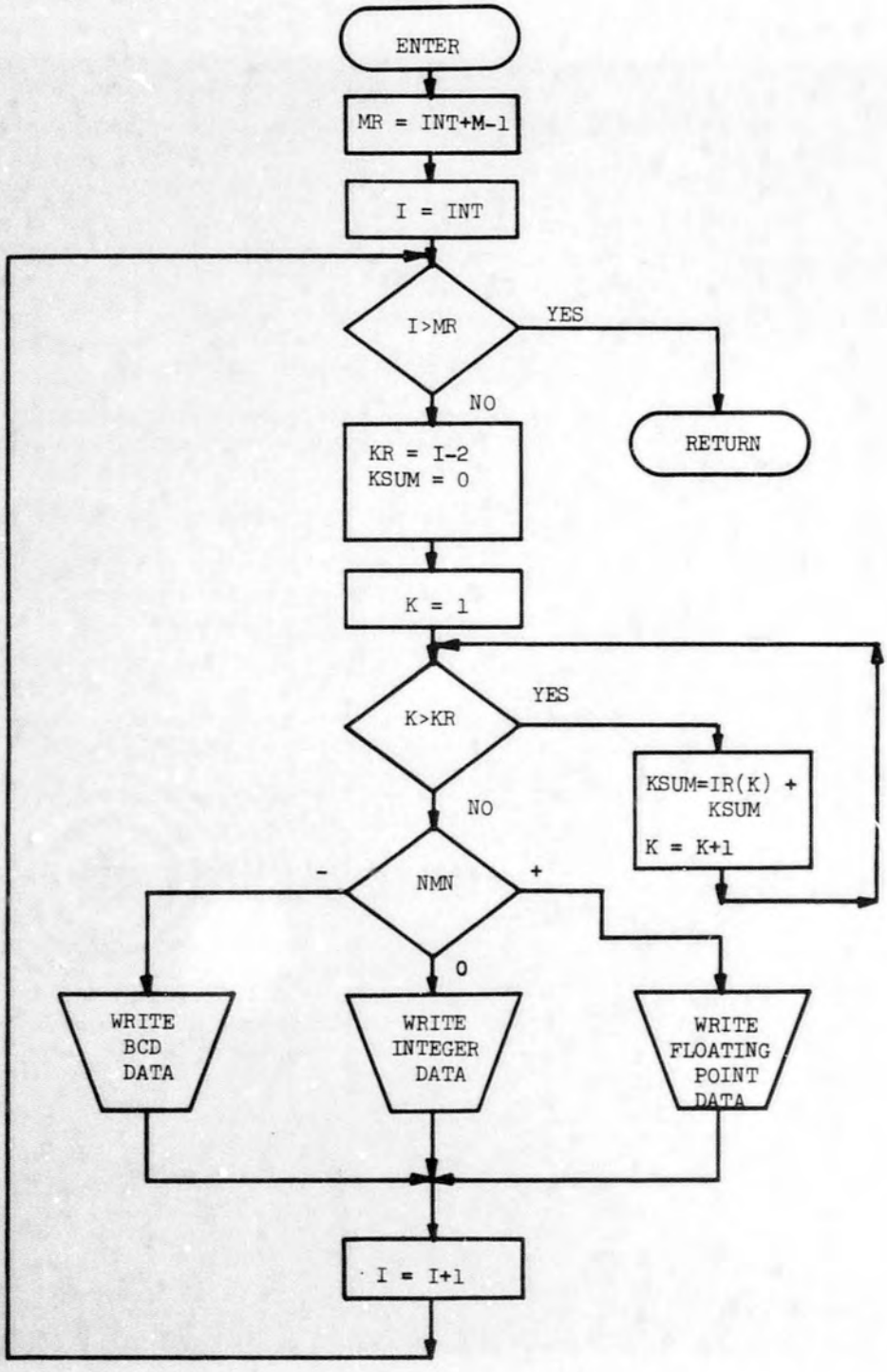
a. Function

Data read in by subroutine GINPUT is printed by this routine. This routine is a general routine for printing data stored in an array such as the one set up in GINPUT.

b. Symbols

Symbolic representation	Meaning or use
BCD	Array of variable names
ARRAY	Array of values of variables
INT	Location in BCD array of first BCD, integer, or floating point (FP) variable name
M	Number of BCD, integer, or FP variables in array
MR	Location in BCD array of last BCD, integer, or FP variable name
I	Index of BCD array
IR	Number of data reads in ARRAY corresponding to each BCD variable
KR	Index in ARRAY corresponding to BCD array
IPR	Lower value of ARRAY index corresponding to a variable in BCD array
KSUM	Upper value of ARRAY index corresponding to a variable in BCD array
NMH	Index specifying type of variable being printed (-1 = BCD, 0 = Integer, +1 = FP)

c. Flow Chart


ENTER
MR = INT+M-1
I = INT
I>MR
YES
RETURN
NO
KR = I-2
KSUM = 0
K = 1
K>KR
YES
KSUM=IR(K) + KSUM
K = K+1
NO
NMN
-
+
0
WRITE BCD DATA
WRITE INTEGER DATA
WRITE FLOATING POINT DATA
I = I+1

D. LISTING

```
      SUBROUTINE DDUMP(ARRAY,BCD,IR)
      DIMENSION ARRAY(1),BCD(1),IR(1)
      COMMON /456 / ITAPE,JTAPE,INT,NMN,M,NGR
      MR = INT + M - 1
      DO 20 I = INT,MR
      KR = I- 2
      KSUM = 0
      DO 10 K = 1, KR
   10 KSUM = IR(K)  + KSUM
      IPR  = KSUM - IR(I-2) + 1
      IF (NMN) 3, 4, 5
    3 WRITE(JTAPE,6)  BCD(I), (ARRAY(J),J= IPR, KSUM )
      GO TO  20
    6 FORMAT (1H0,(10X6A10))
    4 WRITE(JTAPE,7) BCD(I), (ARRAY(J),J=IPR,KSUM)
      GO TO  20
    7 FORMAT(1H0,A6,1H=,(10I6))
    5 WRITE(JTAPE,8) BCD(I), (ARRAY(J),J=IPR,KSUM)
    8 FORMAT(1H0,A6,1H=,(6E15.8))
   20 CONTINUE
      RETURN
      END
```

11. SUBROUTINE SHOCKR

a. Theory

The thermodynamic properties of pressure, temperature, and enthalpy behind either a normal shock or an oblique shock are determined in this subroutine. In addition, this subroutine gives the flow angle behind the shock and obtains the stagnation value of gamma. For a given set of free stream conditions, iterations are performed to satisfy the Rankine-Hugoniot relation

$$\left(\frac{P_2 - P_1}{2}\right)\left(\frac{1}{\rho_1} + \frac{1}{\rho_2}\right) = h_2 - h_1 \tag{61}$$

along with the conservation equations

$$\rho_1 U_1 = \rho_2 U_2 \tag{62}$$

and

$$P_1 + \rho_1 U_1^2 = P_2 + \rho_2 U_2^2 \tag{63}$$

A maximum and a minimum pressure behind the shock are first determined. The maximum pressure is the solution to the momentum equation normal to the shock, assuming the mass flow behind the shock to be zero.

$$P_{MAX} = P_1 + \rho_1 \left(U_1 \sin \eta_s\right)^2 \tag{64}$$

The minimum pressure is obtained by assuming an ideal gas with a gamma of 1.4.

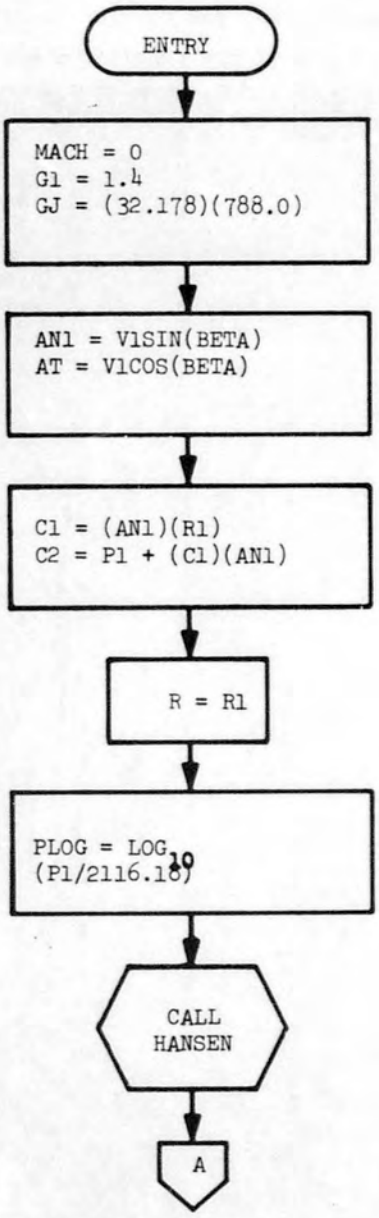
$$P_{MIN} = P_1 + \rho_1 \left(U_1 \sin \eta_s\right)^2 \left[1 - \frac{\gamma - 1 + \frac{2}{M_\infty^2}}{\gamma + 1}\right] \tag{65}$$

A pressure between the maximum and minimum values is selected and the density behind the shock is found from the continuity and momentum equations.

$$\rho_2 = \frac{\rho_1}{1 - \frac{P_2 - P_1}{\rho_1 U_1^2}} \tag{66}$$

The enthalpy is obtained through the real gas routine described in the next section. A check then is made to see if the conditions behind the shock satisfy the Rankine-Hugoniot relation (equation 61). If not, a new estimate is made for the pressure behind the shock, and the above process is repeated until the ratio of the maximum to minimum pressure is within a prescribed tolerance.

b. Flow Chart



INITIALIZE CONSTANTS

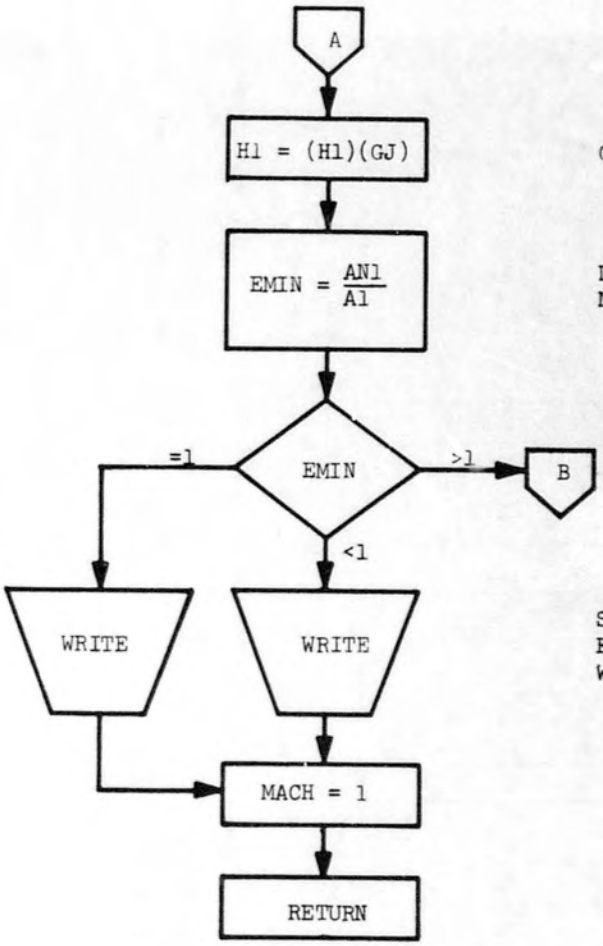
DETERMINE VELOCITY COMPONENTS NORMAL TO AND TANGENT TO THE SHOCK

STORE $(P_1 U_{1n})$ AND $(P_1 + P_1 U_{1n}^2)$

INITIALIZE R FOR SUBROUTINE HANSEN

OBTAIN THE LOG_{10} OF PRESSURE IN ATMOSPHERES FOR SUBROUTINE HANSEN

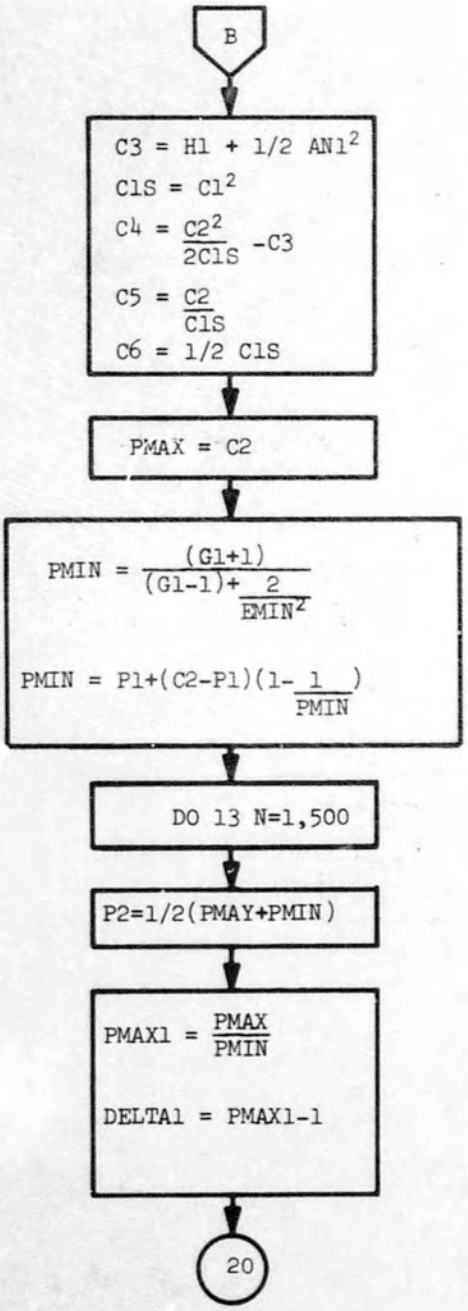
OBTAIN STATIC ENTALPY



CHANGE UNITS OF H1

DETERMINE MACH NUMBER NORMAL TO THE SHOCK

SHOCK ANGLE IS LESS OR EQUAL TO THE MACH ANGLE. WRITE THIS MESSAGE.



STORE TOTAL ENTALPY,

$(P_1U_{in})^2, [\frac{P_1+P_1U^2_{in}}{P_1U_1}]^2, \frac{P_1+P_1U^2_{in}}{(P_1U_1)^2},$

AND $1/2(P_1U_{in})^2$

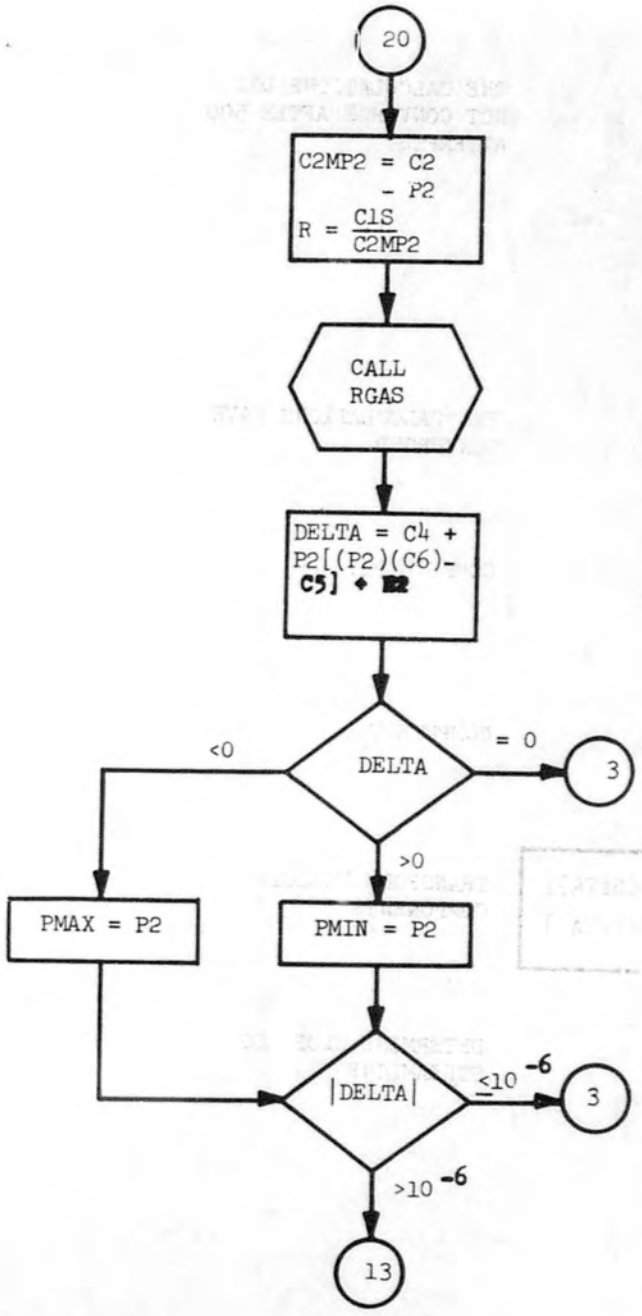
DETERMINE MAXIMUM VALUE OF PRESSURE BEHIND THE SHOCK

DETERMINE MINIMUM VALUE OF PRESSURE BEHIND THE SHOCK BY PERFECT GAS ESTIMATE

START ITERATION LOOP TO DETERMINE PRESSURE BEHIND THE SHOCK, P2.

GUESS AT P2

STORE RATIO OF PMAX TO PMIN



DETERMINE DENSITY BEHIND THE SHOCK

DETERMINE SPECIFIC ENTHALPY BEHIND THE SHOCK

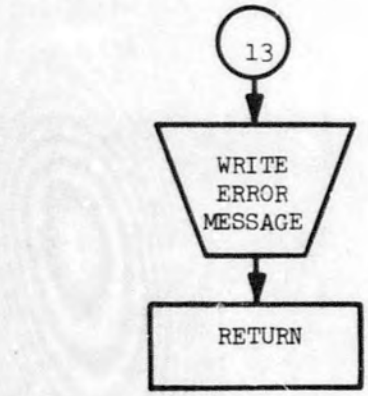
DETERMINE CHANGE IN TOTAL ENTHALPY ACROSS THE SHOCK

IF DELTA IS ZERO, THE RANKING HUGONIOT RELATIONS HAVE BEEN SATISFIED

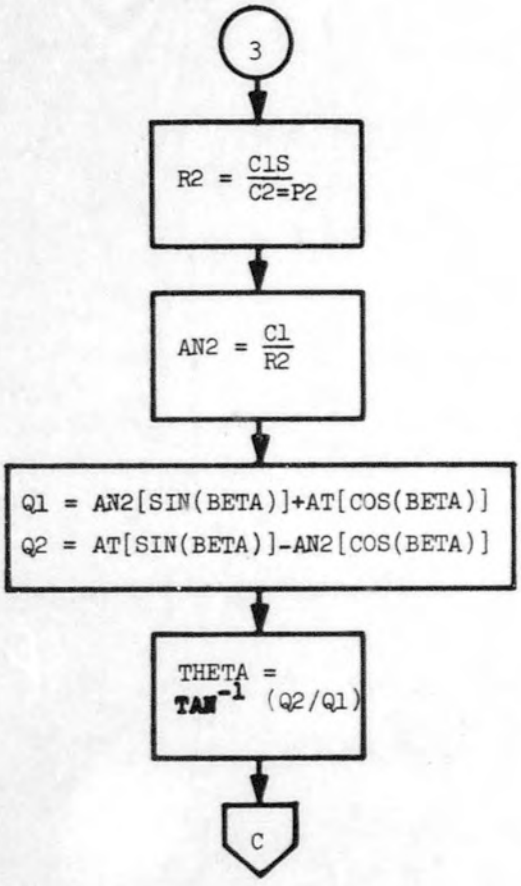
DELTA WAS NOT ZERO. CHANGE PMAX OR PMIN TO CONVERGE ON CORRECT SOLUTION.

IS THE ERROR $\leq 10^{-6}$

CONTINUE DO LOOP



THE CALCULATIONS DID NOT CONVERGE AFTER 500 ATTEMPTS



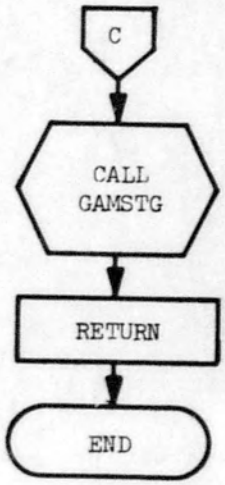
THE CALCULATIONS HAVE CONVERGED

COMPUTE P_2.

COMPUTE U_2.

TRANSFORM VELOCITY COMPONENTS

DETERMINE SLOPE OF STREAMLINE



DETERMINE STAGNATION
VALUE OF γ

```
C.  LISTING
      SUBROUTINE SHOCKR(P1,R1,V1,BETA,THETA,GSTAG,P2,MACH)
C
C     COMPUTES PROPERTIES BEHIND A SHOCK
C
      COMMON /1/ A1,T1,R
      MACH=0
      G1=1.4
      GJ=32.178*778.0
      AN1=V1*SIN(BETA)
      AT=V1*COS(BETA)
      C1=AN1*R1
      C2=P1+AN1*C1
      R=R1
      PLOG=ALOG10(P1/2116.18)
      CALL HANSEN(3,T1,PLOG,H1)
      H1=H1*GJ
C
C     CHECK MACH NUMBER BASED ON NORMAL VELOCITY
C
      EMIN=AN1/A1
      IF(EMIN-1.0)4,5,6
 4    WRITE(6,7)
 7    FORMAT(10X,31HWAVE ANGLE LESS THAN MACH ANGLE)
      GO TO 25
 5    WRITE(6,9)
 9    FORMAT(10X,24HTHIS WAVE IS A MACH WAVE)
 25   MACH=1
      GO TO 8
C
C     TOTAL ENTHALPY
C
 6    C3=H1+(AN1**2)/2.0
      C1S=C1**2
      C4=C2*C2/C1S/2.0-C3
      C5=C2/C1S
      C6=0.5/C1S
C
C     PMAX ASSUMES VELOCITY BEHIND SHOCK IS ZERO
C
      PMAX=C2
      PMIN=(G1+1.0)/((G1-1.0)+2.0/EMIN**2)
      PMIN=P1+(C2-P1)*(1.0-1.0/PMIN)
      DO 13 N=1,500
      P2=(PMAX+PMIN)/2.0
      PMAX1=PMAX/PMIN
      DELT1=PMAX1-1.0
 20   C2MP2=C2-P2
      R=C1S/C2MP2
      CALL RGAS(P2,R,H2,T2)
      DELTA=C4+P2*(P2*C6-C5)+H2
C
C     IF DELTA IS ZERO, THE RANKINE-HUGONIOT RELATIONSHIPS
C     ARE SATISFIED FOR THE ABOVE P2 AND RHO2
C
      IF(DELTA)10,3,12
 10   PMAX=P2
      GO TO 11
```

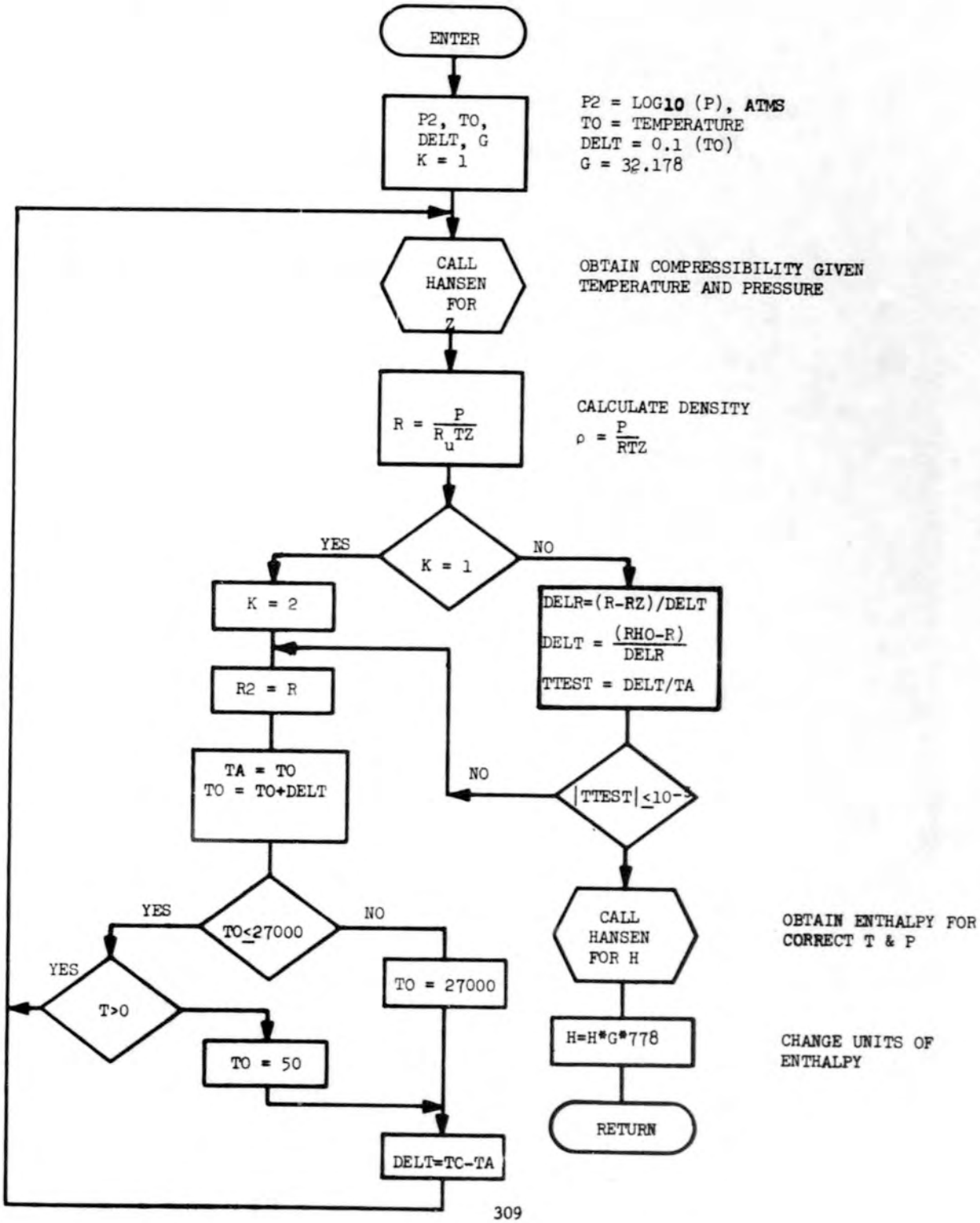
```
   12    PMIN=P2
   11    IF(ABS(DELT1)-0.000001)3,3,13
   13    CONTINUE
         WRITE(6,16)
   16    FORMAT(10X,14HERROR IN SHOCK)
         GO TO 8
    3    R2=C1S/(C2-P2)
         AN2=C1/R2
         Q1=AN2*SIN(BETA)+AT*COS(BETA)
         Q2=AT*SIN(BETA)-AN2*COS(BETA)
         THETA=ATAN(Q2/Q1)
C
C        OBTAIN GAMMA - STAG
C
         CALL GAMSTG(T2,P2,GSTAG)
    8    RETURN
         END
```

12. SUBROUTINE RGAS

a. Function

This subroutine iterates on a temperature which satisfies a given pressure and density. The temperature is then used to obtain an enthalpy at the given pressure. Properties from Subroutine HANSEN are used.

b. Flow Chart


ENTER
P2, TO, DELT, G K = 1
P2 = LOG10 (P), ATMS
TO = TEMPERATURE
DELT = 0.1 (TO)
G = 32.178
CALL HANSEN FOR Z
OBTAIN COMPRESSIBILITY GIVEN TEMPERATURE AND PRESSURE
R = P / R_u TZ
CALCULATE DENSITY
ρ = P / RTZ
K = 1
YES
NO
K = 2
DELR=(R-RZ)/DELT
DELT = (RHO-R) / DELR
TTEST = DELT/TA
R2 = R
TA = TO
TO = TO+DELT
NO
|TTEST| ≤ 10⁻³
YES
TO≤27000
NO
YES
T>0
TO = 27000
TO = 50
DELT=TC-TA
CALL HANSEN FOR H
OBTAIN ENTHALPY FOR CORRECT T & P
H=H*G*778
CHANGE UNITS OF ENTHALPY
RETURN

```
C.  LISTING

      SUBROUTINE RGAS(P,RHO,H,TO)
C
C     COMPUTES ENTHALPY FOR GIVEN PRESSURE AND DENSITY
C     PRESSURE - PSF, DENSITY - SLUGS/FT3
C
      P2=ALOG10(P/2116.18)
      TO=5000.0
      DELT=0.1*TO
      G=32.178
      K=1
  200 CALL HANSEN(5,TO,P2,Z)
      R=P/(53.35*Z*TO)/G
      GO TO(260,270),K
  260 K=2
  261 R2=R
      TA=TO
      TO=TO+DELT
      IF(TO-27000.0)267,267,265
  265 TO=27000.0
  266 DELT=TO-TA
      GO TO 200
  267 IF(TO)268,268,200
  268 TO=50.0
      GO TO 266
  270 DELR=(R-R2)/DELT
      DELT=(RHO-R)/DELR
      TTEST=DELT/TA
      IF(ABS(TTEST)-0.00001)299,299,261
  299 CONTINUE
C
C     HAVE FOUND PROPER TEMPERATURE, NOW GET ENTHALPY
C
      CALL HANSEN(3,TO,P2,H)
      H=H*G*778.0
      RETURN
      END
```

13. SUBROUTINE GAMSTG

a. Function

The stagnation value of gamma, the ratio of specific heats, is obtained by a table look-up procedure from data compiled into the subroutine. Gamma is a function of temperature and pressure. The data corresponds to the dimensionless speed of sound tables found in Subroutine HANSEN (Ref. 14).

B. LISTING

```
      SUBROUTINE GAMSTG(TO,PO,GSOS)
C
C     STAGNATION GAMMA OR DIMENSIONLESS SPEED OF SOUND
C
      DIMENSION T(31),P(7),SOS(217)
      DATA (SOS(I),I=1,124) /
     1 0.000, 1.387, 1.337, 1.306, 1.144, 1.136, 1.302, 1.171, 1.097,
     2 1.092, 1.124, 1.249, 1.263, 1.183, 1.140, 1.128, 1.130, 1.136,
     3 1.145, 1.157, 1.185, 1.244, 1.334, 1.424, 1.489, 1.528, 1.548,
     4 1.558, 1.563, 1.565, 1.567, 0.000, 1.387, 1.337, 1.310, 1.206,
     5 1.119, 1.232, 1.250, 1.137, 1.101, 1.103, 1.133, 1.225, 1.260,
     6 1.205, 1.162, 1.143, 1.140, 1.144, 1.151, 1.159, 1.169, 1.187,
     7 1.221, 1.276, 1.346, 1.415, 1.470, 1.508, 1.532, 1.546, 0.000,
     8 1.387, 1.337, 1.312, 1.259, 1.144, 1.150, 1.265, 1.208, 1.133,
     9 1.111, 1.113, 1.135, 1.193, 1.249, 1.231, 1.193, 1.168, 1.157,
     * 1.155, 1.159, 1.165, 1.173, 1.181, 1.192, 1.207, 1.232, 1.269,
     * 1.317, 1.370, 1.420, 0.000, 1.387, 1.337, 1.312, 1.285, 1.196,
     * 1.147, 1.187, 1.254, 1.196, 1.143, 1.124, 1.124, 1.136, 1.167,
     * 1.216, 1.242, 1.228, 1.203, 1.185, 1.176, 1.173, 1.175, 1.180,
     * 1.187, 1.194, 1.202, 1.210, 1.221, 1.235, 1.254/
      DATA (SOS(I),I=125,217) /
     1 0.000, 1.387, 1.337, 1.312, 1.296, 1.247, 1.181, 1.166, 1.204,
     2 1.241, 1.202, 1.161, 1.141, 1.137, 1.142, 1.156, 1.181, 1.214,
     3 1.237, 1.237, 1.225, 1.212, 1.202, 1.198, 1.197, 1.198, 1.202,
     4 1.208, 1.214, 1.221, 1.228, 0.000, 1.387, 1.337, 1.312, 1.299,
     5 1.276, 1.229, 1.192, 1.187, 1.210, 1.233, 1.217, 1.186, 1.165,
     6 1.155, 1.154, 1.159, 1.169, 1.185, 1.206, 1.228, 1.242, 1.246,
     7 1.242, 1.236, 1.231, 1.227, 1.226, 1.226, 1.228, 1.232, 0.000,
     8 1.387, 1.337, 1.312, 1.300, 1.288, 1.265, 1.234, 1.212, 1.208,
     9 1.217, 1.229, 1.230, 1.214, 1.195, 1.182, 1.175, 1.174, 1.176,
     * 1.181, 1.190, 1.201, 1.216, 1.232, 1.247, 1.257, 1.263, 1.265,
     * 1.265, 1.264, 1.263/
      DATA P/
     1  -4.0,  -3.0,  -2.0,  -1.0,   0.0,   1.0,   2.0/
      DATA T/
     1          0.0,        900.0,       1800.0,       2700.0,
     2       3600.0,       4500.0,       5400.0,       6300.0,
     3       7200.0,       8100.0,       9000.0,       9900.0,
     4      10800.0,      11700.0,      12600.0,      13500.0,
     5      14400.0,      15300.0,      16200.0,      17100.0,
     6      18000.0,      18900.0,      19800.0,      20700.0,
     7      21600.0,      22500.0,      23400.0,      24300.0,
     8      25200.0,      26100.0,      27000.0/
      P1=ALOG10(PO/2116.18)
      CALL DISCOT(TO,P1,T,SOS,P,11,217,7,GSOS,31)
      RETURN
      END
```

14. SUBROUTINE GAMCOF

a. Function

The coefficients for determining the effective gamma with altitude are interpolated from tables compiled into the program with this subroutine. The compiled data are found in block COEFF. For any altitude, gamma itself is given by the equation

$$\gamma_E = C_0 + C_1 V_N + C_2 V_N^2 + C_3 V_N^3 \tag{67}$$

where

$$V_N = V_\infty \left(\sin \eta_s\right) \tag{68}$$

The tabulated data in block COEFF was obtained through a least squares curve fit of the data presented in reference 12.

B. LISTING

```
      SUBROUTINE GAMCOF(C0,C1,C2,C3,ALT)
C
C     TO DETERMINE COEFFICIENTS FOR GAMMA-E WITH ALTITUDE
C
      COMMON/COEFF/C
      DIMENSION C(4,15),XC(4)
C
C     CHECK ALTITUDE RANGE
C
      IF(ALT-40000.0)15,15,25
 15   I1=1
      I2=2
      ALT1=0.0
      ALT2=40000.0
      GO TO 35
 25   I1=ALT/20000.0+0.01
      XI1=I1
      XI2=I1+1
      I2=I1+1
      ALT1=XI1*20000.0
      ALT2=XI2*20000.0
 35   FACTOR=(ALT-ALT1)/(ALT2-ALT1)
      DO 45 I=1,4
      X1=C(I,I1)
      X2=C(I,I2)
      XC(I)=X1+FACTOR*(X2-X1)
 45   CONTINUE
      C0=XC(1)
      C1=XC(2)
      C2=XC(3)
      C3=XC(4)
      RETURN
      END
```

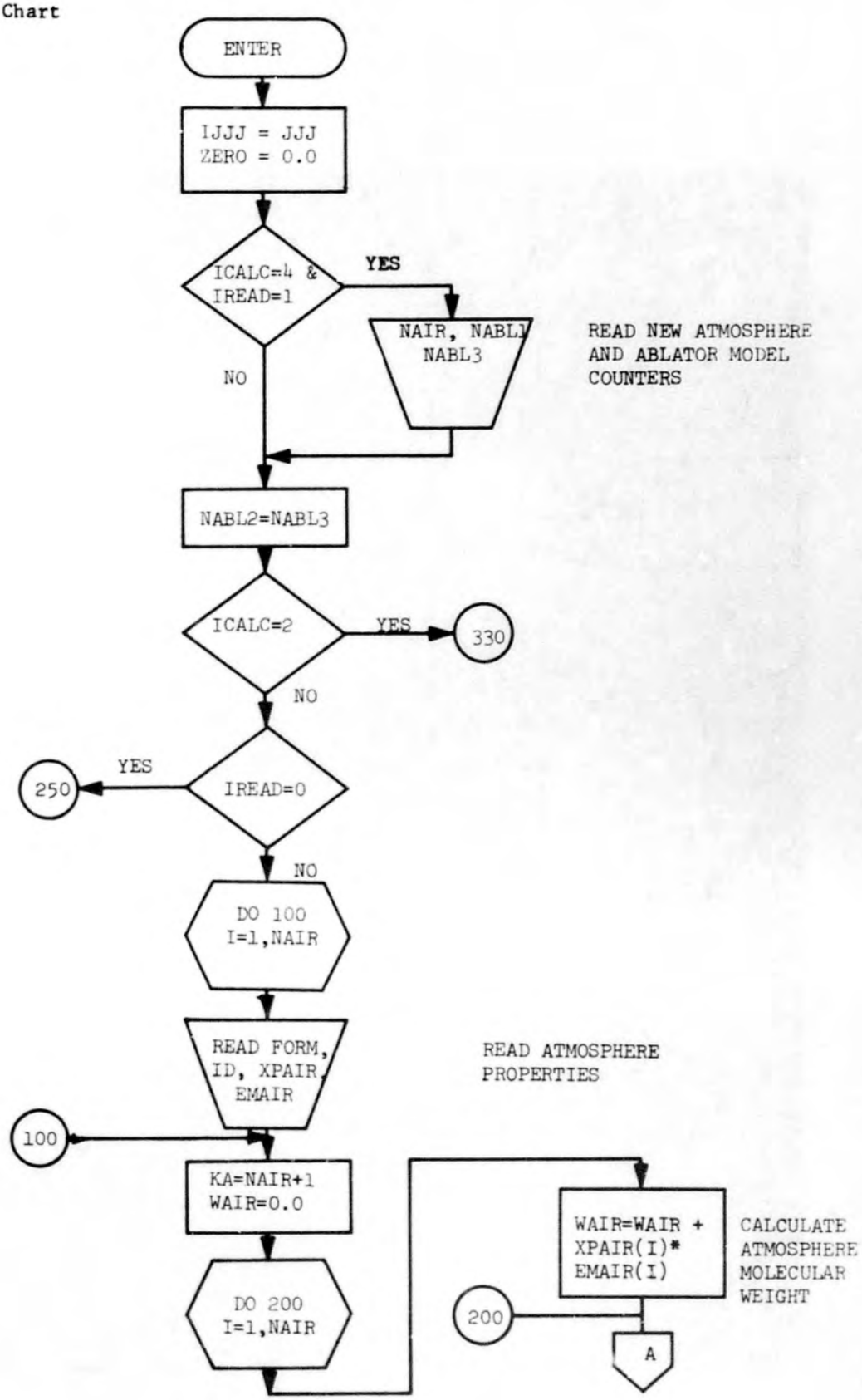
```
C.  BLOCK DATA

      BLOCK DATA
      COMMON/COEFF/C(4,15)
      DATA (C(I),I=1,60,4) /
     1     1.44388008,      1.46454239,      1.47091770,      1.47091770,
     2     1.46761513,      1.48238754,      1.47857666,      1.49412155,
     3     1.49161625,      1.50116444,      1.54762554,      1.57243685,
     4     1.60347462,      1.47962189,      1.36062813/
      DATA (C(I),I=2,60,4) /
     1-.259505614E-04, -.315756333E-04, -.362302671E-04,-.362302671E-04,
     2-.373303046E-04, -.480244926E-04, -.474223634E-04,-.600036292E-04,
     3-.580171909E-04, -.620707287E-04, -.888469949E-04,-.104536768E-03,
     4-.117537551E-03, -.524828210E-04, -.108298846E-04/
      DATA (C(I),I=3,60,4) /
     1 .221432228E-08,  .286723356E-08,  .395843713E-08, .395843713E-08,
     2 .451797177E-08,  .683408174E-08,  .686056723E-08, .972114123E-08,
     3 .942988621E-08,  .102453477E-07,  .156302455E-07, .185593443E-07,
     4 .211443023E-07,  .103922346E-07, -.214342322E-09/
      DATA (C(I),I=4,60,4) /
     1-.147850265E-12, -.172141381E-12, -.254501498E-12,-.254501498E-12,
     2-.305486864E-12, -.469595662E-12, -.480664120E-12,-.686303336E-12,
     3-.680893384E-12, -.739590626E-12, -.109799409E-11,-.128521586E-11,
     4-.146403115E-11, -.910443596E-12, -.361368975E-12/
      END
```

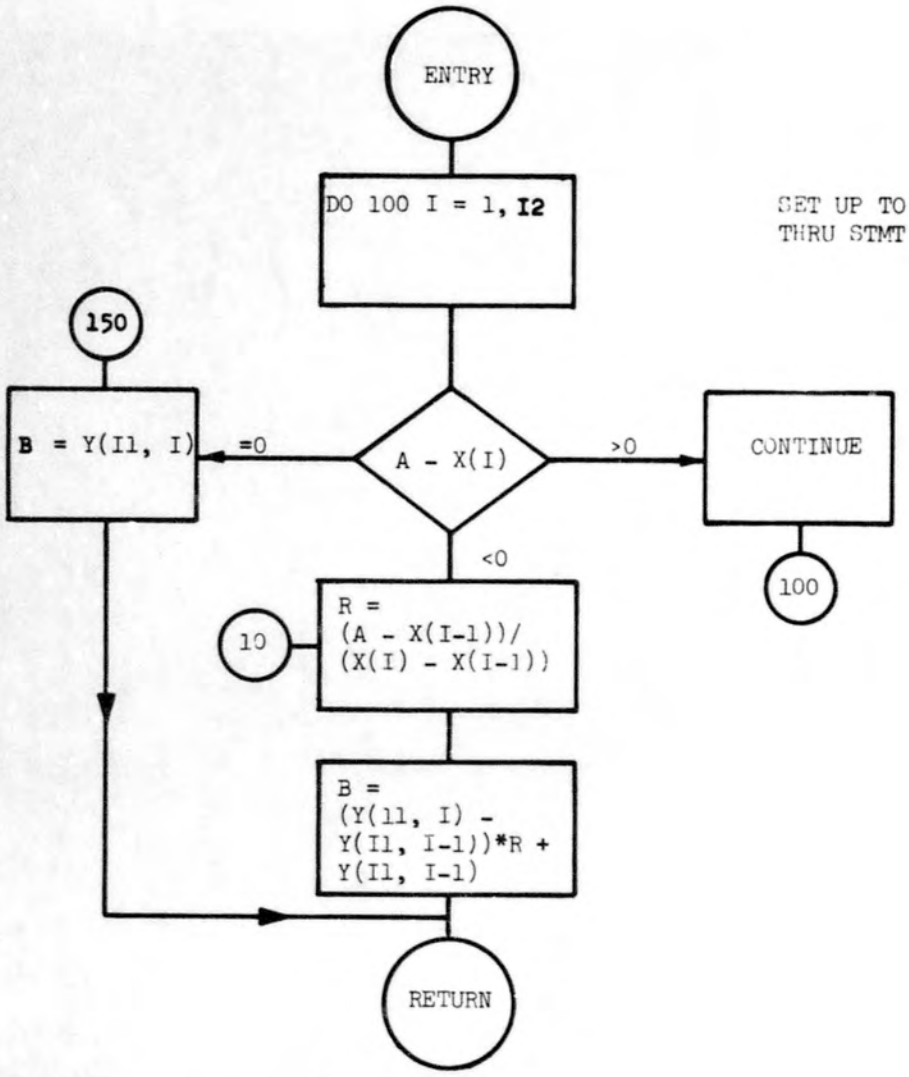
15. SUBROUTINE INPUT

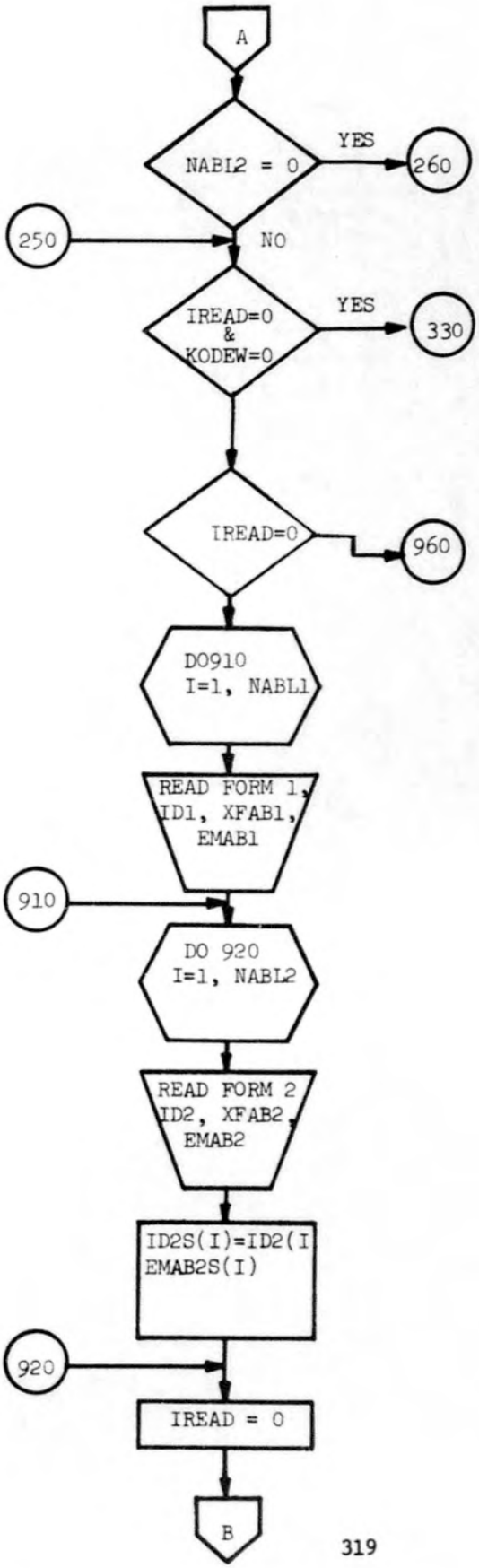
a. Function

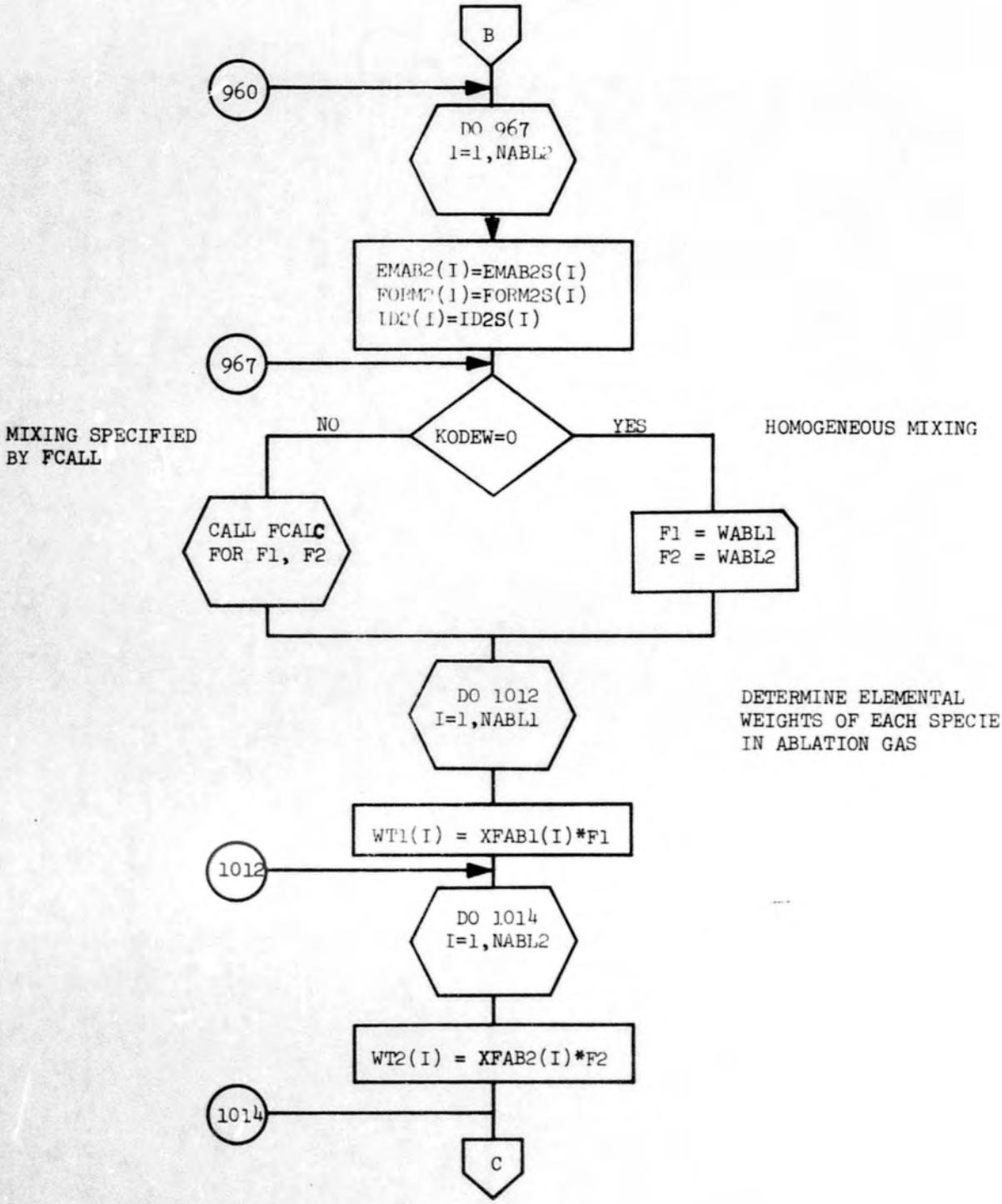
This subroutine converts the local value of mass fraction to the proper values of mole fraction for the appropriate basic chemical species of the ablation gas-air mixtures. The mole fractions thus supplied are case data for the equilibrium thermochemical calculations.

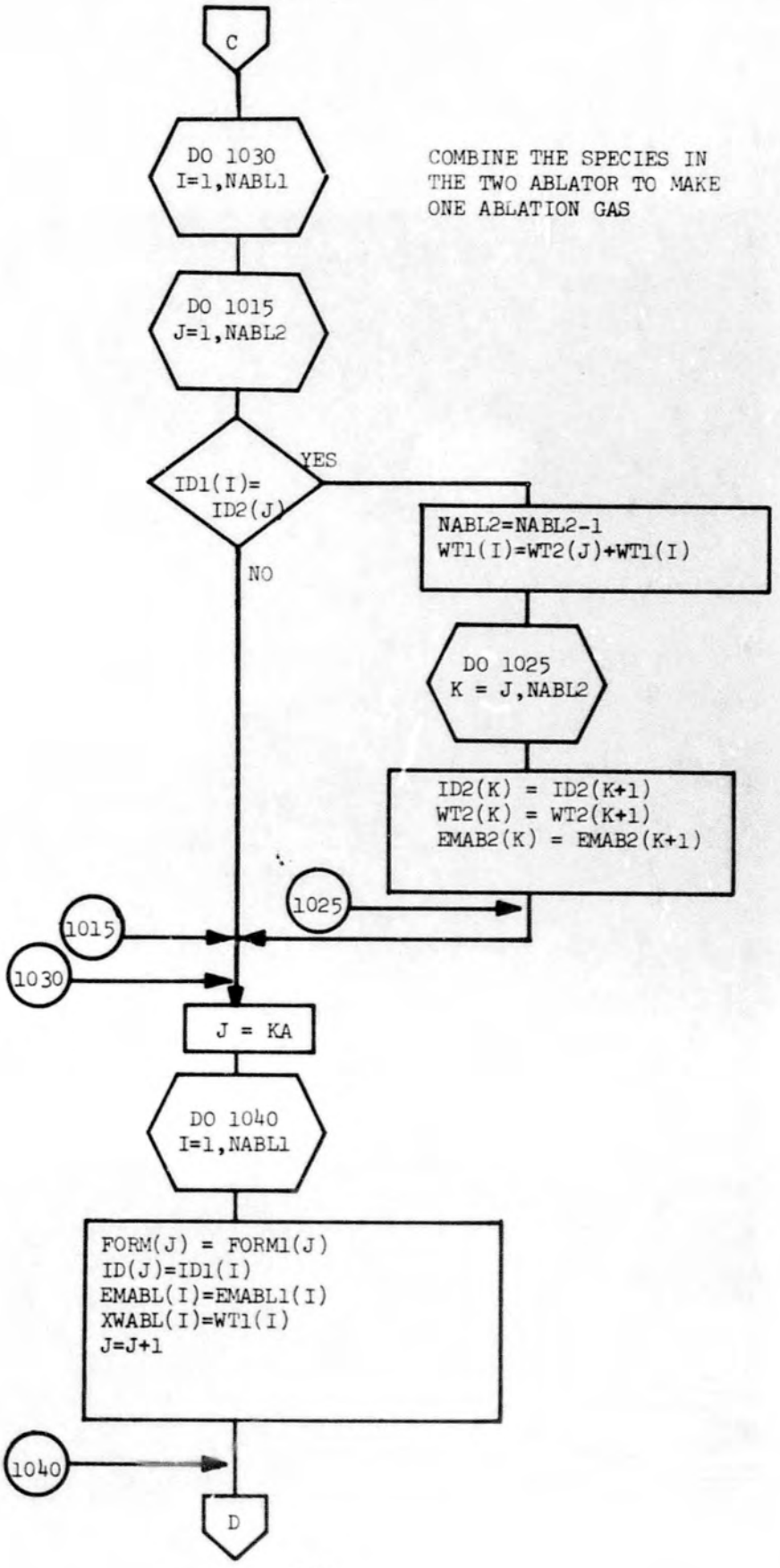
b. Flow Chart

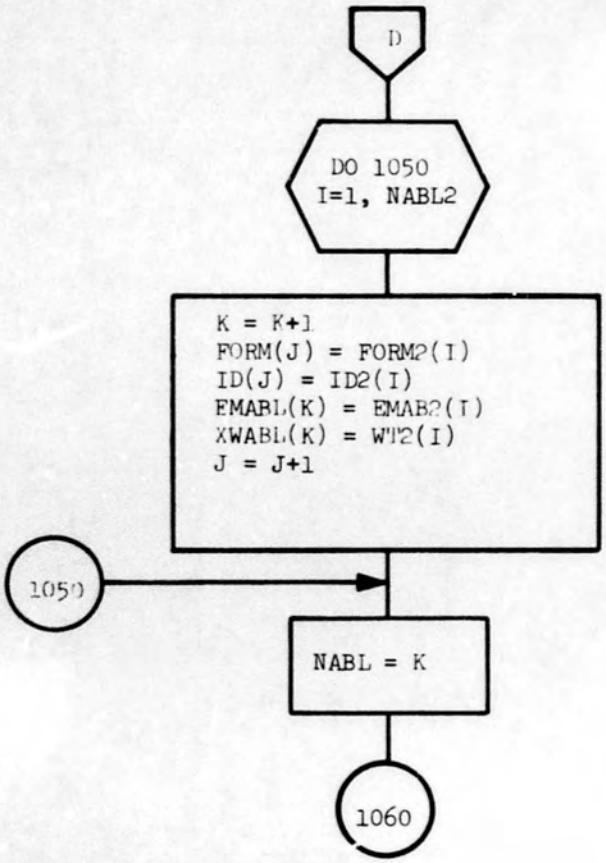

ENTER
IJJJ = JJJ
ZERO = 0.0
ICALC=4 & IREAD=1
YES
NO
NAIR, NABL1 NABL3
READ NEW ATMOSPHERE AND ABLATOR MODEL COUNTERS
NABL2=NABL3
ICALC=2
YES
330
NO
IREAD=0
YES
250
NO
DO 100 I=1,NAIR
READ FORM, ID, XPAIR, EMAIR
READ ATMOSPHERE PROPERTIES
100
KA=NAIR+1
WAIR=0.0
DO 200 I=1,NAIR
WAIR=WAIR + XPAIR(I)* EMAIR(I)
CALCULATE ATMOSPHERE MOLECULAR WEIGHT
200
A

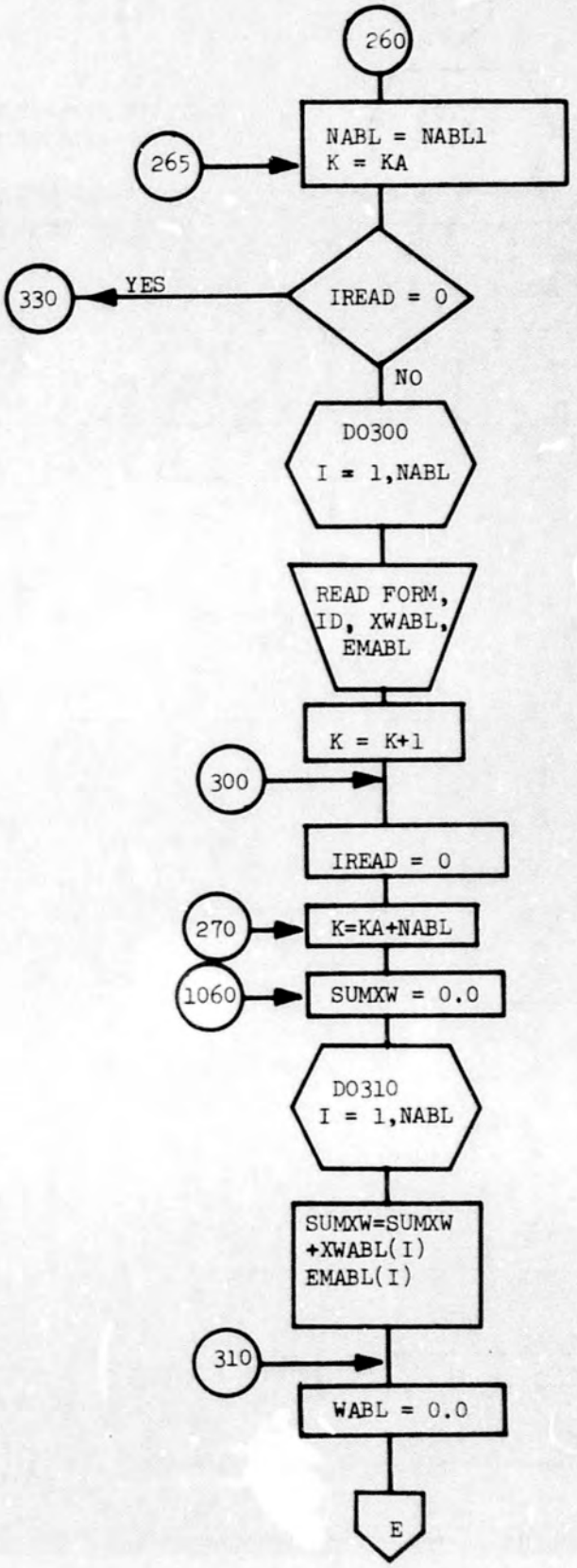

ENTRY
DO 100 I = 1, I2
SET UP TO LOOP 100 TIMES
THRU STMT 100
150
B = Y(I1, I)
=0
A - X(I)
>0
CONTINUE
<0
100
10
R =
(A - X(I-1))/
(X(I) - X(I-1))
B =
(Y(I1, I) -
Y(I1, I-1))*R +
Y(I1, I-1)
RETURN


A
NABL2 = 0
YES
260
IS THERE MORE THAN ONE ABLATION MATERIAL?
250
NO
IREAD=0 & KODEW=0
YES
330
IREAD=0
960
DO910 I=1, NABL1
READ FORM 1, ID1, XFAB1, EMAB1
READ CHEMICAL COMPOSITION OF FIRST ABLATOR
910
DO 920 I=1, NABL2
READ FORM 2 ID2, XFAB2, EMAB2
READ CHEMICAL COMPOSITION OF SECOND ABLATOR
ID2S(I)=ID2(I
EMAB2S(I)
920
IREAD = 0
B


B
960
DO 967
I=1,NABL2
EMAB2(I)=EMAB2S(I)
FORM2(I)=FORM2S(I)
ID2(I)=ID2S(I)
967
MIXING SPECIFIED BY FCALL
NO
KODEW=0
YES
HOMOGENEOUS MIXING
CALL FCALC FOR F1, F2
F1 = WABL1
F2 = WABL2
DO 1012
I=1,NABL1
DETERMINE ELEMENTAL WEIGHTS OF EACH SPECIE IN ABLATION GAS
WT1(I) = XFAB1(I)*F1
1012
DO 1014
I=1,NABL2
WT2(I) = XFAB2(I)*F2
1014
C

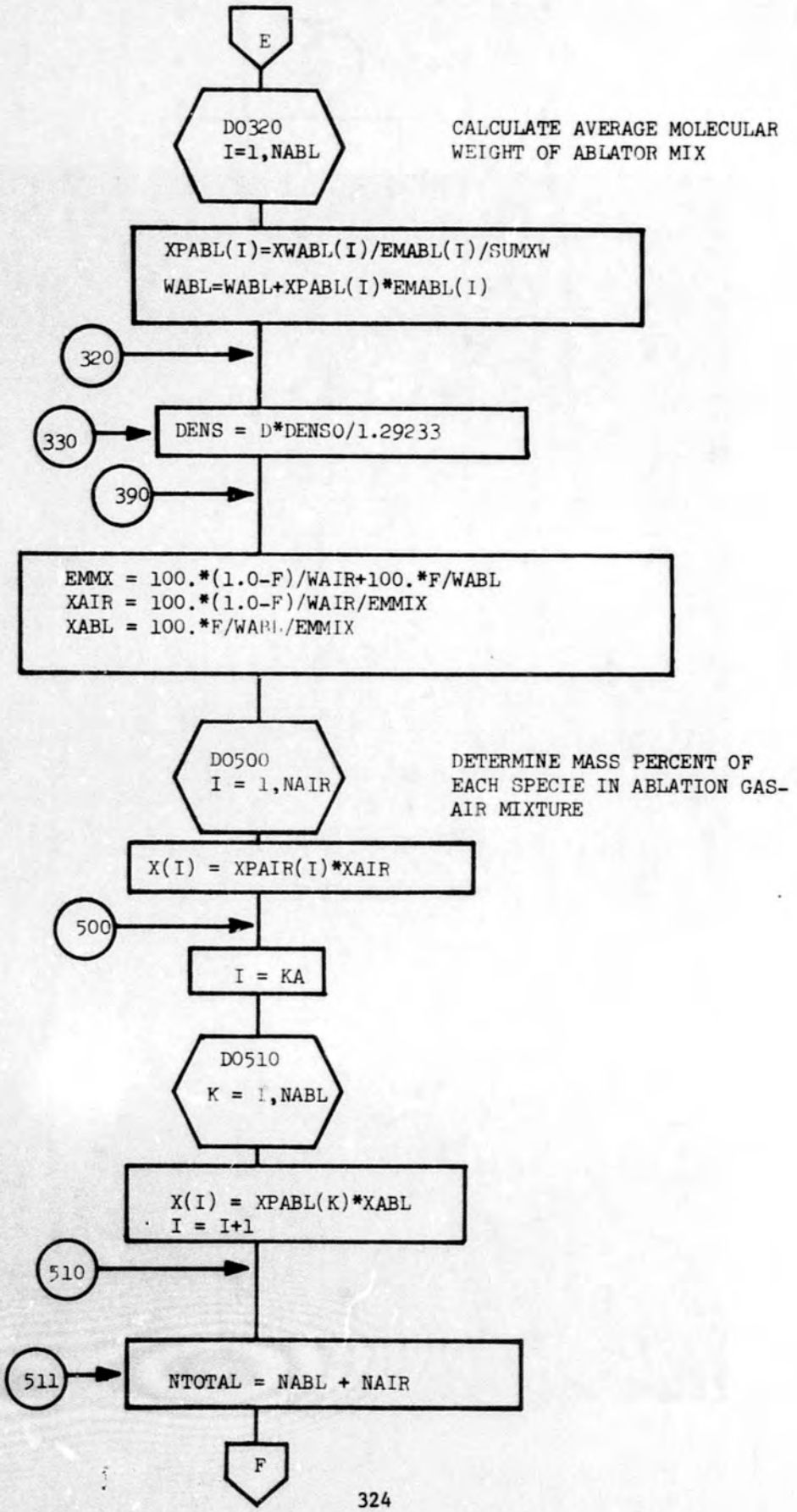

C
DO 1030
I=1,NABL1
COMBINE THE SPECIES IN
THE TWO ABLATOR TO MAKE
ONE ABLATION GAS
DO 1015
J=1,NABL2
ID1(I)=
ID2(J)
YES
NO
NABL2=NABL2-1
WT1(I)=WT2(J)+WT1(I)
DO 1025
K = J,NABL2
ID2(K) = ID2(K+1)
WT2(K) = WT2(K+1)
EMAB2(K) = EMAB2(K+1)
1025
1015
1030
J = KA
DO 1040
I=1,NABL1
FORM(J) = FORM1(J)
ID(J)=ID1(I)
EMABL(I)=EMABL1(I)
XWABL(I)=WT1(I)
J=J+1
1040
D

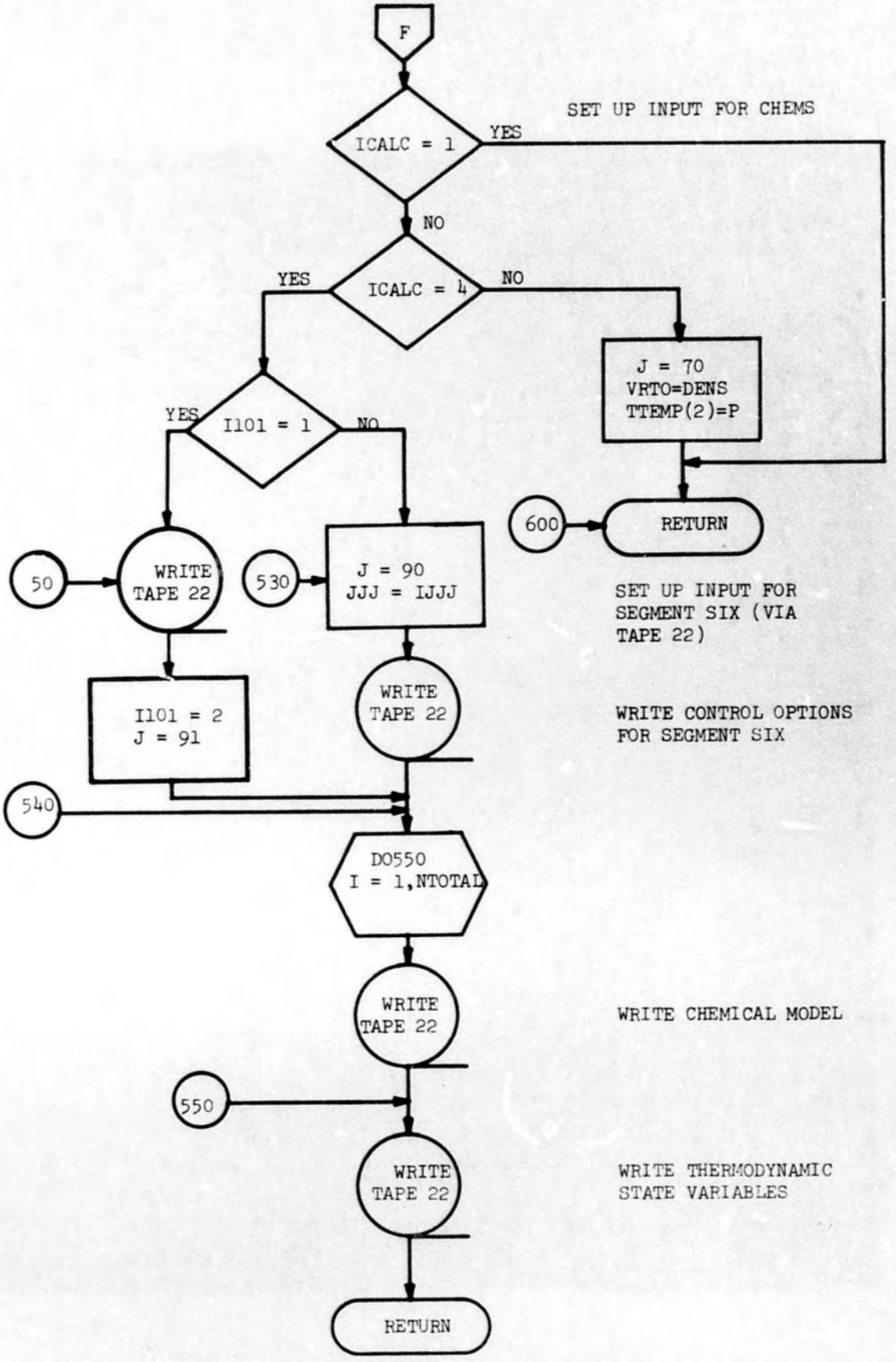

D
DO 1050
I=1, NABL2
K = K+1
FORM(J) = FORM2(I)
ID(J) = ID2(I)
EMABL(K) = EMAB2(I)
XWABL(K) = WT2(I)
J = J+1
1050
NABL = K
1060



READ DATA FOR SINGLE ABLATOR

CALCULATE NUMBER OF MOLES IN POUND OF MIX


E
DO320
I=1,NABL
CALCULATE AVERAGE MOLECULAR WEIGHT OF ABLATOR MIX
XPABL(I)=XWABL(I)/EMABL(I)/SUMXW
WABL=WABL+XPABL(I)*EMABL(I)
320
330
DENS = D*DENSO/1.29233
390
EMMX = 100.*(1.0-F)/WAIR+100.*F/WABL
XAIR = 100.*(1.0-F)/WAIR/EMMIX
XABL = 100.*F/WABL/EMMIX
DO500
I = 1,NAIR
DETERMINE MASS PERCENT OF EACH SPECIE IN ABLATION GAS-AIR MIXTURE
X(I) = XPAIR(I)*XAIR
500
I = KA
DO510
K = 1,NABL
X(I) = XPABL(K)*XABL
I = I+1
510
511
NTOTAL = NABL + NAIR
F


F
ICALC = 1
YES
NO
SET UP INPUT FOR CHEMS
ICALC = 4
YES
NO
J = 70
VRTO=DENS
TTEMP(2)=P
I101 = 1
YES
NO
600
RETURN
50
WRITE TAPE 22
530
J = 90
JJJ = IJJJ
SET UP INPUT FOR SEGMENT SIX (VIA TAPE 22)
I101 = 2
J = 91
WRITE TAPE 22
WRITE CONTROL OPTIONS FOR SEGMENT SIX
540
DO550
I = 1,NTOTAL
WRITE TAPE 22
WRITE CHEMICAL MODEL
550
WRITE TAPE 22
WRITE THERMODYNAMIC STATE VARIABLES
RETURN

C. LISTING

(1) SEGMENT TWO

```
      SUBROUTINE INPUT(NAIR,TO,WABL1,WABL2,NABL1,NABL3,IREAD,D,P,ICALC,
     * DENSO,P,TTEMP,VRTO,NTOTAL,U)
C     INPUT DATA FOR T666 (JPL PROGRAM)
C       THIS SUBROUTINE CREATES A TAPE FOR INPUT TO T666
      COMMON /234/ C(   39),   CON(   25),CONOLD(   25),     X(   15),
     1            CPP(   25),    DD(   25), DELDM(    2),   DSU(   25),
     2            EFR(25,39),    EK(25,39),   ENT(25,39),    FR(   25),
     3           FORM(   15),    ID       ,    NA(   10),   PHI(   25),
     4             SB(   25), SFCON(   25),  STSB(   10), TTEMP(    4),
     5            WTM(   25),   WTR(   25),     Y(25,25),     Z(   25)
      EQUIVALENCE (DUMMY(14),KODEW)
      COMMON /716/ DUMMY(31),K1,DUMMY2(251),II8,II9,DUMMY3(5),IJJJ,DUMMY
      COMMON /333/ XWABL(15),EMABL(15),FORM1(15),FORM2S(15),
     1             ID1(15),ID2S(15),EMAB1(15),EMAB2S(15),XFAB1(15),
     2             XFAB2(15)
      DIMENSION  XPAIR(3),EMAIR(3),ID(15),XPABL(15),WT1(15),WT2(15)
      DIMENSION FORM(3), ID2(15), EMAB2(15), FORM2(15)
      DATA I101/1/
    3 FORMAT (2XA4,I4,2F10.6)
    6 FORMAT(3H 70,5E12.6)
C      MOLECULAR WEIGHT OF AIR
      INTAP = 5
      IOTAP = 6
      IJJJ = JJJ
      ZERO = 0.0
      IF(ICALC.EQ.4.AND.IREAD.EQ.1) READ(5,798) NAIR,NABL1,NABL3
  798 FORMAT (13X,I2,2I5)
      NABL2=NABL3
      IF(ICALC.EQ.2) GO TO 330
      IF(IREAD.EQ.0) GO TO 250
      DO 100 I = 1,NAIR
      READ(INTAP,3) FORM(I),ID(I),XPAIR(I),EMAIR(I)
  100 CONTINUE
      KA = NAIR + 1
C     MOLECULAR WEIGHT OF ATMOSPHERE
      DO 201 I=1,NAIR
      XPAIR(I) = XPAIR(I)/100.
  201 CONTINUE
      WAIR=0.0
      DO 200 I = 1,NAIR
      WAIR=WAIR+XPAIR(I)*EMAIR(I)
  200 CONTINUE
C      MOLECULAR WEIGHT OF ABLATOR
      IF(NABL2.EQ.0) GO TO 260
  250 IF (IREAD.EQ.0 .AND. KODEW.EQ.0) GO TO 330
      IF (IREAD.EQ.0) GO TO 960
C      TWO-ABLATOR MIXTURE
      DO 910 I= 1,NABL1
      READ (INTAP,3) FORM1(I),ID1(I), XFAB1(I),EMAB1(I)
  910 CONTINUE
      DO 920 I=1,NABL2
      READ (INTAP,3) FORM2(I),ID2(I), XFAB2(I),EMAB2(I)
      ID2S(I) = ID2(I)
      EMAB2S(I) = EMAB2(I)
      FORM2S(I) = FORM2(I)
```

```
  920  CONTINUE
       IREAD =0
C      ELEMENTAL WEIGHTS IN ABLATOR
  960  DO 967 I= 1,NABL2
       EMAB2(I) = EMAB2S(I)
       FORM2(I) = FORM2S(I)
  967  ID2(I) = ID2S(I)
       IF (KODEW) 1000,1005,1000
 1000  CALL FCALC(U,F1,F2)
       GO TO 1010
 1005  F1 = WABL1
       F2 = WABL2
 1010  DO 1012 I= 1,NABL1
       WT1(I) = XFAB1(I) * F1
 1012  CONTINUE
       DO 1014 I = 1,NABL2
       WT2(I) = XFAB2(I) * F2
 1014  CONTINUE
C      MATCH ELEMENTS OF WABL2 WITH WABL1
       DO 1030 I=1,NABL1
       DO 1015 J=1,NABL2
       IF(ID1(I)-ID2(J))1015,1020,1015
 1020  NABL2=NABL2-1
       WT1(I)=WT2(J)+WT1(I)
       DO 1025 K=J,NABL2
       ID2(K)=ID2(K+1)
       WT2(K)=WT2(K+1)
        EMAB2(K)=EMAB2(K+1)
       FORM2(K)=FORM2(K+1)
 1025  CONTINUE
       GO TO 1030
 1015  CONTINUE
 1030  CONTINUE
        J=KA
C      MAKE INTO ONE ABLATOR
       DO 1040 I=1,NABL1
        FORM(J)=FORM1(I)
       ID(J)=ID1(I)
         EMABL(I)=EMAB1(I)
         XWABL(I)=WT1(I)
        J=J+1
 1040  CONTINUE
       K=NABL1
        DO 1050 I=1,NABL2
        K=K+1
       FORM(J)=FORM2(I)
       ID(J)=ID2(I)
        EMABL(K)=EMAB2(I)
        XWABL(K)=WT2(I)
        J=J+1
 1050  CONTINUE
 1051  NABL = K
        GO TO 1060
C      SINGLE ABLATOR
  260  NABL=NABL1
  265  K = KA
       IF(IREAD.EQ.0) GO TO 330
       DO 300 I=1,NABL
       READ (INTAP,3) FORM(K),ID(K),XWABL(I),EMABL(I)
       K=K+1
  300   CONTINUE
       IREAD =0
```

```
  270 K = KA + NABL
 1060 SUMXW=0.0
      DO 310 I=1,NABL
      SUMXW=SUMXW+XWABL(I)/EMABL(I)
  310 CONTINUE
      WABL=0.0
      DO 320 I=1,NABL
        XPABL(I)=XWABL(I)/EMABL(I)/SUMXW
      WABL=WABL+XPABL(I)*EMABL(I)
  320 CONTINUE
  330 DENS = D*DENSO /1.29233
  390 CONTINUE
C       MOLES OF MIXTURE
      EMMIX=100.*(1.0-F)/WAIR+100.*F/WABL
      XAIR=100.*(1.0-F)/WAIR/EMMIX
      XABL=100.*F/WABL/EMMIX
      DO 500 I=1,NAIR
      X(I)=XPAIR(I)*XAIR
  500   CONTINUE
      I=KA
      DO 510 K=1,NABL
      X(I)=XPABL(K)*XABL
      I=I+1
  510   CONTINUE
  511 NTOTAL = NABL + NAIR
      DATA JJJ/3H101/
      GO TO (600,560,560,400) , ICALC
C       PRINT-OUT
  400 GO TO (520,530), I101
  520 WRITE (22,3987) IZERO,NTOTAL,IZERO
 3986 FORMAT (1H13I2)
 3987 FORMAT (3I2)
      I101 =2
      J = 91
      GO TO 540
  530 J=90
      JJJ = IJJJ
      WRITE (22,7961) JJJ, (ZERO,I=1,5)
 7961 FORMAT (A3,5E12.5)
      WRITE (22,3987) IZERO,NTOTAL ,IZERO
  540 DO 550 I=1,NTOTAL
  550 WRITE (22,4163) ID(I), FORM(I), X(I)
 4163 FORMAT(I2,A4,E12.5)
      WRITE (22,4192) J,DENS,TTEMP
 4192 FORMAT (I3,5E12.5)
      RETURN
  560 J=70
      VRTO = DENS
      TTEMP(2) = P
  600 RETURN
      END
```

(2) SEGMENT EIGHT

```
      SUBROUTINE INPUT (IREAD,F,RHOD,P,TTEMP,U,KODEW)
      DIMENSION  XPAIR(3),EMAIR(3),ID(15),XPABL(15),WT1(15),WT2(15)
      DIMENSION TTEMP(4),ZERO(5),ID2S(15),X(15),FORM(15),WT2S(15),
      DIMENSION    XWABL(15),EMABL(15),FORM1(15),FORM2(15),
     1             ID1(15),ID2(15),EMAB1(15),EMAB2(15),XFAB1(15),
     2             XFAB2(15),EMAB2S(15)
      DATA JJJ/3H101/
      DATA INTAP,IOTAP,IZERO/5,6,0/,I101/1/
      DATA ZERO/0.0,0.0,0.0,0.0,0.0,0.0,0.0/
    4 FORMAT(2X,A4,I4,2F10.6)
      IF (IREAD.EQ.1) READ(5,798) NAIR,NABL1,NABL3,KODEW,WABL1,WABL2
  798 FORMAT(13X,I2,3I5/10X,2E15.0)
      NABL2 = NABL3
      IF(IREAD.EQ.0) GO TO 205
C
C     READ ATMOSPHERE MODEL
C
      DO 110 I=1,NAIR
      READ(INTAP,4) FORM(I),ID(I),XPAIR(I),EMAIR(I)
  110 CONTINUE
      KI=NAIR
      KA=NAIR+1
      WAIR=0.0
      DO 200 I=1,KI
      WAIR=WAIR+XPAIR(I)*EMAIR(I) /100.
  200 CONTINUE
  205 CONTINUE
C      MOLECULAR WEIGHT OF ABLATOR
      IF(NABL2.EQ.0) GO TO 260
      IF (IREAD.EQ.0) GO TO 960
C      TWO-ABLATOR MIXTURE
      DO 916 I=1,NABL1
      READ (INTAP,4) FORM1(I),ID1(I), XFAB1(I),EMAB1(I)
  916 CONTINUE
      DO 917 I =1,NABL2
      READ (INTAP,4) FORM2(I),ID2(I), XFAB2(I),EMAB2(I)
      ID2S(I) = ID2(I)
      WT2S(I) = WT2(I)
      EMAB2S(I) = EMAB2(I)
  917 CONTINUE
      IREAD =0
C     ELEMENTAL WEIGHTS IN ABLATOR
  960 DO 967 I= 1,NABL2
      WT2(I) = WT2S(I)
      EMAB2(I) = EMAB2S(I)
  967 ID2(I) = ID2S(I)
      IF (KODEW) 1000,1005,1000
 1000 CALL FCALC(U,F1,F2)
      GO TO 1010
 1005 F1 = WABL1
      F2 = WABL2
 1010 DO 1012 I= 1,NABL1
      WT1(I) = XFAB1(I) * F1
 1012 CONTINUE
      DO 1014 I = 1,NABL2
      WT2(I) = XFAB2(I) * F2
 1014 CONTINUE
```

```
C       MATCH ELEMENTS OF WABL2 WITH WABL1
        DO 1030 I=1,NABL1
        DO 1015 J=1,NABL2
        IF(ID1(I)-ID2(J))1015,1020,1015
 1020   NABL2=NABL2-1
        WT1(I)=WT2(J)+WT1(I)
        DO 1025 K=J,NABL2
        ID2(K)=ID2(K+1)
        WT2(K)=WT2(K+1)
        EMAB2(K)=EMAB2(K+1)
        FORM2(K)=FORM2(K+1)
 1025   CONTINUE
        GO TO 1030
 1015   CONTINUE
 1030   CONTINUE
        J=KA
C       MAKE INTO ONE ABLATOR
        DO 1040 I=1,NABL1
        FORM(J)=FORM1(I)
        ID(J)=ID1(I)
        EMABL(I)=EMAB1(I)
        XWABL(I)=WT1(I)
        J=J+1
 1040   CONTINUE
        K=NABL1
        IF (NABL2.EQ.0) GO TO 1051
        DO 1050 I=1,NABL2
        K=K+1
        FORM(J)=FORM2(I)
        ID(J)=ID2(I)
        EMABL(K)=EMAB2(I)
        XWABL(K)=WT2(I)
        J=J+1
 1050   CONTINUE
 1051   NABL = K
        GO TO 1060
C       SINGLE ABLATOR
  260   NABL=NABL1
        K=KA
        IF (IREAD.EQ.0) GO TO 270
        DO 717 I = 1,NABL
        READ (INTAP,4) FORM(K),ID(K),XWABL(I),EMABL(I)
        K = K+1
  717   CONTINUE
        IREAD =0
  270   K = KA + NABL
 1060   SUMXW=0.0
        DO 310 I=1,NABL
        SUMXW=SUMXW+XWABL(I)/EMABL(I)
  310   CONTINUE
        WABL=0.0
        DO 320 I=1,NABL
        XPABL(I)=XWABL(I)/EMABL(I)/SUMXW
        WABL=WABL+XPABL(I)*EMABL(I)
  320   CONTINUE
        DENS = RHOD/1.29232
C       MOLES OF MIXTURE
        EMMIX=100.*(1.0-F)/WAIR+100.*F/WABL
        XAIR=100.*(1.0-F)/WAIR/EMMIX
        XABL=100.*F/WABL/EMMIX
        DO 500 I=1,KI
        X(I)=XPAIR(I)*XAIR/100.
```

```
  500   CONTINUE
        I=KA
        DO 510 K=1,NABL
        X(I)=XPABL(K)*XABL
        I=I+1
  510   CONTINUE
        NTOTAL = NABL + KI
C       PRINT-OUT
        GO TO (520,530), I101
  520 WRITE (22,3987) IZERO,NTOTAL,IZERO
 3987 FORMAT (3I2)
        I101 =2
        J = 91
        GO TO 540
  530 J=90
        WRITE (22,7961) JJJ, (ZERO,I=1,5)
 7961 FORMAT (A3,5E12.5)
        WRITE (22,3987) IZERO,NTOTAL ,IZERO
  540 DO 550 I=1,NTOTAL
        WRITE (22,4163) ID(I), FORM(I), X(I)
  550 CONTINUE
 4163 FORMAT(I2,A4,E12.5)
        WRITE (22,4192) J,DENS,TTEMP
 4192 FORMAT (I3,5E12.5)
        RETURN
        END
```

16. SUBROUTINE FCALC

a. Function

This subroutine is included to permit the user to vary the ratio of material one to material two through the boundary layer. If this subroutine is not called, the ratio remains constant at the amount of ablation gas from the forward surface material to the amount of the second material from the end of the forward surface to the desired station. The fraction of each material F1 and F2 is a function of the velocity ratio. No specific coding has been included with this subroutine. (See Appendix.)

b. Listing

```
SUBROUTINE FCALC (U,F1,F2)
F1 = 1.399975E-1
F2 = 5.299911E-2
RETURN
END
```

17. SUBROUTINE ISNTRP

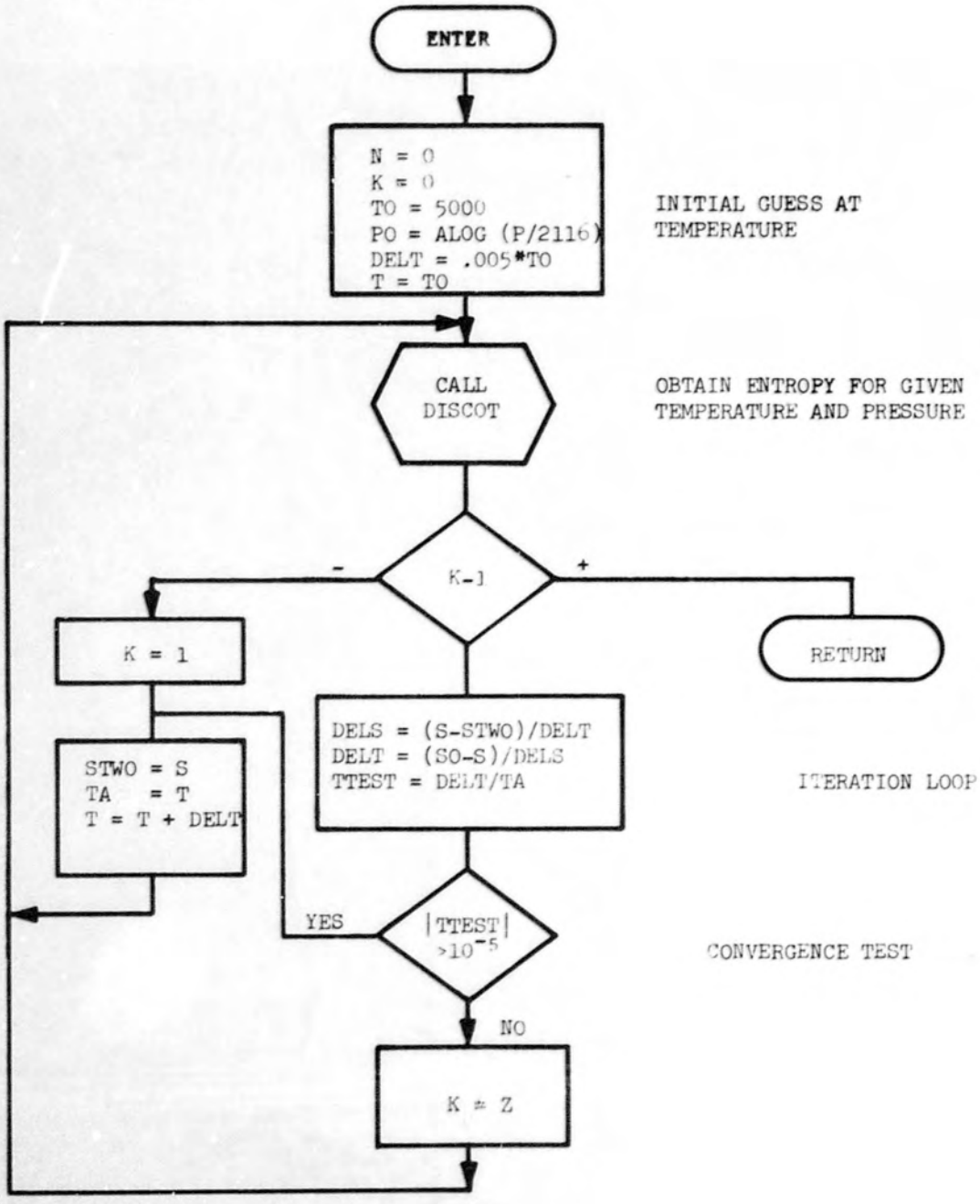
a. Function

This subroutine iterates on a temperature which satisfies a given entropy and pressure. The subroutine uses the table look-up routine DISCOT on entropy data obtained from HANSEN, and compiled into the program in block ENTRPY. The entropy is a function of temperature and pressure.

b. Symbols

Symbolic representation	Meaning or use
DELS	ds/dt
DELT	Δt
ENTRO	Entropy array
K	Program trigger
N	Not used
P	Pressure
PO	Log_{10} of pressure in atmospheres
PRESS	Pressure array
S	Entropy
SCT	Common array used by subroutine DISCOT
SO	Entropy in calling sequences
STWO	Entropy
T	Temperature in calling sequence
TA	Temperature
TEMP	Temperature array
TO	Temperature = 5000
TTEST	Δt/temperature

c. Flow Chart


ENTER
N = 0
K = 0
TO = 5000
PO = ALOG (P/2116)
DELT = .005*TO
T = TO
INITIAL GUESS AT TEMPERATURE
CALL DISCOT
OBTAIN ENTROPY FOR GIVEN TEMPERATURE AND PRESSURE
K-1
-
+
K = 1
RETURN
STWO = S
TA = T
T = T + DELT
DELS = (S-STWO)/DELT
DELT = (SO-S)/DELS
TTEST = DELT/TA
ITERATION LOOP
YES
|TTEST| >10^-5
CONVERGENCE TEST
NO
K = Z

D. LISTING

```
      SUBROUTINE ISNTRP(P,SO,T)
C
C     TO DETERMINE TEMPERATURE FOR GIVEN ENTROPY
C     AND PRESSURE
C
      COMMON/ENTRPY/TEMP,PRESS,ENTRO
      DIMENSION TEMP(31),PRESS(7),ENTRO(217)
      N=0
      K=0
      TO=5000.0
      PO=ALOG10(P/2116.18)
      DELT=0.005*TO
      T=TO
    1 CALL DISCOT(T,PO,TEMP,ENTRO,PRESS,11,217,7,5,31)
      S=1716.0*S
      IF(K-1)2,4,6
    2 K=1
    3 STWO=S
      TA=T
      T=T+DELT
      GO TO 1
    4 DELS=(S-STWO)/DELT
      DELT=(SO-S)/DELS
      TTEST=DELT/TA
      IF(ABS(TTEST)-1.0E-05)5,5,3
    5 K=2
      GO TO 1
    6 RETURN
      END
```

E. BLOCK DATA

```
      BLOCK DATA
      DIMENSION T(31),P(7),S(217)
      COMMON/ENTRPY/ T,P,S
      DATA (S(I),I=1,124) /
     1  27.0,  34.9,  37.5,  39.2,  40.9,  45.9,  47.5,  48.6,  51.5,
     2  60.0,  68.9,  71.3,  72.2,  73.4,  75.4,  79.0,  85.0,  94.1,
     3 104.3, 112.2, 116.6, 118.8, 119.9, 120.7, 121.3, 121.8, 122.3,
     4 122.7, 123.1, 123.5, 123.9,  24.7,  32.6,  35.2,  36.8,  38.2,
     5  41.4,  44.6,  45.6,  46.9,  50.5,  58.0,  64.8,  67.1,  68.1,
     6  69.1,  70.5,  72.8,  76.4,  81.6,  88.6,  96.2, 102.7, 107.1,
     7 109.6, 111.1, 112.1, 112.8, 113.3, 113.8, 114.2, 114.6,  22.4,
     8  30.3,  32.9,  34.5,  35.8,  37.6,  41.0,  42.7,  43.6,  45.1,
     9  48.4,  54.3,  59.9,  62.6,  63.8,  64.6,  65.7,  67.1,  69.2,
     *  72.0,  75.8,  80.6,  86.0,  91.4,  95.8,  99.1, 101.3, 102.5,
     *  103.8, 104.5, 105.1,  20.1,  28.0,  30.6,  32.2,  33.5,  34.7,
     *  36.9,  39.4,  40.6,  41.5,  42.9,  45.5,  49.7,  54.4,  57.6,
     *  59.2,  60.1,  61.0,  61.9,  63.1,  64.6,  66.6,  69.0,  71.9,
     *  75.4,  79.1,  82.9,  86.4,  89.4,  91.7,  93.4/
      DATA (S(I),I=125,217) /
     1  17.7,  25.7,  28.3,  29.9,  31.2,  32.2,  33.5,  35.5,  37.3,
     2  38.3,  39.2,  40.4,  42.2,  44.9,  48.3,  51.6,  53.9,  55.3,
     3  56.2,  57.0,  57.8,  58.7,  59.7,  60.9,  62.4,  64.0,  66.0,
     4  68.2,  70.5,  73.1,  75.6,  15.5,  23.4,  26.0,  27.6,  28.9,
     5  29.9,  30.8,  32.0,  33.5,  34.9,  36.0,  36.8,  37.7,  38.9,
     6  40.5,  42.7,  45.1,  47.6,  49.5,  51.0,  52.0,  52.8,  53.4,
     7  54.1,  54.8,  55.6,  56.4,  57.3,  58.4,  59.5,  60.8,  13.1,
     8  21.1,  23.7,  25.3,  26.6,  27.5,  28.4,  29.3,  30.2,  31.3,
     9  32.4,  33.4,  34.2,  34.9,  35.8,  36.8,  38.0,  39.4,  41.1,
     *  42.8,  44.5,  45.9,  47.1,  48.0,  48.8,  49.4,  50.0,  50.5,
     *  51.1,  51.7,  52.2/
      DATA P/
     1  -4.0,  -3.0,  -2.0,  -1.0,   0.0,   1.0,   2.0/
      DATA T/
     1          0.0,        900.0,       1800.0,       2700.0,
     2       3600.0,       4500.0,       5400.0,       6300.0,
     3       7200.0,       8100.0,       9000.0,       9900.0,
     4      10800.0,      11700.0,      12600.0,      13500.0,
     5      14400.0,      15300.0,      16200.0,      17100.0,
     6      18000.0,      18900.0,      19800.0,      20700.0,
     7      21600.0,      22500.0,      23400.0,      24300.0,
     8      25200.0,      26100.0,      27000.0/
      END
```

18. SUBROUTINE M367

a. General

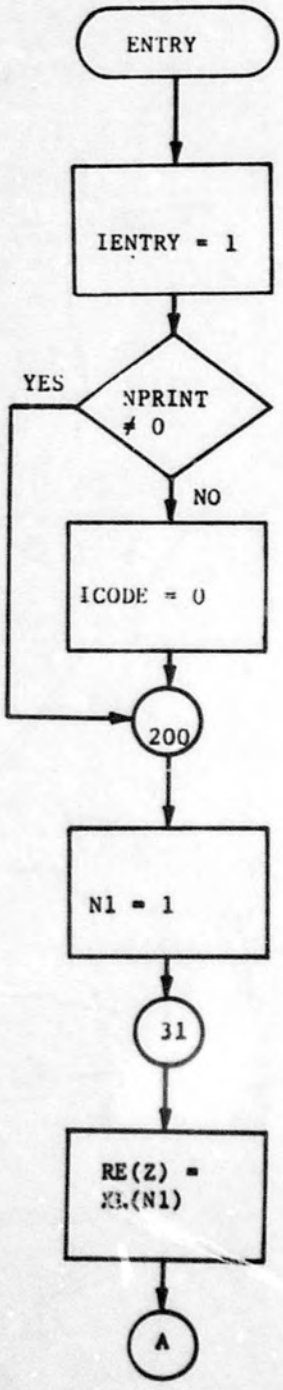
Subroutine M367 is used to provide the total mass loss rate and local mass loss rate for the vehicle under consideration. Other information such as the trajectory point or flow conditions can be obtained by using the print option, or from the common block.

Certain data for this routine is transmitted through common block /367/. Since the subroutine is used by different programs (Refs. 1, 5), the actual common block locations will change; however, the locations of items relative to each other remain fixed.

Subroutine M367 is also used in the calculations for a multiple ablator vehicle. It calls subroutine ABMSLS 2N-1 times for a vehicle constructed of N different ablators (maximum of five ablators).

As was mentioned previously, the complete theory for the mass loss calculations is not discussed in this report. The details of theory and numerical method are discussed in Volume II of this series (Ref. 1). However, the flow charts, listings, and block data required for the boundary layer solution are provided in this volume.

b. Flow Chart



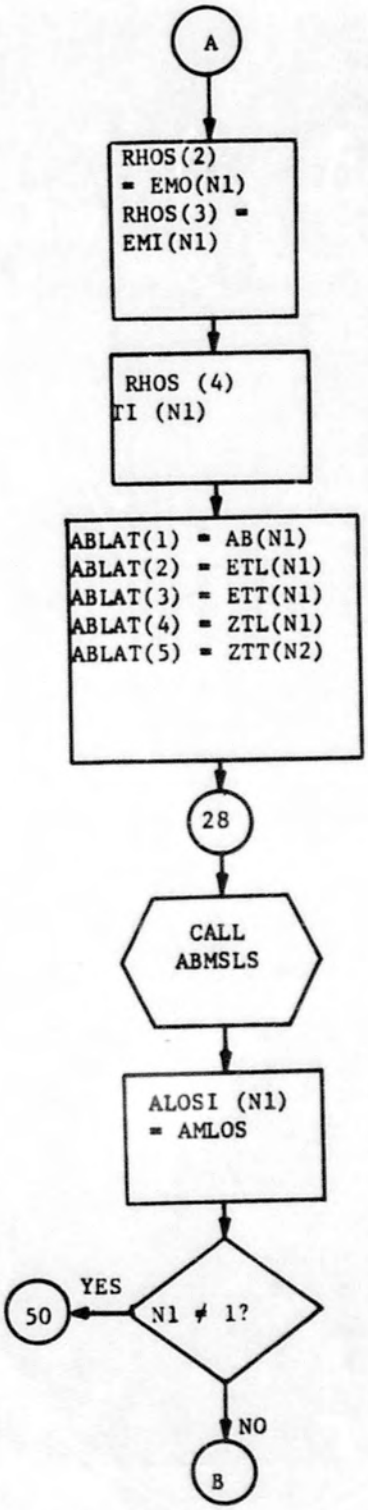
SET GEOMETRY INDICATOR TO DO THE GEOMETRIC CALCULATIONS

DETERMINE IF ALL PRINTOUT SHOULD BE SURPRESSED

SET CODE TO SURPRESS PRINTOUT

SET UP THE ABLATOR INDEX FOR THE FIRST ABLATOR

SET UP THE DISTANCE TO THE BACK OF THE ABLATOR



SET UP CONSTANTS USED TO CALCULATE THE EMISSIVITY FOR THIS ABLATOR

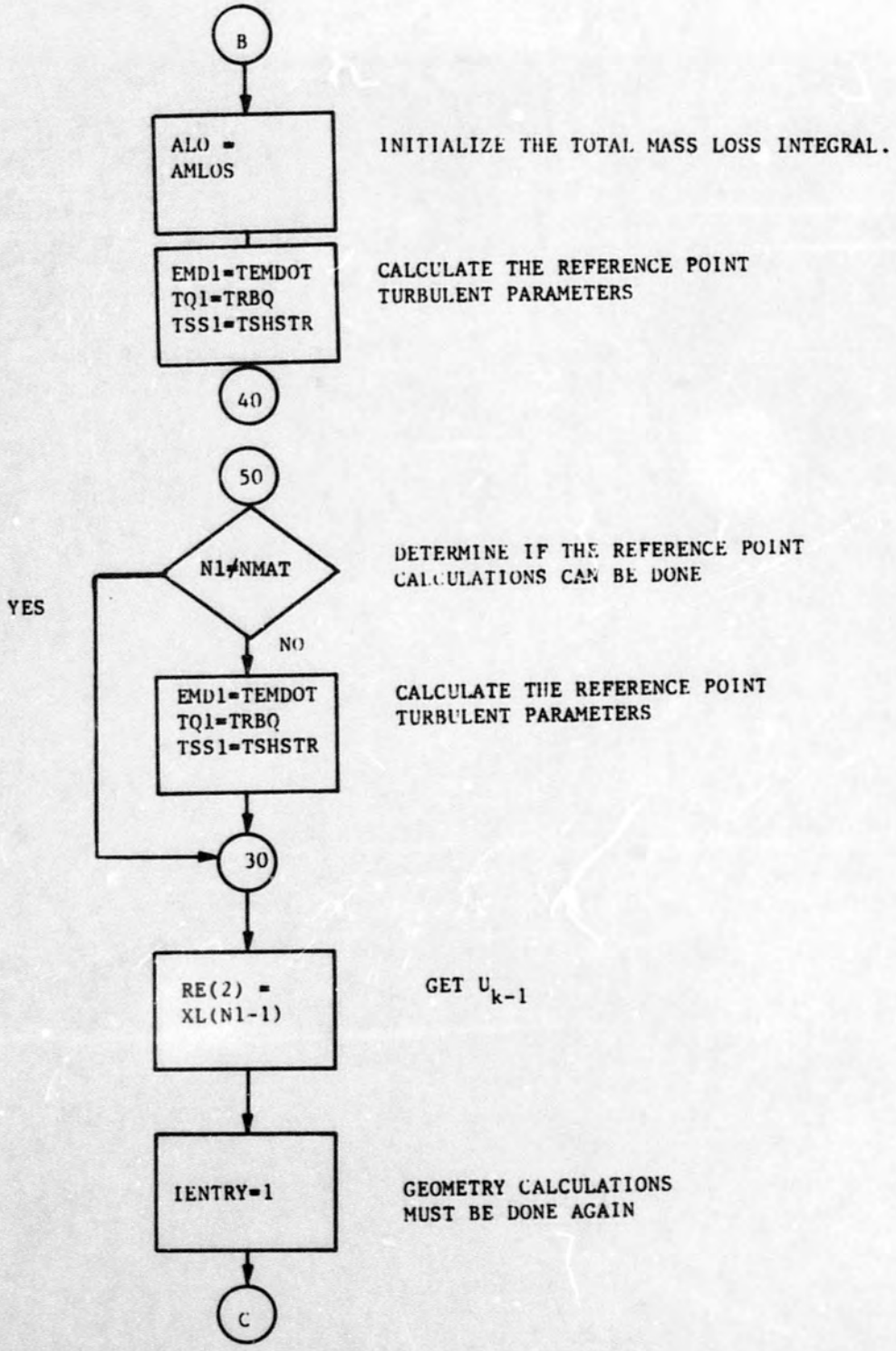
SET UP THE ABLATION TEMPERATURE FOR ABLATOR N1

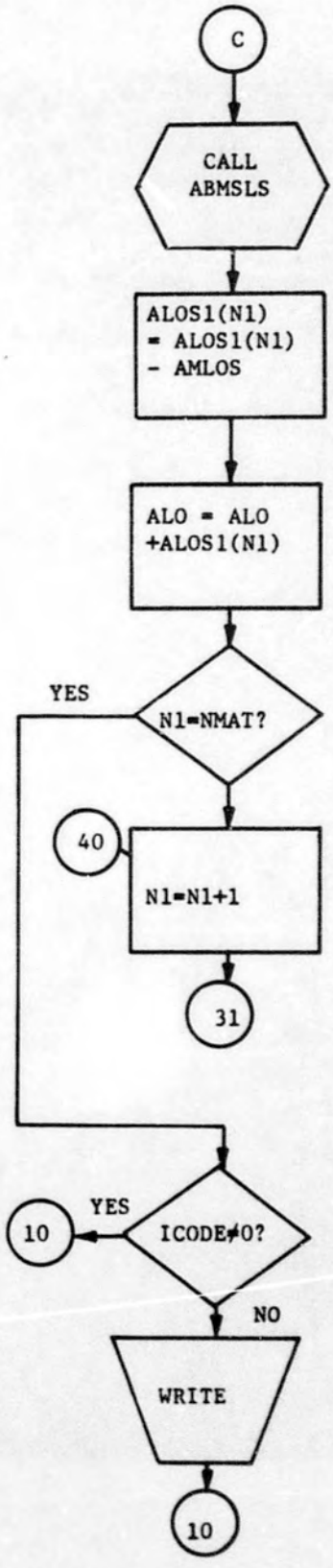
SET UP ABLATION MODEL CONSTANTS FOR ABLATOR N1

GET VALUE OF $\int_0^{u_k}$

$$\left(\frac{dM}{dt}\right)_K = \int_0^{u_k} \frac{d^2M}{dAdt} dA$$

CHECK FOR THE FIRST ABLATOR


B
ALO = AMLOS
INITIALIZE THE TOTAL MASS LOSS INTEGRAL.
EMD1=TEMDOT
TQ1=TRBQ
TSS1=TSHSTR
CALCULATE THE REFERENCE POINT TURBULENT PARAMETERS
40
50
N1≠NMAT
DETERMINE IF THE REFERENCE POINT CALCULATIONS CAN BE DONE
YES
NO
EMD1=TEMDOT
TQ1=TRBQ
TSS1=TSHSTR
CALCULATE THE REFERENCE POINT TURBULENT PARAMETERS
30
RE(2) = XL(N1-1)
GET U_{k-1}
IENTRY=1
GEOMETRY CALCULATIONS MUST BE DONE AGAIN
C



GET $\int_0^{u_{k-1}}$

GET $\int_0^{u_k} - \int_0^{u_{k-1}}$

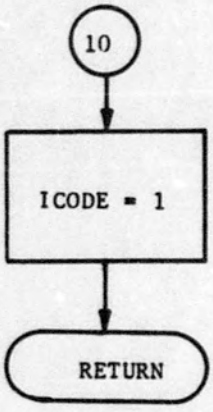
ADD THIS ABLATORS CONTRIBUTION TO THE TOTAL MASS LOSS

LAST ABLATOR?

INCREMENT FOR LOOP INDEX

DETERMINE IF PRINTOUT IS REQUIRED

PRINTOUT REFERENCE POINT AND MASS LOSS PARAMETERS



RESET THE PRINT CODE
TO THE NO PRINT POSITION

C. LISTING

```
      SUBROUTINE M367 (ALT,VO,ALO,TSHSTR,EMD1)
      COMMON /367/ AM367(150)
      DIMENSION      XL    (  5),     CO    (  5),     CI    (  5),                  15
     *               RO    (  5),     EMO   (  5),     EMI   (  5),                  30
     *               TI    (  5),     AB    (  5),     ETL   (  5),                  45
     *               ETT   (  5),     ZTL   (  5),     ZTT   (  5),                  60
     *CPO   (   3),    RHOS (  4),
     *               ABLAT(  5),     TITLA(  9),     TITLB(  9),                  490
     *               TITLC(  9),     KASE (  5),     MVSV (  6),                  510
     *               RE    (  3),     ALOS1(  5)
      EQUIVALENCE   (AM367(  1), PAGE    ),(AM367(  2), ICODE     )                   2
     *,             (AM367(  3),TITLA(1)),(AM367( 12), TITLB(1) )                  20
     *,             (AM367( 21),TITLC(1)),(AM367( 30), IRETML   )
     *,             (AM367( 35),NOK     ),(AM367( 41), I20      )
     *,             (AM367( 42),I22     ),(AM367( 43), NMAT     )                  43
     *,             (AM367( 44),RAD     ),(AM367( 45), RETMLC   )                  45
     *,             (AM367( 46),DTAU    ),(AM367( 47), TR       )                  47
     *,             (AM367( 48),RE(1)   ),(AM367( 51), DELS     )                  51
     *,             (AM367( 52),XL(1)   ),(AM367( 57),CO(1)     )
     *,             (AM367( 62),CI(1)   ),(AM367( 67),RO(1)     )
     *,             (AM367( 72),EMO(1)  ),(AM367( 77),EMI(1)    )
     *,             (AM367( 82),TI(1)   ),(AM367( 87),AB(1)     )
     *,             (AM367( 92),ETL(1)  ),(AM367( 97),ETT(1)    )
     *,             (AM367(102),ZTL(1)  ),(AM367(107),ZTT(1)    )
     *,             (AM367(112),CPO(1)  ),(AM367(115),RHOS(1)   )
     *,             (AM367(119),ABLAT(1)),(AM367(124),NPRNT(1)  )
     *,             (AM367(125),TIME    )
      EQUIVALENCE (AM367(126),TRAD),(AM367(131),AMODEL),
     1            (AM367(132),TRCUT)
      EQUIVALENCE(AM367(31),K1)
      DIMENSION TRAD(5)
      COMMON /716/ TITLE(5),DUMMY(23),WABL1,WABL2,D3(257),TSS1,D4(5)
32113 FORMAT (1H0,11HMASS LOSS =,E14.6,7H LB/SEC,9X,9H EMDOT = E14.6,/
     *12H TURB. Q   =E14.6,16X,22H TURB. SHEAR STRESS = E14.6)
      IENTRY = 1
      IF(NPRNT.NE.0) GO TO 200
      ICODE=0
200   CONTINUE
      DO 40 N1=1,NMAT
   31 RE(2)=XL(N1)
      RHOS(2)=EMO(N1)
      RHOS(3)=EMI(N1)
      RHOS(4)=TI(N1)
      IF(K1.EQ.6) GO TO 28
      TRCUT = TRAD(N1)
      ABLAT(1)=AB(N1)
      ABLAT(2)=ETL(N1)
      ABLAT(3)=ETT(N1)
      ABLAT(4)=ZTL(N1)
      ABLAT(5)=ZTT(N1)
   28 CALL ABMSLS(IENTRY,ALT,VO,AMLOS,TEMDOT,TRBQ,TSHSTR,ICODE)
      IF(K1.EQ.6) GO TO 29
      ALOS1(N1)=AMLOS
      IF(N1.NE.1) GO TO 50
      ALO=AMLOS
      EMD1=TEMDOT
      TQ1=TRBQ
```

```
   20 CONTINUE
      TSS1=TSHSTR
      IF(K1.EQ.6) CALL CHAIN(ONE)
      GO TO 40
   50 CONTINUE
      IF(N1.NE.NMAT) GO TO 30
      EMD1=TEMDOT
      TQ1=TRBQ
      TSS1=TSHSTR
   30 RE(2)=XL(N1-1)
      IENTRY=1
      CALL ABMSLS(IENTRY,ALT,VC,AMLOS,TEMDOT,TRBQ,TSHSTR,ICODE)
      ALOS1(N1)=ALOS1(N1)-AMLOS
      ALO=ALO+ALOS1(N1)
   40 CONTINUE
      WABL1 = ALOS1(1)
      WABL2 = ALOS1(2)
      IF(ICODE.NE.0) GO TO 10
      WRITE (6,100) (AL(IX), ALOS1(IX),IX=1,NMAT)
  100 FORMAT(15H0      MAXIMUM XE15.6,15H       MASS LOSSE15.6)
      WRITE (6,32113)ALO,EMD1,TQ1,TSS1
   10 CONTINUE
      ICODE=1
      RETURN
      END
```

19. SUBROUTINE ABMSLS

a. General

Subroutine ABMSLS does geometry, flow environment, heat transfer coefficient, heating rate, mass loss rate, and turbulent shearing stress calculations for a single ablator vehicle. In the integrals of equation (49) of reference 1, the radiation cutoff and laminar-to-turbulent transition point checks must be made as these effects change the integrands. The integrals are therefore calculated by subroutine ABMSLS. Subroutine ABMSLS does all integrand checking.

Since most of the calculations performed by Subroutine ABMSLS are a straightforward implementation of the theory in reference 1, only a correlation between the program and the theory section is given here. The following outline shows the program calculations:

(1) Initialize reference point variables

(2) Optionally calculate geometric parameters

(3) Calculate free stream and local parameters

(4) Calculate radiation parameters

(5) Optionally print out the above results

(6) A DO LOOP is set up to perform the calculations that are different for laminar and turbulent flow. The first pass does the laminar calculations, and the second pass does the turbulent calculation.

(7) Calculate Eckert's reference conditions

(8) On the first pass calculate the flow transition point

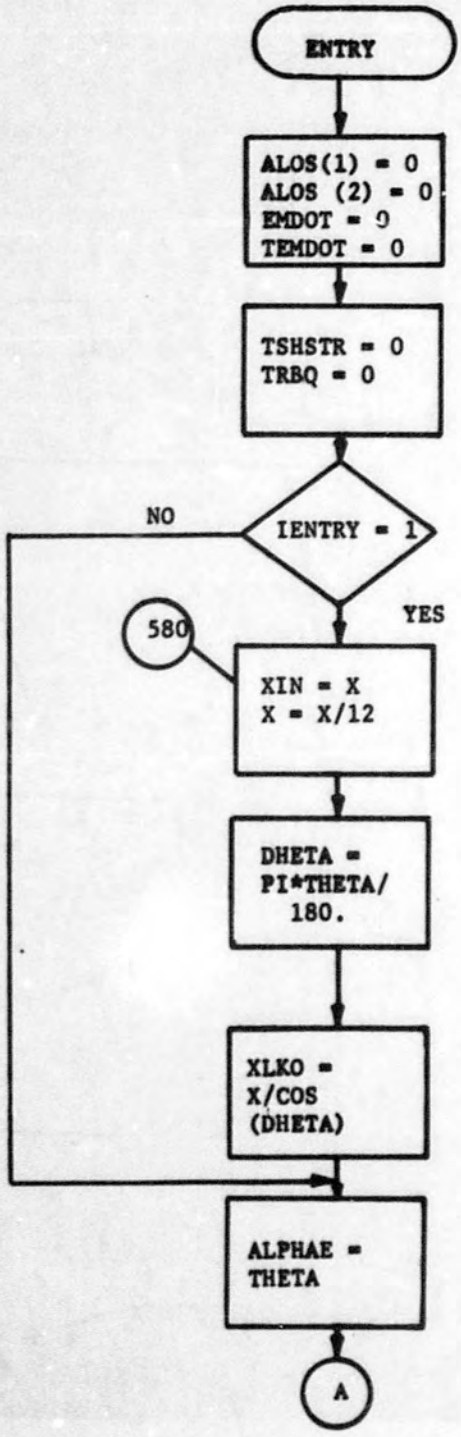
(9) Set up the integration limits

(10) Calculate mass losses

(11) Optionally print out laminar and turbulent results

(12) Calculate the reference point conditions

b. Flow Chart



INITIALIZE PARAMETERS

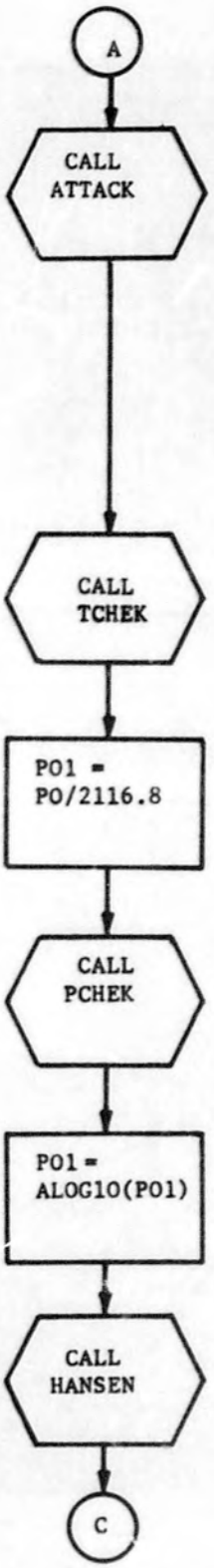
DO THE GEOMETRIC CALCULATIONS

SAVE X IN INCHES. CONVERT THE LENGTH TO FEET

CONVERT THE CONE HALF ANGLE TO RADIANS

CALCULATE THE REFERENCE POINT

SET UP THE EFFECTIVE ANGLE-OF-ATTACK



GO INTO ATTACK WITH THE ALTITUDE, EFFECTIVE ANGLE-OF-ATTACK, AND THE VEHICLE SPEED. COME OUT WITH THE FREE-STREAM PRESSURE AND TEMPERATURE, THE LOCAL TEMPERATURE, PRESSURE, AND FLOW SPEED. ALSO OBTAIN THE RATIOS OF LOCAL-TO-FREE-STREAM VALUES AND THE FREE-STREAM MACH NUMBER

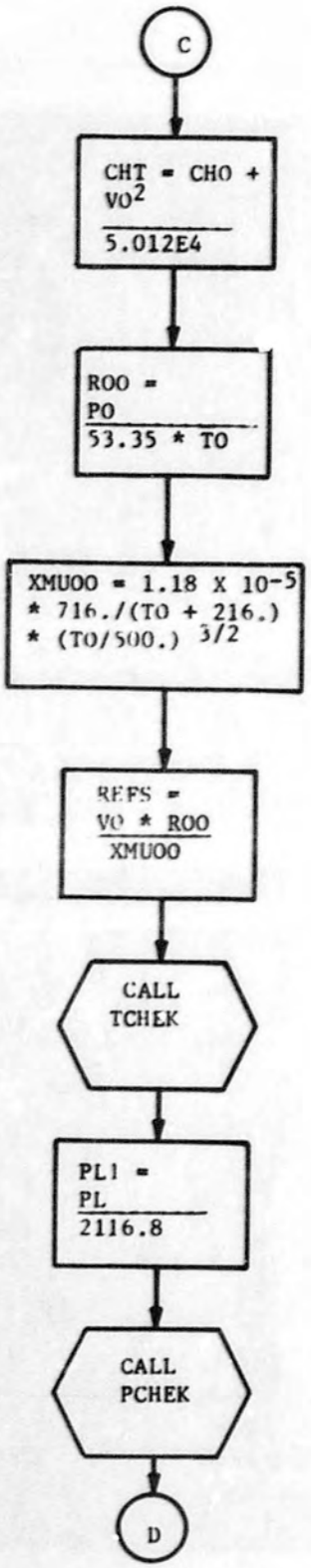
RANGE CHECK THE FRLE STREAM TEMPERATURE

CONVERT THE FREE-STREAM PRESSURE TO ATMOSPHERES

RANGE CHECK THE FREE-STREAM PRESSURE

SETUP THE FREE-STREAM PRESSURE FOR HANSEN TABLES

GET THE ENTHALPY OF FREE-STREAM AIR



CALCULATE THE TOTAL FREE-STREAM ENTHALPY (SPECIFIC ENTHALPY)

CALCULATE THE FREE-STREAM SPECIFIC WEIGHT

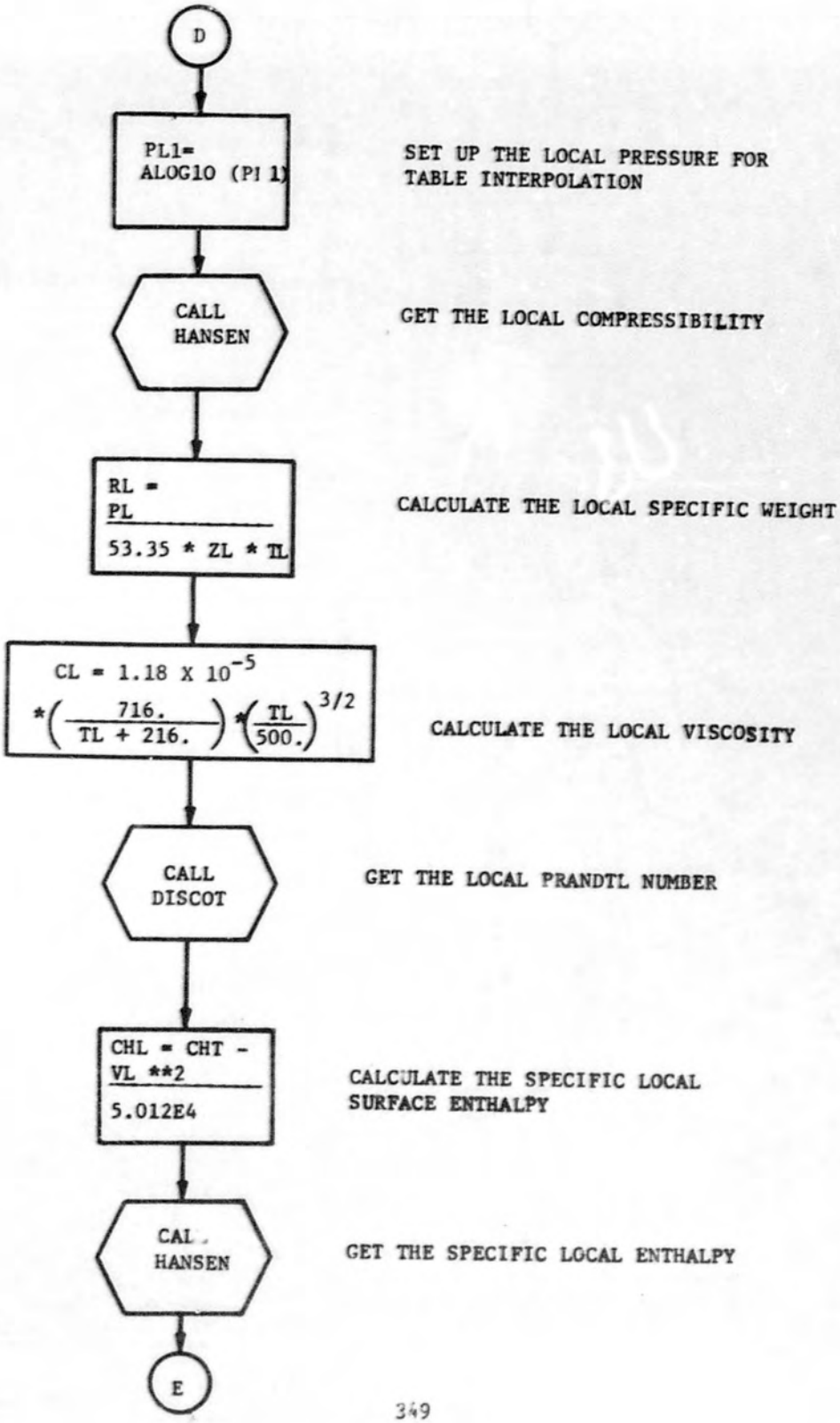
CALCULATE THE FREE-STREAM VISCOSITY

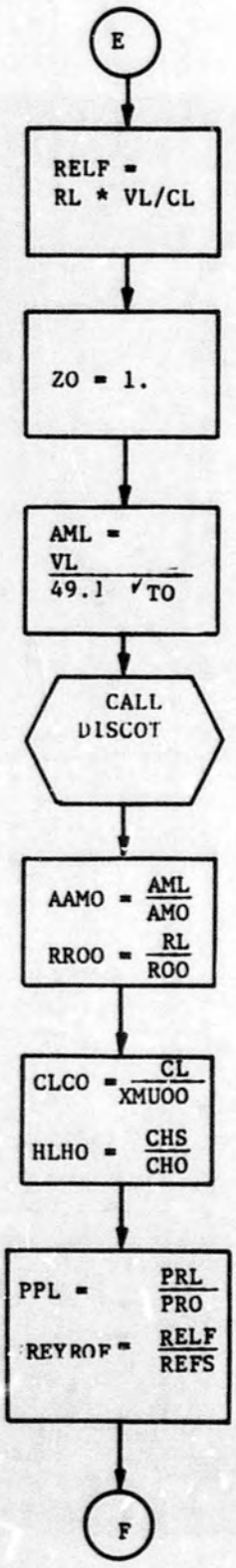
CALCULATE THE FREE-STREAM REYNOLD'S NUMBER PER FOOT

CHECK THE LOCAL TEMPERATURE

CONVERT THE LOCAL PRESSURE TO ATMOSPHERES

CHECK THE LOCAL PRESSURE


D
PL1=
ALOG10 (PI 1)
SET UP THE LOCAL PRESSURE FOR TABLE INTERPOLATION
CALL
HANSEN
GET THE LOCAL COMPRESSIBILITY
RL =
PL
53.35 * ZL * TL
CALCULATE THE LOCAL SPECIFIC WEIGHT
CL = 1.18 X 10^-5
*(716. / TL + 216.) *(TL / 500.)^3/2
CALCULATE THE LOCAL VISCOSITY
CALL
DISCOT
GET THE LOCAL PRANDTL NUMBER
CHL = CHT -
VL **2
5.012E4
CALCULATE THE SPECIFIC LOCAL SURFACE ENTHALPY
CAL
HANSEN
GET THE SPECIFIC LOCAL ENTHALPY
E



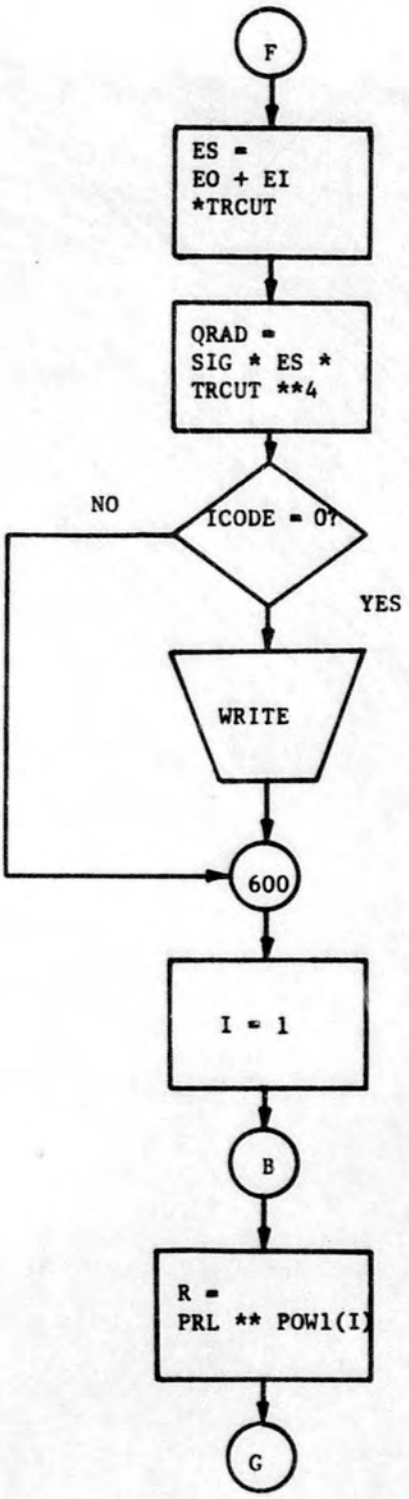
CALCULATE THE LOCAL REYNOLD'S NUMBER PER FOOT

SET THE FREE-STREAM COMPRESSIBILITY FOR THE PRINTOUT

CALCULATE THE LOCAL MACH NUMBER

GET THE LOCAL PRANDTL NUMBER

CALCULATE LOCAL-TO-FREE-STREAM RATIOS FOR PRINTOUT



CALCULATE THE ABLATOR EMISSIVITY CONSTANT

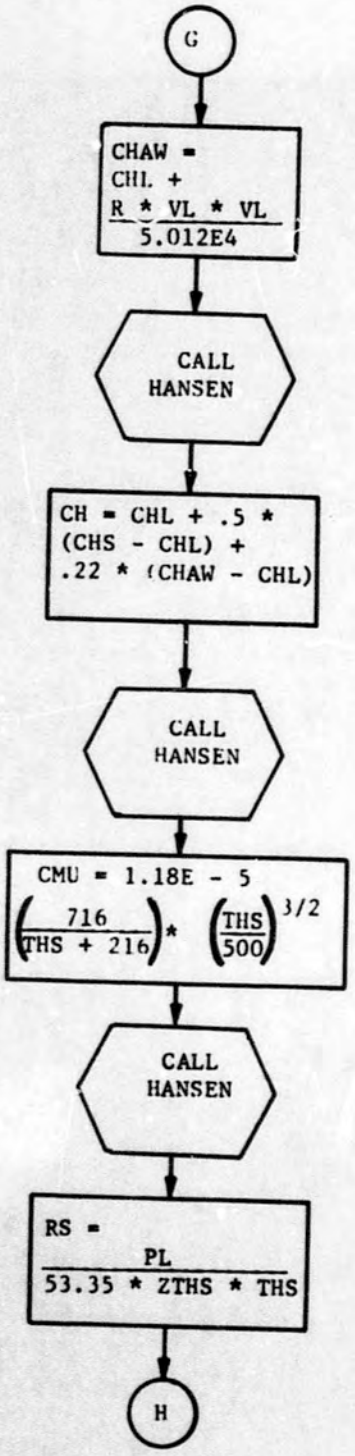
CALCULATE THE RATE PER UNIT AREA THAT THE ABLATOR LOSES ENERGY BY RADIATION

CHECK FOR PRINT OUT OPTION

PRINT OUT ATMOSPHERIC PARAMETERS

SET THE DO LOOP INDEX FOR LAMINAR HEATING

CALCULATE THE ENERGY RECOVERY FACTOR



CALCULATE THE ADIABATIC WALL ENTHALPY

FIND THE ADIABATIC WALL TEMPERATURE

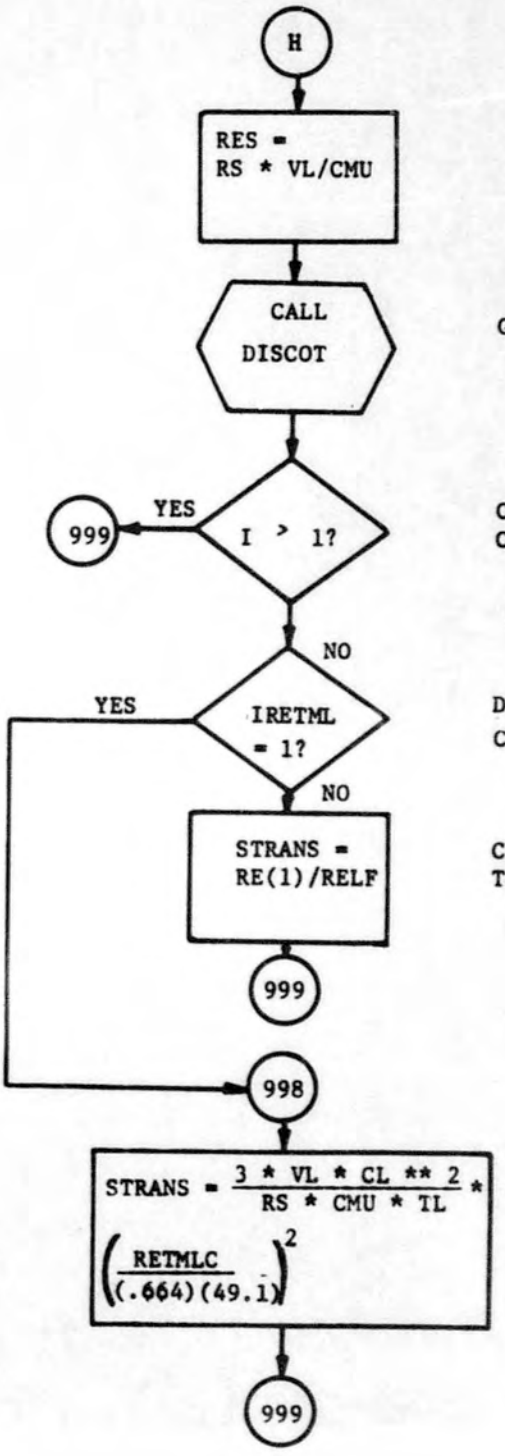
CALCULATE ECKERT'S REFERENCE ENTHALPY

FIND THE REFERENCE TEMPERATURE

CALCULATE THE REFERENCE VISCOSITY

GET THE REFERENCE COMPRESSIBILITY

CALCULATE THE REFERENCE SPECIFIC WEIGHT



CALCULATE THE REFERENCE REYNOLD'S NUMBER PER FOOT

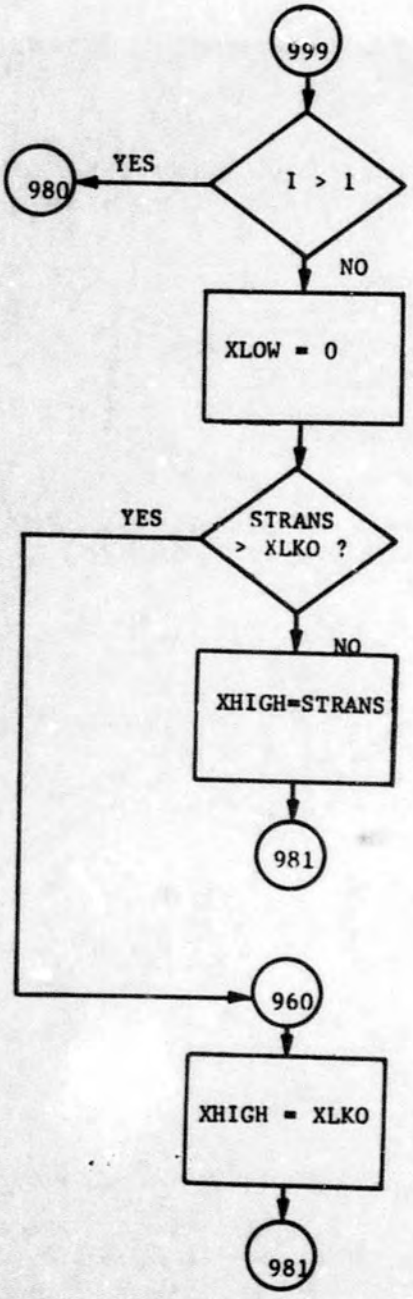
GET THE REFERENCE PRANDTL NUMBER

CHECK TO DETERMINE IF TRANSITION CALCULATIONS ARE TO BE DONE

DETERMINE WHICH TRANSITION CRITERION IS TO BE USED

CRITICAL LOCAL REYNOLD'S NUMBER TRANSITION CRITERION CALCULATION

CRITICAL LOCAL MOMENTUM THICKNESS REYNOLD'S NUMBER TRANSITION CRITERION CALCULATION



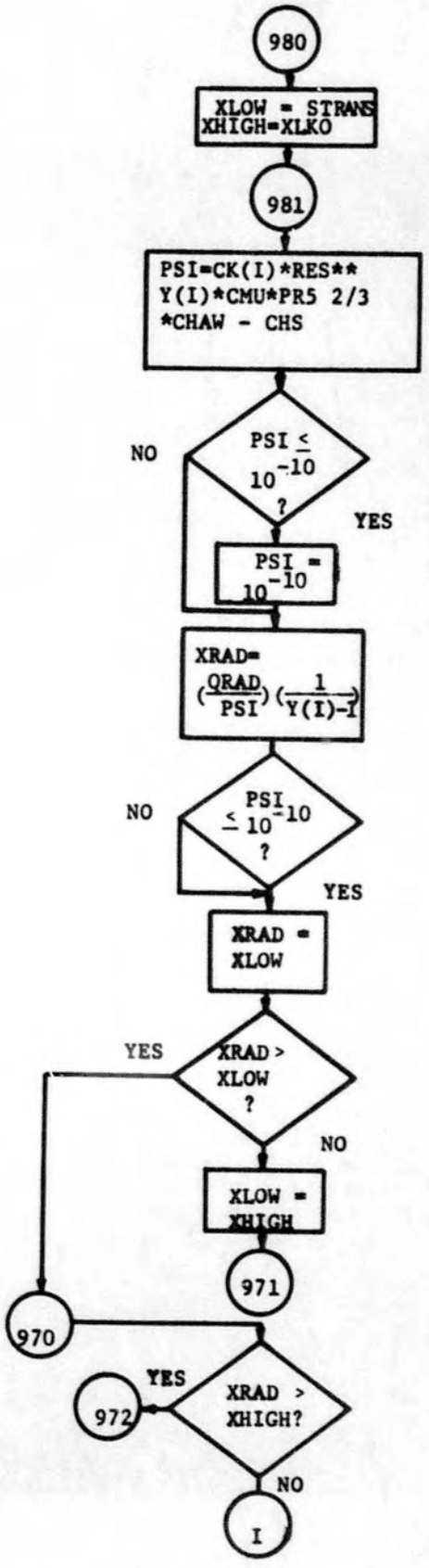
TEST USED TO SET UP LOWER INTEGRATION LIMIT

SET THE LOWER INTEGRATION FOR LAMINAR HEATING

IS THE TRANSITION POINT FURTHER BACK THAN THE REFERENCE POINT?

SET THE LAMINAR UPPER INTEGRATION LIMIT EQUAL TO THE TRANSITION POINT

SET THE UPPER LAMINAR INTEGRATION LIMIT TO THE REFERENCE POINT



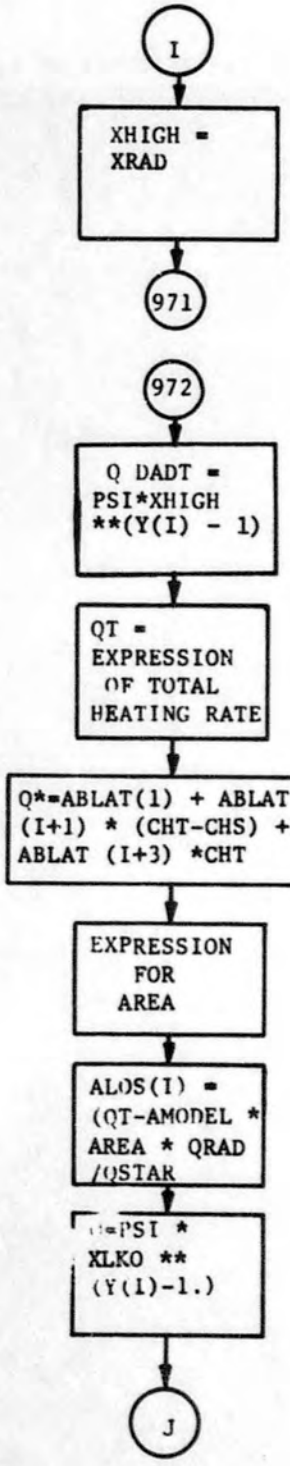
SET THE INTEGRATION LIMITS FOR TURBULENT HEATING

CALCULATE THE RADIATION CUTOFF COORDINATE

DETERMINE IF THE RADIATION CUTOFF IS BELOW THE LOWER INTEGRATION LIMIT

EQUATE INTEGRATION LIMITS TO ASSURE ZERO FOR INTEGRALS

TEST FOR RADIATION CUTOFF ON THE UPPER LIMIT



SET UPPER INTEGRATION LIMIT TO THE RADIATION CUTOFF VALUE

CALCULATE THE HEATING RATE PER UNIT AREA AT THE UPPER INTEGRATION LIMIT

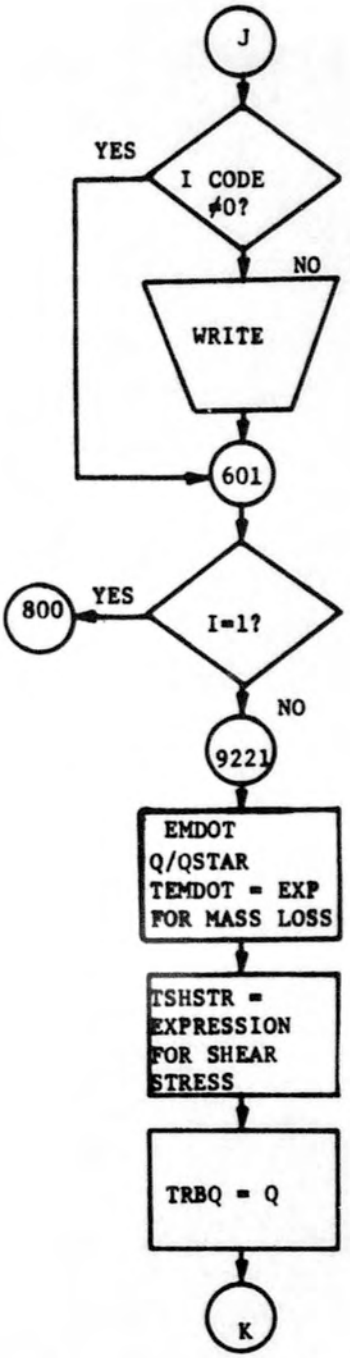
CALCULATE THE TOTAL HEATING RATE

CALCULATE THE ABLATION CONSTANT

CALCULATE THE VEHICLE SURFACE AREA

CALCULATE TOTAL MASS LOSS DUE TO LAMINAR HEATING

CALCULATE THE HEATING RATE PER UNIT AREA AT THE REFERENCE POINT



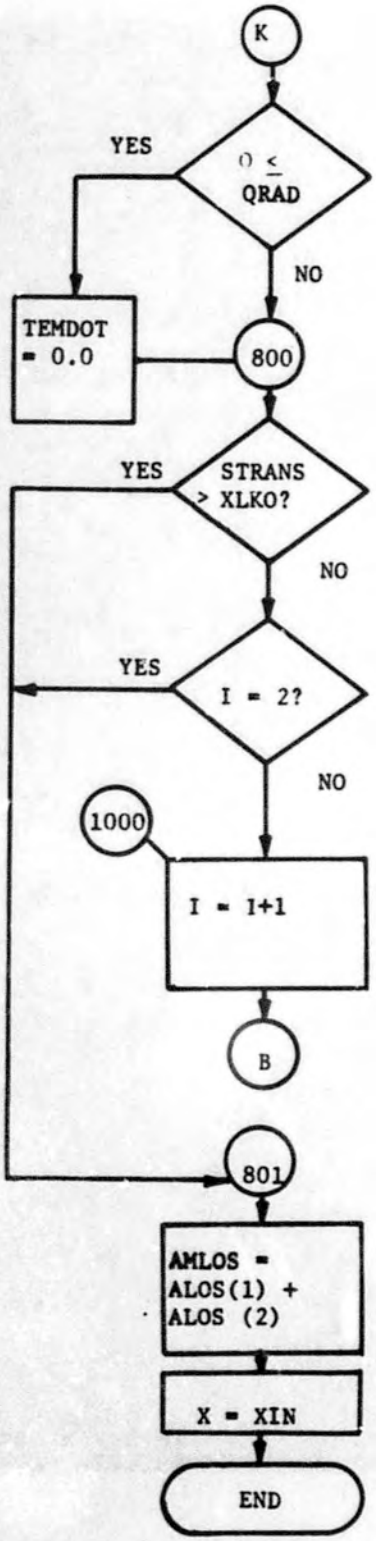
TEST FOR PRINT OPTION

PRINT HEATING RATE AND MASS LOSS PARAMETERS

TEST FOR LAMINAR HEATING

CALCULATE THE TURBULENT MASS LOSS RATE PER UNIT AREA AT THE REFERENCE POINT

CALCULATE THE TURBULENT SHEARING STRESS AT THE REFERENCE POINT



IS ALL THE HEATING LAMINAR?

TEST DO DOOP INDEX

INCREMENT DO INDEX

ADD THE LAMINAR AND TURBULENT MASS LOSS RATE

RESTORE X TO INCHES

```
C.  LISTING

      SUBROUTINE ABMSLS (IENTRY,ALT,VO,AMLOS,TEMDOT,TRBQ,TSHSTR,ICODE)
      DIMENSION XMODE(2)
      DIMENSION POW1(2),CK(2),Y(2),ALOS(2)
      DATA XMODE   /7HLAMINAR,9HTURBULENT/
      DATA POW1/.5,.33333333/
      DATA CK    /.575,.03404/
      DATA Y    /.5,.8/
C
C
      COMMON /PRMTAB/TAB13,TAB15,TAB17,TAB56,TAB57,TAB58,TAB59,TAB60
      COMMON /367/ AM367(150)
C
C
      DIMENSION TAB13(31),TAB15(7),
     1TAB59(19,9),TAB60(19,9),TAB17(31,7),TAB56(9),TAB57(19),TAB58(19,9)
      DIMENSION     XL    (  5),     CO    (  5),    CI    (  5),                15
     *              RO    (  5),     EMO   (  5),    EMI   (  5),                30
     *              TI    (  5),     AB    (  5),    ETL   (  5),                45
     *              ETT   (  5),     ZTL   (  5),    ZTT   (  5),                60
     *CPO   (   3),    RHOS (  4),
     *              ABLAT(  5),     TITLA(  9),    TITLB(  9),                490
     *              TITLC(  9),     KASE (  5),    MVSV (  6),                510
     *              RE    (  3)
C
C
      EQUIVALENCE  (AM367(  1), PAGE    )
     *,            (AM367(  3),TITLA(1)),(AM367( 12), TITLB(1) )                20
     *,            (AM367( 21),TITLC(1)),(AM367( 30), IRETML   )
     *,            (AM367( 35),NOK     ),(AM367( 41), I20      )
     *,            (AM367( 42),I22     ),(AM367( 43), NMAT     )                43
     *,            (AM367( 44),RAD     ),(AM367( 45), RETMLC   )                45
     *,            (AM367( 46),DTAU    ),(AM367( 47), TR       )                47
     *,            (AM367( 48),RE(1)   ),(AM367( 51), DELS     )                51
     *,            (AM367( 52),XL(1)   ),(AM367( 57),CO(1)     )
     *,            (AM367( 62),CI(1)   ),(AM367( 67),RO(1)     )
     *,            (AM367( 72),EMO(1)  ),(AM367( 77),EMI(1)    )
     *,            (AM367( 82),TI(1)   ),(AM367( 87),AB(1)     )
     *,            (AM367( 92),ETL(1)  ),(AM367( 97),ETT(1)    )
     *,            (AM367(102),ZTL(1)  ),(AM367(107),ZTT(1)    )
     *,            (AM367(112),CPO(1)  ),(AM367(115),RHOS(1)   )
     *,            (AM367(119),ABLAT(1)),(AM367(124),NPRNT(1)  )
     *,            (AM367(125),TIME    )
      EQUIVALENCE (AM367(126),TRAD),(AM367(131),AMODEL),
     1            (AM367(132),TRCUT)
      DIMENSION TRAD(5)
      EQUIVALENCE                    (RHOS(4),TSI)
     3,(RHOS(2),EO),(RHOS(3),EI),(RE(2),X),(RE(3),THETA)
C
      EQUIVALENCE (AM367(31),K1)
C
      DATA PI,SIG/3.1415927,.475833E-12/
C
C****          SIG IS THE BLACK BODY RADIATION CONSTONT IN UNITS OF
C****          BTU/(SEC*FT**2(RANKINE DEGREE)**4)
C
925   FORMAT(24H0ATMOSPHERE PARAMETERS, /34X,11HFREE STREAM10X,5HLOCAL15
     1X5HUNITS20H   LOCAL/FREE STREAM/11H FLOW SPEED19X2E15.6,7H FT/SEC
```

```
     213X,E15.6/12H TEMPERATURE18X2E15.6,10H DEGREES R10XE15.6/9H PRESSU
     3RE21X2E15.6,9H LB/FT**211XE15.6/12H MACH NUMBER18X2E15.6,20XE15.6
     4/16H SPECIFIC WEIGHT14X2E15.6,9H LB/FT**311XE15.6/10H VISCOSITY20X
     52E15.6,15H LB*(SEC/FT)**35XE15.6/9H ENTHALPY21X2E15.6,7H BTU/LB
     6E28.6/16H COMPRESSIBILITY14X2E15.6,20X,E15.6/15H PRANDTL NUMBER15X
     72E15.6,20XE15.6/25H REYNOLDS NUMBER PER FOOT5X2E15.6,6H 1./FT14X
     8 E15.6)
950   FORMAT(2H0 A10/10H ABLAT(1) ,E25.6,7H BTU/LB,8X,4H ETA,E31.6/5H ZE
     1TA,E30.6,15X,2H YE33.6/3H CKE32.6,15X,16H RECOVERY FACTOR,E19.6/
     215H RADIATION RATE, E20.6,15H BTU/SEC/FT**2 ,16H ABLATION ENERGY,
     3E19.6,7H BTU/LB/16H DQDADT AT XHIGH,E19.6,15H BTU/SEC-FT**2 ,
     419H TOTAL HEATING RATE,E16.6,8H BTU/SEC   /17H RADIATION CUTOFF,
     5E18.6,3H FT,12X,17H TRANSITION POINT,E18.6,3H FT/5H XLOW,  E30.6,
     63H FT,12X,6H XHIGH,E29.6,3H FT/20H ADIABATIC WALL TEMP,E15.6,10H D
     7EGREES R,5X,13H ECKERTS TEMP,E22.6,10H DEGREES R/19H ECKERTS PRAND
     8TL NO,E16.6,15X,20H ECKERTS REYNOLDS NO,E15.6,5H 1/FT/16H COMPRESS
     9IBILITYE19.6,15X18H ECKERTS VISCOSITYE17.6,15H LB*(SEC/FT)**3/
     *20H ECKERTS SPECIFIC WTE15.6,9H LB/FT**36X19H ADIABATIC ENTHALPY
     *E16.6,7H BTU/LB/17H ECKERTS ENTHALPYE18.6,7H BTU/LB8X,
     *12H WEIGHT LOSS E23.6,3H LB/14H ABLATION TEMP E21.6,10H DEGREES R
     * 6X17HHEATING RATE AT X E17.6,14H BTU/SEC-FT**2)
937   FORMAT( 6H0TIME E15.6, 9H  SECONDS11H   ALTITUDEE15.6,6H  FEET)
  922 FORMAT (1H130X24HTURBULENT BOUNDARY LAYER/1H030X5A10)
C
      IF(K1.EQ.6) GO TO 580
C
      ALOS(1) = 0.
      ALOS(2) = 0.
      EMDOT =0.
      TEMDOT=0.
      TSHSTR=0.
      TRBQ =0.
      GO TO (580,585),IENTRY
580   CONTINUE
      XIN=X
      X=X/12.0
      DHETA = PI * THETA / 180.
      XLKO=X/COS(DHETA)
      CONTINUE
C
C****          SET THE ANGLE OF ATTACK, ALPHAE, EQUAL TO THE CONE
C****          HALF ANGLE, THETA.
C
      ALPHAE = THETA
C
C****          GET THE LOCAL VALUES OF TEMPERATURE, TL, PRESSURE, PL,
C****          FLOW SPEED, VL, THE FREE STREAM VALUES OF PRESSURE, PO,
C****          TEMPERATURE, TO, THE RATIO OF LOCAL TO FREE STREAM VALUES
C****          FOR PRESSURE, PPO, TEMPERATURE, TTO, FLOW SPEED, VVO,
C****          AND THE MACH NUMBER, AMO.
C
C****          UNITS.
C
C****          TL              DEGREES RANKINE.
C****          PL              LB/FT**2.
C****          VL              FEET/SEC.
C****          PO              LB/FT**2.
C****          TO              DEGREES RANKINE.
C****          PPO             UNITLESS RATIO.
C****          TTO             UNITLESS RATIO.
C****          VVO             UNITLESS RATIO.
C****          AMO             UNITLESS.
```

```
C
      CALL ATTACK(ALT,ALPHAE,VO,PO,TO,VL,PL,TL,PPO,TTO,VVO,AMO)
C
C****          CHECK THE FREE STREAM TEMPERATURE.  MUST BE BETWEEN
C****          0. AND 27000. DEGREES RANKINE.  TEMPERATURES OUT OF
C****          RANGE ARE SET TO THE APPROPIATE TEMPERATURE LIMIT.
C
      CALL TCHEK(TO,1)
C
C****          CONVERT THE FREE STREAM PRESSURE FROM LB/FT**2 TO
C****          ATMOSPHERES.
C
      PO1 = PO/2116.8
C
C****          CHECK THE FREE STREAM PRESSURE.  THE PRESSURE MUST BE
C****          BETWEEN 0.0001 AND 100 AATMOSPHERES.  PRESSURES OUT OF
C****          RANGE ARE SET TO THE APPROPIATE PRESSURE LIMIT.
C
      CALL PCHEK(PO1,2)
C
C****          SETUP THE FREE STREAM PRESSURE FOR HANSEN TABLES.
C
      PO1=ALOG10(PO1)
C
C****          GET THE NO FLOW FREE STREAM ENTHALPY.  UNITS ARE
C****          BTU/LB.
C
      CALL HANSEN (3,TO,PO1,CHO)
C
C****          ADD THE FREE STREAM FLOW ENTHALPY.  UNITS ARE BTU/LB.
C
      CHT=CHO+((VO**2)/(5.012E4))
C
C**            CALCULATE THE FREE STREAM DENSITY IN    (LB-MASS)/FT**3.
C
      ROO=(PO)/(53.35*TO)
C
C****          CALCULATE THE FREE STREAM VISCOSITY. UNITS ARE
C****          (LB-FORCE)* SEC/FT**3.
C
      XMUOO=(1.18E-5)*(716./(TO+216.))*(TO/500.0)**(3./2.)
C
C****          CALCULATE THE FREE STREAM  REYNOLDS NUMBER PER FOOT.
C
      REFS=(VO*ROO)/(XMUOO)
C
C****          CHECK THE LOCAL TEMPERATURE.  MUST BE BETWEEN 0. AND
C****          27000. DEGREES RANKINE.  TEMPERATURES OUT OF RANGE ARE
C****          SET TO THE APPROPIATE TEMPERATURE LIMIT.
C
      CALL TCHEK(TL,2)
C
C****          CONVERT THE LOCAL PRESSURE FROM (LB-FORCE)/FT**2
C****          TO ATMOSPHERES.
C
      PL1 = PL/2116.8
C
C****          CHECK THE LOCAL PRESSURE.  THE PRESSURE MUST BE BETWEEN
C****          0.0001 AND 100 ATMOSPHERES.  PRESSURES OUT OF RANGE
C****          ARE SET TO THE APPROPIATE PRESSURE LIMIT.
C
      CALL PCHEK(PL1,1)
```

```
C
C****          SETUP THE LOCAL PRESSURE FOR HANSEN TABLES.
C
      PL1=ALOG10(PL1)
C
C****          GET THE COMPRESSIBILITY FOR LOCAL CONDITIONS. (UNITLESS)
C
      CALL HANSEN (5,TL,PL1,ZL)
C
C****          CALCULATE THE LOCAL DENSITY.  UNITS ARE (LB-MASS)/FT**3.
C
      RL=PL/(53.35*ZL*TL)
C
C****          CALCULATE THE LOCAL VISCOSITY.  UNITS ARE
C****          (LB-FORCE)*SEC/FT**3.
C
      CL=(1.18E-5)*(716./(TL+216.))*((TL/500.)**(3./2.))
C
C****          INTERPOLATE TO GET THE LOCAL PRANDTL NUMBER AS A FUNCTION
C****          OF TEMPERATURE AND PRESSURE.
      CALL DISCOT (TL,PL1,TAB13,TAB17,TAB15,111,217,7,PRL,31)
C
C****          CALCULATE THE DIFFERENCE BETWEEN THE FREE STREAM AND
C****          LOCAL FLOW ENTHALPYS.
C
      CHL=CHT-((VL**2)/(5.012E4))
C
C****          CALCULATE THE GAS ENTHALPY AT THE WALL.  UNITS BTU/LB.
C
      CALL HANSEN(3,TSI,PL1,CHS)
      RELF=RL*VL/CL
      ZO = 1.
      AML = VL/(49.1*SQRT(TL))
      CALL DISCOT(TO,PO1,TAB13,TAB17,TAB15,1,217,7,PRO,31)
      AAMO = AML/AMO
      RROO = RL/ROO
      CLCO = CL/XMUOO
      HLHO = CHS/CHO
      PPL  = PRL/PRO
      REYROF = RELF/REFS
C****          CALCULATE THE MATERIAL RADIATION CONSTANT.
C
      ES = EO + EI * TRCUT
C
C****          CALCULATE THE RATE ENERGY IS RADIATED AWAY PER UNIT AREA.
C****          UNITS ARE BTU/SEC/FT**2
C
      QRAD = SIG * ES * TRCUT**4
      IF(ICODE.NE.0) GO TO 600
      IF(K1.EQ.6) GO TO 600
      WRITE (6,922) (TITLA(I),I=1,5)
      WRITE(6,937) TIME,ALT
      WRITE(6,925)VO,VL,VVO,TO,TL,TTO,PO,PL,PPO,AMO,AML,AAMO,ROO,RL,RROO
     1,XMUOO,CL,CLCO,CHO,CHS,HLHO,ZO,ZL,ZL,PRO,PRL,PPL,REFS,RELF,REYROF
600   CONTINUE
      DO 1000 I=1,2
      IF(K1.EQ.6.AND.I.EQ.1) GO TO 1000
      R=PRL**POW1(I)
C
C****          CALCULATE THE TOTAL LOCAL ENTHALPY FOR AN ADIABATIC WALL.
C****          UNITS ARE BTU/LB.
C
```

```
      CHAW=CHL+((R*VL*VL)/(5.012E4))
C
C
      CALL HANSEN (4,CHAW,PL1,TAW)
      CH=CHL+.5*(CHS-CHL)+.22*(CHAW-CHL)
      CALL HANSEN (4,CH,PL1,THS)
      CMU=(1.18E-5)*(716./(THS+216.))*((THS/500.)**(3./2.))
      CALL HANSEN (5,THS,PL1,ZTHS)
      RS=PL/(53.35*ZTHS*THS)
      RES=RS*VL/CMU
      IF(K1.EQ.6) GO TO 650
      CALL DISCOT (THS,PL1,TAB13,TAB17,TAB15,111,217,7,PRS,31)
      IF(I.GT.1) GO TO 999
      IF(IRETML.GT.1) GO TO 998
      STRANS=RE(1)/RELF
      GO TO 999
998   STRANS=3.*VL*CL/RS/CMU/TL*(RETMLC/(.664*49.1))**2 *CL
999    CONTINUE
      IF(I.GT.1) GO TO 980
      XLOW=0.
      IF(STRANS.GT.XLKO) GO TO 960
      XHIGH=STRANS
      GO TO 981
960   XHIGH = XLKO
      GO TO 981
980   XLOW =STRANS
      XHIGH=XLKO
981   CONTINUE
      PSI=CK(I)*RES**Y(I)*CMU*PRS**(-.6666667)*(CHAW-CHS)
      IF (PSI.LE. 1.0E-10)  PSI = 1.0E-10
      XRAD=(QRAD/PSI)**(1./(Y(I)-1.))
      IF (PSI.LE. 1.0E-10)  XRAD = XLOW
      IF(XRAD.GT.XLOW) GO TO 970
      XLOW=XHIGH
      GO TO 971
970   CONTINUE
      IF(XRAD.GT.XHIGH) GO TO 972
      XHIGH=XRAD
971   CONTINUE
972   QDADT=PSI*XHIGH**(Y(I)-1.)
      QT=2.*PI*PSI*((SIN(DHETA)/(Y(I)+1.))*(XHIGH**(Y(I)+1.)-XLOW**(
     1 Y(I)+1.))+(RAD*(1.-SIN(DHETA))/Y(I)/COS(DHETA))*(XHIGH**Y(I)-
     2 XLOW**Y(I))/12.)
      QSTAR=       ABLAT(1)+ABLAT(I+1)*(CHT-CHS)+ABLAT(I+3)*CHT
      ALOS(I) = (QT - AMODEL * AREA * QRAD) / QSTAR
      Q = PSI*XLKO**(Y(I)-1.0)
      IF(ICODE.NE.0) GO TO 601
      WRITE(6,950) XMODE(I),ABLAT(1),ABLAT(I+1),ABLAT(I+3), Y(I), CK(I),
     1 R,QRAD, QSTAR, QDADT,QT,XRAD,STRANS,XLOW,XHIGH,TAW,THS,PRS,RES,
     2 ZTHS,CMU,RS,CHAW,CH,ALOS (I),TSI
     3 ,Q
601   CONTINUE
      IF(I.EQ.1) GO TO 800
2221  EMDOT=Q/QSTAR
      TEMDOT = EMDOT - QRAD / QSTAR * AMODEL
650   CONTINUE
      TSHSTR = CK(I)/(RES*XLKO)**.2 *RS*VL*VL
      IF(K1.EQ.6) RETURN
      TRBQ=Q
      IF (Q.LE.QRAD)  TEMDOT = 0.0
800   IF(STRANS.GT.XLKO) GO TO 801
1000   CONTINUE
```

```
801   CONTINUE
      AMLOS=ALOS(1)+ALOS(2)
      X = XIN
      RETURN
         END
```

D. BLOCK DATA

```
      BLOCK DATA
      COMMON /PRMTAB/TAB13,TAB15,TAB17,TAB56,TAB57,TAB58,TAB59,TAB60
      DIMENSION TAB13(31),TAB15(7),TAB17(31,7),TAB56(9),TAB57(19),
     *TAB58(19,9),TAB59(19,9),TAB60(19,9)
      DATA TAB13/0.0,900.0,1800.0,2700.0,3600.0,4500.0,5400.0,6300.0,
     17200.0,8100.0,9000.0,9900.0,10800.0,11700.0,12600.0,13500.0,
     214400.0,15300.0,16200.0,17100.0,18000.0,18900.0,19800.0,20700.0,
     321600.0,22500.0,23400.0,24300.0,25200.0,26100.0,27000.0/
      DATA TAB15/-4.0,-3.0,-2.0,-1.0,0.0,1.0,2.0/
      DATA TAB17/2*.738,.756,.767,.614,.771,.714,.606,.587,.764,.993,
     1.871,.455,.392,.361,.342,.322,.279,.200,.114,.361,.0314,.0213,
     2.0167,.0143,.0129,.0121,.0110,.0108,.0109,.011,2*.738,.756,.767,
     3.668,.654,.745,.658,.580,.611,.799,.989,.891,.464,.404,.371,.351,
     4.335,.316,.279,.216,.145,.0877,.0524,.0346,.0238,.019,.0162,.0149,
     5.013,.012,2*.738,.756,.767,.724,.611,.74,.737,.619,.578,.624,.785,
     6.969,.955,.83,.424,.387,.363,.348,.336,.319,.295,.254,.201,.146,
     7.101,.0688,.047,.0345,.0245,.0129,2*.738,.756,.767,.766,.645,.636,
     8.744,.759,.610,.581,.617,.736,.906,.986,.969,.648,.411,.382,.364,
     9.348,.339,.327,.312,.292,.263,.227,.185,.144,.0986,.0819,2*.738,
     0.756,.767,.773,.696,.627,.660,.762,.752,.611,.583,.602,.673,.796,
     1.927,.983,.943,.807,.497,.429,.404,.382,.369,.355,.343,.333,.319,
     2.302,.277,.253,2*.738,.756,.767,.773,.751,.68,.631,.662,.743,.767,
     3.620,.592,.592,.62,.688,.788,.891,.961,.966,.872,.532,.463,.434,
     4.412,.396,.383,.369,.360,.349,.341,2*.738,.756,.767,.773,.762,.74,
     5.678,.64,.654,.702,.748,.763,.61,.593,.595,.62,.666,.73,.806,.886,
     6.937,.955,.947,.908,.728,.525,.438,.421,.401,.394/
      DATA TAB56,TAB57/2.0,5.0,7.5,10.0,12.5,15.0,20.0,25.,30.,2.,3.,4.,
     15.,6.,7.,8.,9.,10.,11.,12.,13.,14.,15.,17.,19.,21.,23.,25./
      DATA TAB58/.9964,.9969,.9971,.9974,.9976,.9978,.9979,.9980,.9981,
     1.9982,.9983,.9984,.9985,.9986,.9987,.9988,.9990,.9991,.9992,.9833,
     2.9866,.9884,.9898,.9908,.9915,.9920,.9924,.9929,.9932,.9935,.9938,
     3.9940,.9941,.9942,.9945,.9947,.9949,.9951,.9688,.9753,.9789,.9814,
     4.9831,.9846,.9854,.9861,.9868,.9872,.9876,.9879,.9881,.9883,.9886,
     5.9889,.9890,.9891,.9892,.9515,.9620,.9680,.9715,.9741,.9758,.9770,
     6.9780,.9786,.9792,.9797,.9800,.9803,.9806,.9810,.9811,.9815,.9819,
     7.9821,.9325,.9470,.9554,.9600,.9629,.9650,.9664,.9674,.9682,.9689,
     8.9693,.9698,.9700,.9702,.9707,.9709,.9711,.9715,.9719,.9110,.9305,
     9.9408,.9463,.9498,.9520,.9538,.9549,.9557,.9562,.9569,.9572,.9576,
     0.9579,.9581,.9585,.9587,.9590,.9591,.8610,.8925,.9060,.9135,.9175,
     1.9203,.9223,.9233,.9243,.9249,.9253,.9258,.9261,.9266,.9269,.9271,
     2.9274,.9278,.9281,.8060,.8535,.8660,.8737,.8787,.8817,.8840,.8859,
     3.8870,.8880,.8885,.8890,.8893,.8895,.8901,.8907,.8911,.8913,.8918,
     4.7350,.7905,.8105,.8207,.8260,.8294,.8319,.8335,.8347,.8355,.8362,
     5.8368,.8371,.8372,.8377,.8380,.8383,.8388,.8390/
      DATA TAB59/1.005,1.0101,1.0165,1.024,1.0327,1.042,1.052,1.0635,
     11.075,1.086,1.098,1.110,1.123,1.137,1.164,1.194,1.227,1.261,1.299,
     21.0265,1.048,1.074,1.102,1.133,1.169,1.203,1.242,1.282,1.328,
     31.372,1.421,1.475,1.530,1.650,1.787,1.935,2.100,2.29,1.0448,1.087,
     41.133,1.182,1.240,1.30,1.369,1.441,1.523,1.614,1.710,1.815,1.925,
     52.04,2.30,2.59,2.91,3.28,3.70,1.077,1.134,1.203,1.281,1.370,1.475,
     61.59,1.714,1.850,2.00,2.16,2.350,2.55,2.750,3.20,3.7,4.27,4.91,
     75.62,1.108,1.188,1.285,1.395,1.525,1.677,1.845,2.03,2.25,2.49,
     82.74,3.02,3.33,3.65,4.35,5.1,5.95,6.9,8.0,1.137,1.241,1.37,1.515,
     91.7,1.91,2.15,2.45,2.73,3.07,3.45,3.85,4.28,4.70,5.73,6.90,8.16,
     09.65,11.40,1.216,1.37,1.58,1.85,2.15,2.51,2.95,3.41,3.93,4.5,5.15,
     15.86,6.60,7.40,9.1,11.1,13.3,15.9,18.8,1.285,1.51,1.83,2.20,2.68,
     23.20,3.83,4.53,5.35,6.25,7.23,8.30,9.40,10.6,13.35,16.35,19.9,
```

```
323.9,28.5,1.365,1.690,2.11,2.65,3.30,4.05,4.93,5.91,7.05,8.30,
49.70,11.15,12.90,14.5,18.0,22.3,27.0,32.4,38.7/
 DATA TAB60/1.01,1.035,1.06,1.09,1.125,1.16,1.195,1.235,1.275,1.32,
11.36,1.41,1.46,1.51,1.64,1.78,1.94,2.10,2.33,1.095,1.18,1.285,
21.40,1.545,1.705,1.890,2.09,2.305,2.545,2.81,3.1,3.4,3.73,4.47,
35.27,6.2,7.16,8.25,1.185,1.345,1.546,1.795,2.09,2.425,2.82,3.25,
43.730,4.25,4.85,5.49,6.19,6.90,8.5,10.3,12.3,14.5,16.90,1.295,
51.56,1.89,2.31,2.81,3.40,4.11,4.89,5.70,6.65,7.65,8.75,9.9,11.2,
614.0,17.35,20.65,24.5,28.75,1.42,1.8,2.31,2.83,3.73,4.63,5.65,
76.8,8.1,9.55,11.15,12.9,14.75,16.65,21.1,26.25,31.8,38.0,44.9,
81.565,2.08,2.79,3.695,4.79,6.05,7.50,9.25,11.1,13.2,15.45,17.90,
920.5,23.25,29.5,36.7,44.60,53.6,63.0,1.91,2.8,4.03,5.57,7.45,9.7,
012.25,15.1,18.45,22.0,26.0,30.2,35.0,40.0,50.9,63.5,77.5,92.5,
1110.0,2.325,3.64,5.5,7.85,10.7,14.15,18.0,22.5,27.4,33.0,39.0,
245.7,52.9,60.0,77.0,96.0,115.0,135.0,165.0,2.8,4.635,7.2,10.5,
314.5,19.4,25.0,31.0,38.0,45.7,54.3,63.0,73.5,83.5,107.5,135.0,
4165.0,190.0,230.0/
 END
```

20. SUBROUTINE ATTACK

a. General

Subroutine ATTACK calculates the free-stream temperature and pressure, the local temperature, pressure, flow speed, free-stream Mach number, local-to-free-stream temperature, pressure, and flow speed ratios. Altitude, speed, and effective angle-of-attack are input parameters to ATTACK. The effective angle-of-attack is assumed the same as the half-cone angle. Free-stream temperature and pressure are calculated from the 1959 ARDC Model Atmosphere (Section IV, 4).

The free-stream Mach number, M, is calculated from

$$M = V_o / \left(49.1\sqrt{T_o}\right) \tag{67}$$

Local parameters (functions of M and α_e) are therefore,

$$V = \left(V/V_o\right)_{M,\alpha_e} V_o \tag{68}$$

$$T = \left(T/T_o\right)_{M,\alpha_e} T_o \tag{69}$$

$$P = \left(P/P_o\right)_{M,\alpha_e} P_o \tag{70}$$

If α_e or M is ≤ 2, the local-to-free-stream ratios are set to 1. Figures 7 through 9 show the curves used to determine local-to-free-stream ratios for $2 \leq \alpha_e \leq 30$ and $2 \leq M \leq 25$. These values are tabulated as block data in paragraph (d) of this subroutine. "Linear" extrapolation is used for values of $\alpha_e > 30$ and $M > 25$. If $\alpha_e < 2$ or $M < 2$ a message is printed.

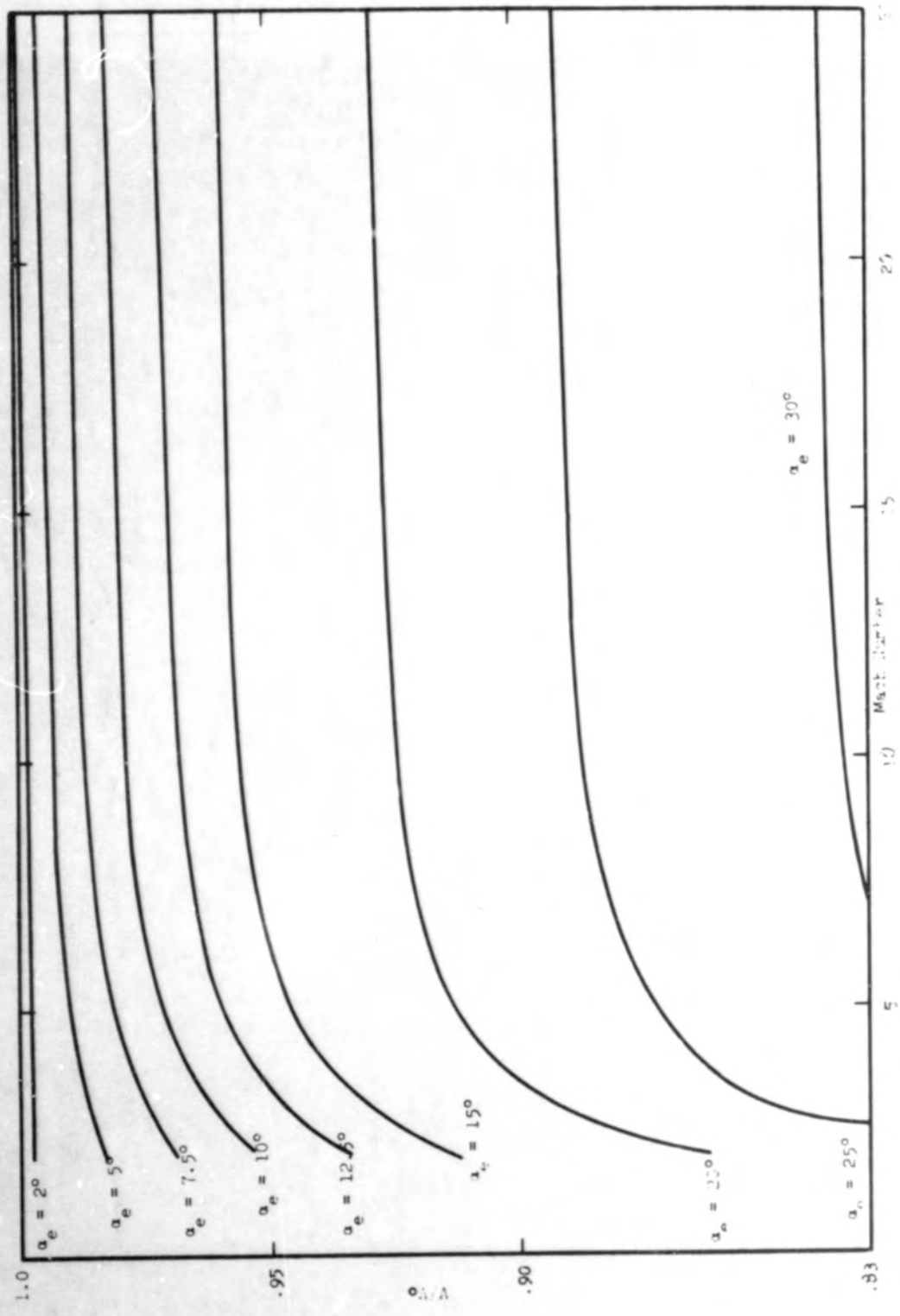


Figure 7. Ratio of Local Flow Speed to Free-Stream Flow Speed

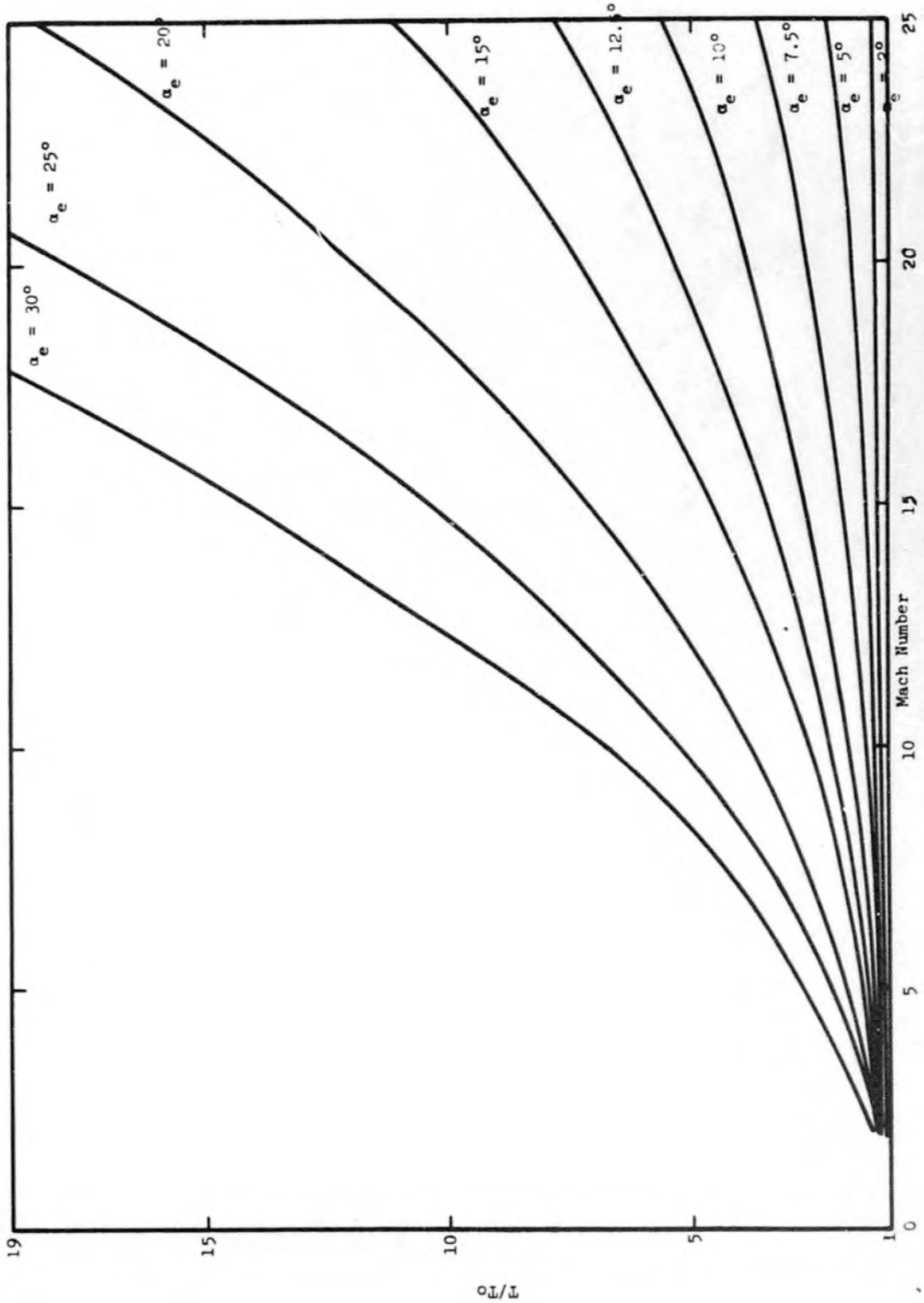


Figure 8. Ratio of Local Temperature to Free-Stream Temperature

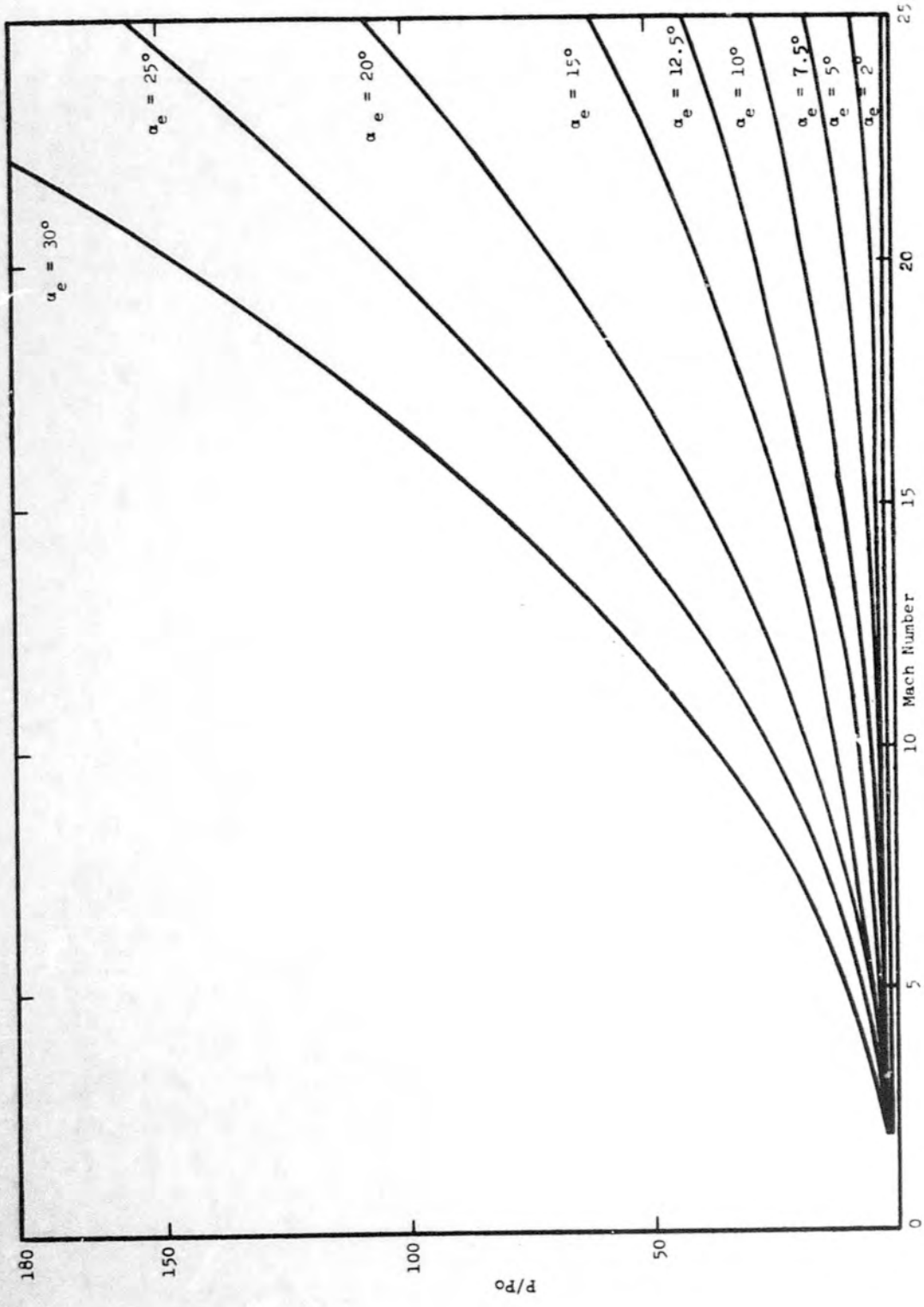
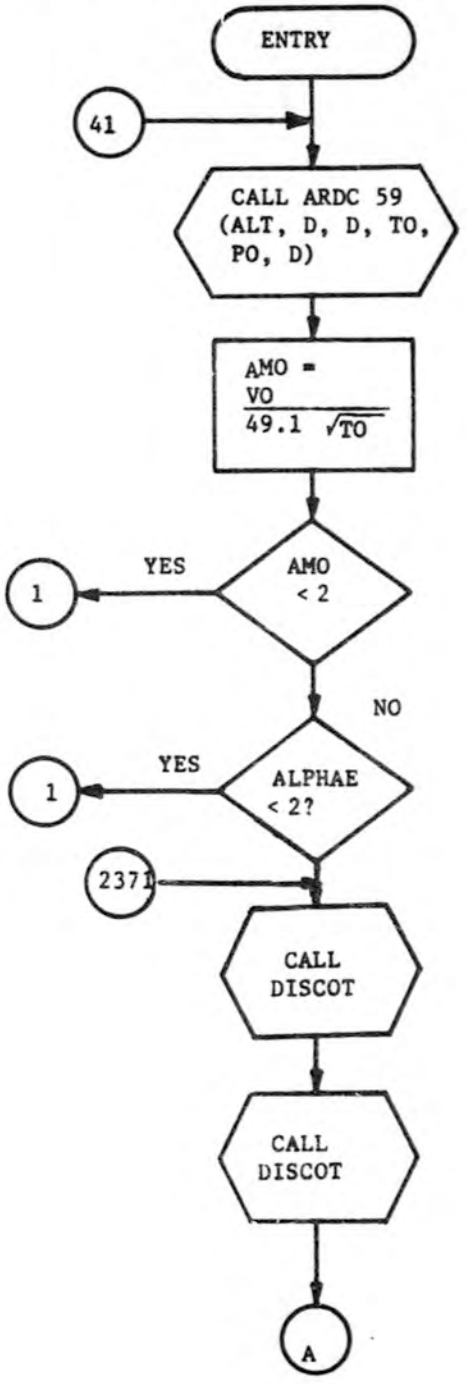


Figure 9. Ratio of Local Pressure to Free-Stream Pressure

b. Flow Chart



GET THE FREE-STREAM TEMPERATURE AND PRESSURE

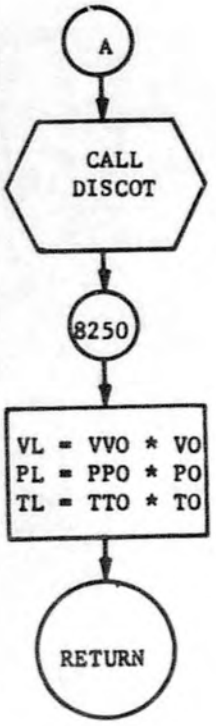
CALCULATE THE FREE-STREAM MACH NUMBER

IS THE MACH NUMBER LESS THAN 2?

IS THE EFFECTIVE ANGLE-OF-ATTACK LESS THAN 2?

GET THE RATIO OF THE LOCAL TO FREE STREAM VELOCITY, VVO. "LINEAR" INTERPOLATION AS A FUNCTION OF α_e AND M

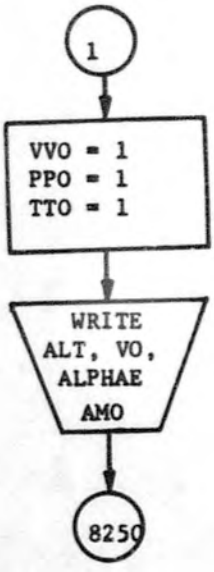
GET THE RATIO OF THE LOCAL TO FREE-STREAM TEMPERATURE, TTO. "LINEAR" INTERPOLATION AS A FUNCTION OF α_e AND M



GET THE RATIO OF THE LOCAL-TO-FREE-STREAM PRESSURE, PPO. "LINEAR" INTERPOLATION AS A FUNCTION OF α_e AND M

CALCULATE THE LOCAL VELOCITY, PRESSURE, AND TEMPERATURE, VL, PL AND TL

BACK TO THE CALLING PROGRAM



SET RATIOS WHEN THE MACH NUMBER AND/OR THE EFFECTIVE ANGLE-OF-ATTACK IS OFF THE LOWER END

PRINT OUT VALUES, SINCE THE RESULTS MAY BE IN DOUBT

C. LISTING

```
      SUBROUTINE ATTACK(ALT,ALPHAE,VO,PO,TO,VL,PL,TL,PPO,TTO,VVO,AMO)
      COMMON /PRMTAB/TAB13,TAB15,TAB17,TAB56,TAB57,TAB58,TAB59,TAB60
      DIMENSION TAB13(31),TAB15(7),
     1TAB59(19,9),TAB60(19,9),TAB17(31,7),TAB56(9),TAB57(19),TAB58(19,9)
41    CALL ARDC59(ALT,D,D,TO,PO,D)
      AMO=VO/(49.1*SQRT(TO))
      IF(AMO.LT. 2.) GO TO 1
      IF(ALPHAE.LT.2.) GO TO 1
2371  CALL DISCOT (AMO,ALPHAE,TAB57,TAB58,TAB56,111,171,9,VVO,19)
      CALL DISCOT (AMO,ALPHAE,TAB57,TAB59,TAB56,111,171,9,TTO,19)
      CALL DISCOT (AMO,ALPHAE,TAB57,TAB60,TAB56,111,171,9,PPO,19)
8250  VL=VVO*VO
      PL=PPO*PO
      TL=TTO*TO
      RETURN
1     VVO=1.0
      PPO=1.0
      TTO=1.0
      WRITE(6,50) ALT,VO,ALPHAE,AMO
50    FORMAT(56H0 ANGLE OF ATTACK AND MACH NUMBER TABLE RANGES EXCEEDED.
     110H ALTITUDE 20XE15.6,5H FEET/18H FREE STREAM SPEED12XE15.6,9H FEE
     2T/SEC/16H ANGLE OF ATTACK14XE15.6,8H DEGREES/12H MACH NUMBER18XE15
     3.6)
      GO TO 8250
      END
```

D. **BLOCK DATA**

```
      BLOCK DATA
      COMMON /PRMTAB/TAB13,TAB15,TAB17,TAB56,TAB57,TAB58,TAB59,TAB60
      DIMENSION TAB13(31),TAB15(7),TAB17(31,7),TAB56(9),TAB57(19),
     *TAB58(19,9),TAB59(19,9),TAB60(19,9)
      DATA TAB13/0.0,900.0,1800.0,2700.0,3600.0,4500.0,5400.0,6300.0,
     17200.0,8100.0,9000.0,9900.0,10800.0,11700.0,12600.0,13500.0,
     214400.0,15300.0,16200.0,17100.0,18000.0,18900.0,19800.0,20700.0,
     321600.0,22500.0,23400.0,24300.0,25200.0,26100.0,27000.0/
      DATA TAB15/-4.0,-3.0,-2.0,-1.0,0.0,1.0,2.0/
      DATA TAB17/2*.738,.756,.767,.614,.771,.714,.606,.587,.764,.993,
     1.871,.455,.392,.361,.342,.322,.279,.200,.114,.361,.0314,.0213,
     2.0167,.0143,.0129,.0121,.0110,.0108,.0109,.011,2*.738,.756,.767,
     3.668,.654,.745,.658,.580,.611,.799,.989,.891,.464,.404,.371,.351,
     4.335,.316,.279,.216,.145,.0877,.0524,.0346,.0238,.019,.0162,.0149,
     5.013,.012,2*.738,.756,.767,.724,.611,.74,.737,.619,.578,.624,.795,
     6.969,.955,.83,.424,.387,.363,.348,.336,.319,.295,.254,.201,.146,
     7.101,.0688,.047,.0345,.0245,.0129,2*.738,.756,.767,.766,.645,.636,
     8.744,.759,.610,.581,.617,.736,.906,.986,.969,.648,.411,.382,.364,
     9.348,.339,.327,.312,.292,.263,.227,.185,.144,.0986,.0819,2*.738,
     0.756,.767,.773,.696,.627,.660,.762,.752,.611,.583,.602,.673,.796,
     1.927,.983,.943,.807,.497,.429,.404,.382,.369,.355,.343,.333,.319,
     2.302,.277,.253,2*.738,.756,.767,.773,.751,.68,.631,.662,.743,.767,
     3.620,.592,.592,.62,.688,.789,.891,.961,.966,.872,.532,.463,.434,
     4.412,.396,.383,.369,.360,.349,.341,2*.738,.756,.767,.773,.762,.74,
     5.678,.64,.654,.702,.748,.763,.61,.593,.595,.62,.666,.73,.806,.886,
     6.937,.955,.947,.908,.728,.525,.438,.421,.401,.394/
      DATA TAB56,TAB57/2.0,5.0,7.5,10.0,12.5,15.0,20.0,25.,30.,2.,3.,4.,
     15.,6.,7.,8.,9.,10.,11.,12.,13.,14.,15.,17.,19.,21.,23.,25./
      DATA TAB58/.9964,.9969,.9971,.9974,.9976,.9978,.9979,.9980,.9981,
     1.9982,.9983,.9984,.9985,.9986,.9987,.9988,.9990,.9991,.9992,.9835,
     2.9866,.9884,.9898,.9908,.9915,.9920,.9924,.9929,.9932,.9935,.9938,
     3.9940,.9941,.9942,.9945,.9947,.9949,.9951,.9688,.9753,.9789,.9814,
     4.9831,.9846,.9854,.9861,.9868,.9872,.9876,.9879,.9881,.9883,.9886,
     5.9889,.9890,.9891,.9892,.9515,.9620,.9680,.9715,.9741,.9758,.9770,
     6.9780,.9786,.9792,.9797,.9800,.9803,.9806,.9810,.9811,.9815,.9819,
     7.9821,.9325,.9470,.9554,.9600,.9629,.9650,.9664,.9674,.9682,.9689,
     8.9693,.9698,.9700,.9702,.9707,.9709,.9711,.9715,.9719,.9110,.9305,
     9.9408,.9463,.9498,.9520,.9538,.9549,.9557,.9562,.9569,.9572,.9576,
     0.9579,.9581,.9585,.9587,.9590,.9591,.8610,.8925,.9060,.9135,.9175,
     1.9203,.9223,.9233,.9243,.9249,.9253,.9258,.9261,.9266,.9269,.9271,
     2.9274,.9278,.9281,.8060,.8535,.8660,.8737,.8787,.8817,.8840,.8859,
     3.8870,.8880,.8885,.8890,.8893,.8895,.8901,.8907,.8911,.8913,.8916,
     4.7350,.7905,.8105,.8207,.8260,.8294,.8319,.8335,.8347,.8355,.8362,
     5.8368,.8371,.8372,.8377,.8380,.8383,.8388,.8390/
      DATA TAB59/1.005,1.0101,1.0165,1.024,1.0327,1.042,1.052,1.0635,
     11.075,1.086,1.098,1.110,1.123,1.137,1.164,1.194,1.227,1.261,1.299,
     21.0265,1.048,1.074,1.102,1.133,1.169,1.203,1.242,1.282,1.328,
     31.372,1.421,1.475,1.530,1.650,1.787,1.935,2.100,2.29,1.0448,1.087,
     41.133,1.182,1.240,1.30,1.369,1.441,1.523,1.614,1.710,1.815,1.925,
     52.04,2.30,2.59,2.91,3.28,3.70,1.077,1.134,1.203,1.281,1.370,1.475,
     61.59,1.714,1.850,2.00,2.16,2.350,2.55,2.750,3.20,3.7,4.27,4.91,
     75.62,1.108,1.188,1.285,1.395,1.525,1.677,1.845,2.03,2.25,2.49,
     82.74,3.02,3.33,3.65,4.35,5.1,5.95,6.9,8.0,1.137,1.241,1.37,1.515,
     91.7,1.91,2.15,2.45,2.73,3.07,3.45,3.85,4.28,4.70,5.73,6.90,8.16,
     09.65,11.40,1.216,1.37,1.58,1.85,2.15,2.51,2.95,3.41,3.93,4.5,5.15,
```

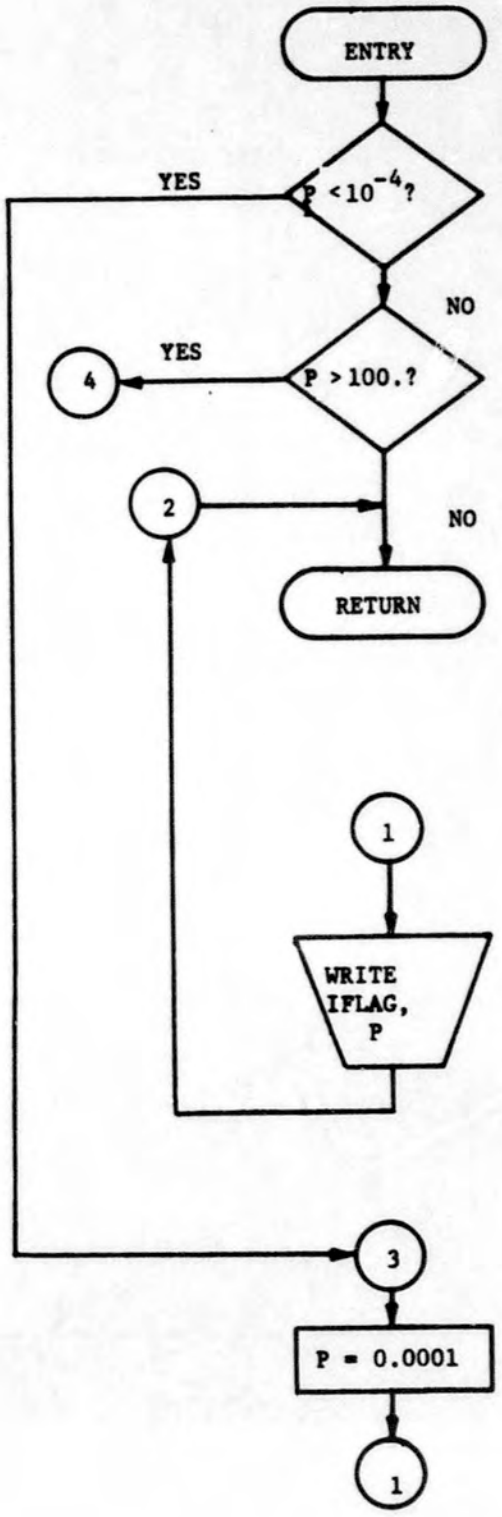
```
15.86,6.60,7.40,9.1,11.1,13.3,15.9,18.3,1.285,1.51,1.83,2.20,2.68,
23.20,3.83,4.53,5.35,6.25,7.23,8.30,9.40,10.6,13.35,16.35,19.9,
323.9,28.5,1.365,1.690,2.11,2.65,3.30,4.05,4.93,5.93,7.05,8.30,
49.70,11.15,12.90,14.5,18.0,22.3,27.0,32.4,38.7/
 DATA TAB60/1.01,1.035,1.06,1.09,1.125,1.16,1.195,1.235,1.275,1.32,
11.36,1.41,1.46,1.51,1.64,1.78,1.94,2.10,2.33,1.095,1.18,1.285,
21.40,1.545,1.705,1.890,2.09,2.305,2.545,2.81,3.1,3.4,3.73,4.47,
35.27,6.2,7.16,8.25,1.185,1.345,1.546,1.795,2.09,2.425,2.82,3.25,
43.730,4.25,4.85,5.49,6.19,6.90,8.5,10.3,12.3,14.5,16.90,1.295,
51.56,1.89,2.31,2.81,3.40,4.11,4.89,5.70,6.65,7.65,8.75,9.9,11.2,
614.0,17.35,20.65,24.5,28.75,1.42,1.8,2.31,2.83,3.73,4.63,5.65,
76.8,8.1,9.55,11.15,12.9,14.75,16.65,21.1,26.25,31.8,38.0,44.9,
81.565,2.08,2.79,3.695,4.79,6.05,7.50,9.25,11.1,13.2,15.45,17.90,
920.5,23.25,29.5,36.7,44.60,53.6,63.0,1.91,2.8,4.03,5.57,7.45,9.7,
012.25,15.1,18.45,22.0,26.0,30.2,35.0,40.0,50.8,63.5,77.5,92.5,
1110.0,2.325,3.64,5.5,7.85,10.7,14.15,18.0,22.5,27.4,33.0,39.0,
245.7,52.9,60.0,77.0,96.0,115.0,135.0,165.0,2.8,4.635,7.2,10.5,
314.5,19.4,25.0,31.0,38.0,45.7,54.3,63.0,73.5,83.5,107.5,135.0,
4165.0,190.0,230.0/
 END
```

21. SUBROUTINE PCHEK

a. General

Subroutine PCHEK is used to check the pressure. The pressure must be in the range 0.0001 to 100 atmospheres. Pressures out of range are set to the appropriate limit, and a message is printed on logical device 6. Two parameters, P and IFLAG, are input to subroutine PCHEK through the argument list.

b. Flow Chart



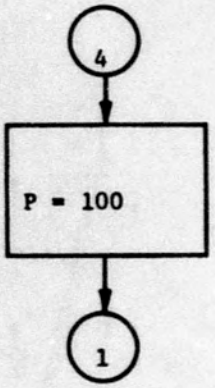
IS THE PRESSURE LESS THAN 10^{-4} ATMOSPHERES?

IS THE PRESSURE GREATER THAN 100 ATMOSPHERES?

BACK TO THE CALLING PROGRAM

LIST MESSAGE, RESULTING PRESSURE AND IDENTIFICATION NUMBER, IFLAG

SET THE PRESSURE EQUAL TO THE LOWER PRESSURE LIMIT



SET THE PRESSURE EQUAL TO THE UPPER LIMIT

C. LISTING

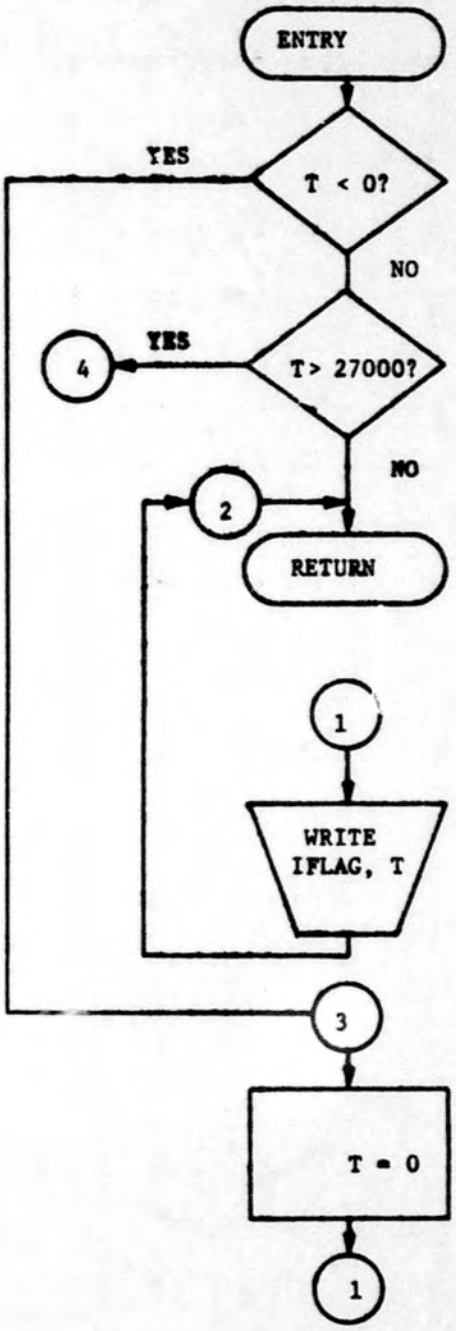
```
      SUBROUTINE PCHEK     (P,IFLAG)
C
C****          SUBROUTINE USED TO CHECK PRESSURE RANGE.
C
C****          RANGE LIMITS ARE 0.0001 AND 100 ATMOSPHERES.
C
C****          P HAS UNITS OF ATMOSPHERES.
C
C****          IFLAG IS AN IDENTIFICATION NUMBER.
C
      IF(P.LT.0.0001) GO TO  3
      IF(P.GT.100.) GO TO   4
2     RETURN
1     WRITE(6,5) IFLAG, P
5     FORMAT(41H0 PRESSURE IS OUT OF RANGE.  ENTRY NUMBERI10,2HP=E15.6)
      GO TO 2
3     P = 0.0001
      GO TO 1
4     P =100.
      GO TO 1
      END
```

22. SUBROUTINE TCHEK

a. General

Subroutine TCHEK is used to check the temperatures. The temperature must be in the range 0.0 to 27,000° Rankine. Temperatures out of range are set to the appropriate limit, and a message is printed on logical device 6. Two parameters, T and IFLAG, are input to subroutine TCHEK through the argument list.

b. Flow Chart



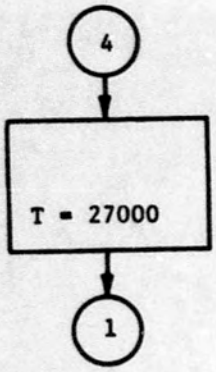
IS THE TEMPERATURE LESS THAN °R?

IS THE TEMPERATURE GREATER THAN 27000°R?

BACK TO THE CALLING PROGRAM

LIST MESSAGE AND RESULTING TEMPERATURE.
LIST ENTRY IDENTIFICATION NUMBER, IFLAG

FORCE THE TEMPERATURE TO 0°R



SET THE TEMPERATURE TO 27000°R

C. LISTING

```
      SUBROUTINE TCHEK(T,IFLAG)
C
C****          SUBROUTINE USED TO CHECK THE TEMPERATURE RANGE.
C
C****          RANGE LIMITS ARE 0 AND 27000 DEGREES RANKINE.
C
C****          T HAS UNITS OF DEGREES RANKINE.
C
C****          IFLAG IS AN IDENTIFICATION NUMBER.
C
      IF(T.LT.0.) GO TO 3
      IF(T.GT.27000.) GO TO 4
2     RETURN
1     WRITE(6,5) IFLAG,T
5     FORMAT(44H0 TEMPERATURE IS OUT OF RANGE.  ENTRY NUMBERI10,2HT=E15.
     16)
      GO TO 2
3     T=0.
      GO TO 1
4     T=27000.
      GO TO 1
      END
```

SECTION V

PROGRAM INPUT-OUTPUT

1. INTRODUCTION

This section discusses in detail the input necessary to run all the options of the flow field and turbulent boundary layer computer program and presents selected pages of the output for the sample case. The input is treated at length because a complete understanding of this portion of the program is necessary in order to use the program successfully. Sample input data cases to exercise most of program options are included to assist the user in setting up his particular case.

2. INPUT

a. Input Forms

The input follows two basic forms or groupings. One form is fixed formats with data corresponding to all input statements. The second form is symbolic input, conforming to specified formats but not necessarily including all of the input variables. Symbolic input is useful in setting up multiple cases, offering only those variables which change in value from case to case. In general, the data required for the chemistry library and running of the real gas segments and subroutines follows the fixed format form. The data required for the main boundary layer and flow field calculations and the mass loss calculations is input symbolically.

b. Input Cards

(1) Card Set 1

The cards in this set provide the data for Segment TWO, which is the input for the boundary layer solution. Not all of the cards to be described below are required for each calculation option. The optional data is identified as it is discussed. For this set of cards, the format restrictions of Subroutine GINPUT are applicable. In the description that follows, the following data fields and formats must be observed:

Field 1 - Column 1-6 - Format (A6)*
Field 2 - Column 7-10 - Format (I4)
Field 3 - Column 11-14 - Format I4)**
Field 4 - Column 15-20 - Format (I5)
Field 5 - Column 21-70 - Format (either I10 or 5E10.0)
Field 6 - Column 73-76 - Format (A4)

Card 1

Field 1	NBCD
Field 2	Blank
Field 4	1
Field 5	Value of NBCD
Field 6	T716

NBCD is the number of BCD data cards to follow. Only two values are permitted (0 or 1) for this program. The routine name, T716, must appear on all cards in this set.

Card 2

Field 1	TITLE
Field 2	1
Field 3	Sequence number
Field 4	5
Field 5	Title for masthead, any 50 alphanumeric characters, columns 21 to 70
Field 6	T716

TITLE is 50 (or less) alphanumeric characters that will be printed as a masthead to identify the particular problem being run.

Card 3

Field 1	NFIXED
Field 2	Blank
Field 3	Sequence number
Field 4	1
Field 5	Value of NFIXED
Field 6	T716

NFIXED is the number of integer data cards to follow; similar to NBCD but can be >1; maximum is 9.

---

*Field 1 is left justified; the others are right justified.
**Use of this field is optional.

Card 4

Field 1	IOPT
Field 2	6
Field 3	Sequence number
Field 4	1
Field 5	Value of IOPT
Field 6	T716

IOPT controls the manner in which ablation mass loss rate and integrated mass loss are input to the program. IOPT = 1; single values of EM and EMDOT (see cards 25 and 26) are read in as data. IOPT = 2; EM and EMDOT are calculated in the mass loss segment (Segment FOUR). When IOPT = 2 and NABL2 = 0, WABL1 and WABL2 are also calculated in Segment FOUR.

Card 5

Field 1	IBOD
Field 2	7
Field 3	Sequence number
Field 4	1
Field 5	Value of IBOD
Field 6	T716

IBOD = 0 for flat plate boundary layer computation. IBOD = 1 for cone.

Card 6

Field 1	IEND
Field 2	8
Field 3	Sequence number
Field 4	1
Field 5	Value of IEND
Field 6	T716

IEND controls the type of boundary layer computations to be made. IEND = 1 for the first of multiple boundary layer cases. IEND = 2 for the multiple boundary layer cases following the first. IEND = 3 for a single boundary layer case using the large chemical composition program. IEND = 4 for multiple chemical composition cases using Segment EIGHT. For this type of run, the only data required is IEND = 4 and IBOD = 1. Omit the remaining data in this card set.

Card 7

Field 1	NABL1
Field 2	9
Field 3	Sequence number
Field 4	1
Field 5	Value of NABL1
Field 6	T716

NABL1 is the number of compounds in the first ablation material. Two ablation materials are permitted. If a "clean air" calculation is desired, set NABL1, EM, and EMDOT = 0.

Card 8

Field 1	NABL2
Field 2	10
Field 3	Sequence number
Field 4	1
Field 5	Value of NABL2
Field 6	T716

NABL2 is the number of compounds in the second ablator. If only one ablator is considered, set NABL2 = 0.

Card 9

Field 1	IREAD
Field 2	11
Field 3	Sequence number
Field 4	1
Field 5	Value of IREAD
Field 6	T716

IREAD controls the reading of ablator composition is Subroutine INPUT. IRFAD = 1 when data is to be read. IREAD = 0 when data is not to be read (used only when IEND = 2 or clean air is desired).

Card 10

Field 1	KODE
Field 2	12
Field 3	Sequence number
Field 4	1
Field 5	Value of KODE
Field 6	T716

Card 10 (cont'd)

KODE is the number of species in the atmosphere model, i.e., KODE = 2 for O_2, N_2, or KODE = 3 for O_2, N_2, Argon. Any atmosphere model having less than four species is acceptable. Omit for clean air.

Card 11

Field 1	IHEMIS
Field 2	13
Field 3	Sequence number
Field 4	1
Field 5	Value of IHEMIS
Field 6	T716

IHEMIS specifies the nose shape (spherical or elliptical) to the shock shape calculation. IHEMIS = 0 for a hemispherical nose; IHEMIS = 1 for an elliptical nose. If IHEMIS = 0, omit cards 19-23; if IHEMIS = 1, the data on cards 19-23 must be included to specify the ellipse.

Card 12

Field 1	KODEW
Field 2	14
Field 3	Sequence number
Field 4	1
Field 5	Value of KODEW
Field 6	T716

KODEW controls the mixing option, homogenous or preferential. KODEW = 0 is homogenous mixing. WABL1 and WABL2 are obtained from the mass loss routine or from input data. Also KODEW = 0 if NABL2 = 0 or for clean air. KODEW = 1 is preferential mixing. WABL1 and WABL2 are calculated by Subroutine FCALC (see Section IV.16 and Appendix).

Card 13

Field 1	NFLOAT
Field 2	Blank
Field 3	Sequence number
Field 4	1
Field 5	Value of NFLOAT
Field 6	T716

NFLOAT is the number of floating point data cards to follow. Similar to NBCD and NFIXED. Maximum is 17.

Card 14

Field 1	ANTL
Field 2	15
Field 3	Sequence number
Field 4	1
Field 5	Value of ANTL
Field 6	T716

ANTL is the cone half-angle in degrees. Omit if IBOD = 0.

Card 15

Field 1	RN
Field 2	16
Field 3	Sequence number
Field 4	1
Field 5	Value of RN
Field 6	T716

RN is the radius of the hemispherical nose cap in inches. Must be >0.0. Omit if IBOD = 0.

Card 16

Field 1	ALT
Field 2	17
Field 3	Sequence number
Field 4	1
Field 5	Value of ALT
Field 6	T716

ALT is the vehicle altitude in feet. Omit if IBOD = 0.

Card 17

Field 1	VEL
Field 2	18
Field 3	Sequence number
Field 4	1
Field 5	Value of VEL
Field 6	T716

VEL is the vehicle free-stream velocity in ft/sec. Omit if IBOD = 0.

Card 18

Field 1	XINCH
Field 2	19
Field 3	Sequence number
Field 4	1
Field 5	Value of XINCH
Field 6	T716

XINCH is the axial distance in inches from the nose tip to the body station of interest.

Card 19

Field 1	Y1
Field 2	20
Field 3	Sequence number
Field 4	1
Field 5	Value of Y1
Field 6	T716

Y1 is the body radius in inches at the cone-nose junction for an elliptical nose. Omit if IHEMIS = 0 or IBOD = 0.

Card 20

Field 1	SMXB
Field 2	21
Field 3	Sequence number
Field 4	1
Field 5	Value of SMXB
Field 6	T716

SMXB is the body station location (in inches) of the stagnation point for an elliptical nose. Omit if IHEMIS = 0 or IBOD = 0.

Card 21

Field 1	XMXB
Field 2	22
Field 3	Sequence number
Field 4	1
Field 5	Value of XMXB
Field 6	T716

XMXB is the body station (in inches) at which Y1 is determined. Omit if IHEMIS = 0 or IBOD = 0.

Card 22

Field 1	DYDX1
Field 2	23
Field 3	Sequence number
Field 4	1
Field 5	Value of DYDX1
Field 6	T716

DYDX1 is the slope at the cone-ellipse junction, or at Y1. Omit if IHEMIS = 0 or IBOD = 0.

Card 23

Field 1	CP10
Field 2	24
Field 3	Sequence number
Field 4	1
Field 5	Value of CP10
Field 6	T716

CP10 is the nose pressure drag computed by reference 5, the nose deformation program. Omit if IHEMIS = 0 or IBOD = 0.

Card 24

Field 1	TW
Field 2	25
Field 3	Sequence number
Field 4	1
Field 5	Value of TW
Field 6	T716

TW is the wall temperature in degrees Rankine.

Card 25

Field 1	EM
Field 2	26
Field 3	Sequence number
Field 4	1
Field 5	Value of EM
Field 6	T716

EM is the vehicle surface integrated mass loss in lbm/sec. Omit if IOPT = 2 or clean air calculation is to be made.

Card 26

Field 1	EMDOT
Field 2	27
Field 3	Sequence number
Field 4	1
Field 5	Value of EMDOT
Field 6	T716

EMDOT is the local mass injection rate in lbm/ft^2-sec. Omit if IOPT = 2 or for clean air.

Card 27

Field 1	WA
Field 2	28
Field 3	Sequence number
Field 4	1
Field 5	Value of WA
Field 6	T716

WA is the average molecular weight of the free stream gas. Omit if program uses molecular weight of air = 28.962.

Card 28

Field 1	WABL1
Field 2	29
Field 3	Sequence number
Field 4	1
Field 5	Value of WABL1
Field 6	T716

WABL1* is the integrated mass loss (lbm/sec) from ablator number 1. Omit if IOPT = 2, KODEW = 1, or NABL2 = 0. Also omit for clean air.

Card 29

Field 1	WABL2
Field 2	30
Field 3	Sequence number
Field 4	1
Field 5	Value of WABL2
Field 6	T716

WABL2* is the integrated mass loss (lbm/sec) from ablator number 2. Omit if IOPT = 2, NABL2 = 0, or KODEW = 1. Also omit for clean air.

---

*Note that WABL1 + WABL2 = EM.

Card 30

Field 1	YBB
Field 2	31
Field 3	Sequence number
Field 4	1
Field 5	Value of YBB
Field 6	T716

YBB is the body radius in inches at station XINCH. Omit if IBOD = 0.

Card 31

Field 1	ENDFIL
Field 6	T716

ENDFIL signifies the end of the data for card set 1.

(2) Card Set 2

Data in this card set is included only if IEND = 4, the option for multiple chemistry calculations only. Data for this option must be obtained from a calculation where IEND = 1, 2, or 3.

Card 1

Problem title. Format (5A10).

Card 2

Values of wall mass fraction (FS), wall static enthalpy (HW), adiabatic wall enthalpy (HAW), and boundary layer edge static enthalpy (HE). Enthalpy units are Btu/lbm. Format (10X, 4E15.0).

Card 3

Values of wall density in lbm/ft^3 (RHOW), parameter (C5), parameter (C6), and boundary layer edge velocity in ft/sec (UE). Format (10X, 4E15.0).

Card 4

Values of parameter (B), vehicle circumference in feet (CIR), wall temperature in degrees Kelvin (T), and wall pressure in atmospheres (P). Format (10X, 4E15.0).

Card 5

Number of species in atmosphere (KODE), number of compounds in number 1 ablator (NABL1), number of compounds in number 2 ablator (NABL2), value of the mixing indicator (KODEW). Format (10X, 4I5).

Card 6

Values of WABL1 and WABL2. Format (10X, 2E15.0).

(3) Card Set 3

Omit this set if IOPT = 1 or IEND = 4. The format restrictions of Subroutine GINPUT apply to this set and the card fields are the same as those of Card Set 1.

Card 1

Field 1	NBCD
Field 2	Blank
Field 3	Sequence number
Field 4	1
Field 5	0
Field 6	M367

Card 2

Field 1	NFIXED
Field 2	Blank
Field 3	Sequence number
Field 4	1
Field 5	2
Field 6	M367

Card 3

Field 1	IRETML
Field 2	30
Field 3	Sequence number
Field 4	1
Field 5	Value of IRETML
Field 6	M367

IRETML controls the transition criteria used. IRETML = 1, transition from laminar to turbulent heating based on local Reynolds number. The critical Reynolds number is input as RE. IRETML = 2, transition based on momentum thickness Reynolds number divided by local Mach number, Re_θ/M_L. The critical value is input as RETMLC.

Card 4

Field 1	NMAT
Field 2	43
Field 3	Sequence number
Field 4	1
Field 5	Value of NMAT
Field 6	M367

Card 4 (cont'd)

NMAT is the number of different ablator materials covering the surface (maximum of 2).

Card 5

Field 1	NFLOAT
Field 2	Blank
Field 3	Sequence number
Field 4	1
Field 5	14
Field 6	M367

Card 6

Field 1	RAD
Field 2	44
Field 3	Sequence number
Field 4	1
Field 5	Value of RAD
Field 6	M367

RAD is the vehicle nose radius in inches (must be >0.0).

Card 7

Field 1	RETMLC
Field 2	45
Field 3	Sequence number
Field 4	1
Field 5	Value of RETMLC
Field 6	M367

RETMLC is the critical value of momentum thickness Reynolds number divided by local Mach number at which transition occurs. Used if IRETML = 2, see Card 3.

Card 8

Field 1	RE
Field 2	48
Field 3	Sequence number
Field 4	3
Field 5	Value of RE, X_L, and θ_c
Field 6	M367

Card 8 (cont'd)

RE is the critical value of local Reynolds number for transition. Used if IRETML = 1, see Card 3. X_L is the body length in inches. θ_c is the cone half-angle in degrees.

Card 9

Field 1	XL
Field 2	52
Field 3	Sequence number
Field 4	2
Field 5	Value of XL(1) and XL(2)
Field 6	M367

XL(1) and XL(2) are, respectively, the length of the nose section and the total body length.

Card 10

Field 1	EMO
Field 2	72
Field 3	Sequence number
Field 4	2
Field 5	Value of EMO(1) and EMO(2)
Field 6	M367

The constants for each ablation material where the emissivity is expressed by ε = EMO + EMI* (TEMPERATURE).

Card 11

Field 1	EMI
Field 2	77
Field 3	Sequence number
Field 4	2
Field 5	Values of EMI(1) and EMI(2)
Field 6	M367

The EMI values for each ablator, as above.

Card 12

Field 1	TI
Field 2	82
Field 3	Sequence number
Field 4	2
Field 5	Values of TI(1) and TI(2)

Card 12 (cont'd)

Field 6	M367

Wall temperature (°R) for heating rate calculation, usually ablation temperature.

Card 13

Field 1	AB
Field 2	87
Field 3	Sequence number
Field 4	2
Field 5	Values of AB(1) and AB(2)
Field 6	M367

AB values for each ablator used in Q* calculation.

$$Q^*_{Lam} = AB + ETL(\Delta H) + ZTL(H_S)$$

$$Q^*_{Turb} = AB + ETT(\Delta H) + ZTT\ (H_S)$$

Card 14

Field 1	ETL
Field 2	92
Field 3	Sequence number
Field 4	2
Field 5	Values of ETL
Field 6	M367

Card 15

Field 1	ETT
Field 2	97
Field 3	Sequence number
Field 4	2
Field 5	Values of ETT
Field 6	M367

Card 16

Field 1	ZTL
Field 2	102
Field 3	Sequence number
Field 4	2
Field 5	Values of ZTL
Field 6	M367

Card 17

Field 1	ZTT
Field 2	107
Field 3	Sequence number
Field 4	2
Field 5	Values of ZTT
Field 6	M367

Card 18

Field 1	TRAD
Field 2	126
Field 3	Sequence number
Field 4	2
Field 5	Values of TRAD
Field 6	M367

TRAD is the surface radiation temperature of each ablation material, usually the ablation temperature.

Card 19

Field 1	AMODEL
Field 2	131
Field 3	Sequence number
Field 4	1
Field 5	Values of AMODEL
Field 6	M367

AMODEL = 1.0, $Q_{net} = Q_{conv} - Q_{RAD}$. AMODEL = 0, $Q_{net} = Q_{conv}$.

Card 20

Field 1	ENDFIL
Field 6	M367

(4) Card Set 4

Chemical Model Data for thermochemistry calculation; omit if IEND = 4.

Card 1

Data for atmosphere model based on species contained in Card Set 6. The number of lines of data must equal KODE (see Set 1, Card 10). Data required is the specie formula (FORM), identification number (ID), weight percent of specie in atmosphere model (WT), and molecular weight of specie (EMAIR). Format (3X,A3, I4, 2E10.0).

Card 2

Data for first ablator; number of lines of data must equal NABL1. Required data is formula, identification number, weight percent, and molecular weight. Format (3X, A3, I4, 2E10.0).

Card 3

Data for second ablator; number of lines of data must be equal to NABL2. Same type of data as above.

(5) Card Set 5

Flow field data for flat plate option. Omit if IBOD = 0.

Card 1

Six pieces of data to specify flow field, ΔY for iteration increment, and plate width. Format (6E12.0). Enter edge velocity (UEP) in ft/sec; edge enthalpy (HEP) in B/lb; edge density (RHOEP) in slug/ft^3; edge pressure (PP) in lb/ft^2; normal distance increment (DELY) in inches; plate width (CIR) in ft.

(6) Card Set 6

Library data for the thermochemistry calculations performed by Subroutine CHEMS. Omit if IEND = 2 or 4 or NABL1 = 0 (clean air).

Card 1

Number of elements and the number of species in the library to follow. Format (2I2). Maximum is 10 for number of species and 25 for number of elements.

Card 2

The number of temperature values for which the thermochemical data is tabulated. Format (I2). Value is normally 39; maximum is 39.

Card 3

The temperature array for which values of thermochemical data are tabulated. Format (20X, 5E7.0). Repeat cards until the entire array is entered, normally 8 cards.

Card 4

The chemical symbol, formula vector, and molecular weight of the specie for which data is being tabulated on cards 5, 6, and 7. See reference 2 or 9 for a discussion on how to set up the formula vector. Format (12X, A4, 10I2, 10X, E10.0).

Card 5

Free energy array. The value of $-(F - H_o)/T$ for the specie being tabulated. The values correspond to the temperature array of Card 3. Enter 5 values per card until all values are entered. Format (20X, 5E12.0). Units are cal/mol-°K.

Card 6

Enthalpy array. The values of $(H - H_o)/T$ for the specie being tabulated. Enter as in Card 5. Units are cal/mole-°K.

Card 7

Heat of formation for the specie being tabulated. Units are kcal/mole. Format (20X, E12.0).

Cards 4 through 7 are repeated for each specie in the library. The number of packets will equal the number of species designated on Card 1. The elements must occupy the last positions in the assembled group of packets.

Card 8

The array of concentration guesses for the elemental species. Required to begin the iteration procedure. On each card, enter the specie formula and the guess in format (A4, E12.0). One card is required for each element. A good initial guess is 10^{-15}.

(7) Card Set 7

A control card used to change the chemistry model from the short version to the long version. Used by Subroutine INPUT in preparing a tape for Segment SIX after the real gas boundary layer has been solved.

Card 1

Contains the number of species in the atmosphere model, NAIR; the number of species in each of the two ablators, NABL1 and NABL2. Format (10X, 3I5).

(8) Card Set 8

The expanded atmosphere and ablation chemical model used for the full chemistry calculations. The data is used by Subroutine INPUT to prepare a data tape for Segment SIX.

Card 1

Data for atmosphere model based on the species contained in Card Set 9. The number of lines of data must equal NAIR (see Card Set 7). Data required is the specie formula (FORM), identification number (ID), weight percent of specie in atmosphere model (WT), and molecular weight of the specie (EMAIR). Format (3X, A3, I4, 2E10.0).

Card 2

Data for first ablator, number of lines of data must equal NABL1. Required data is formula, identification number, weight percent, and molecular weight. Format same as Card 1.

Card 3

Data for second ablator, number of lines of data must equal NABL2. Same type of data as above.

(9) Card Set 9

Library and control data for the large thermochemistry calculation performed by Segment SIX.

Card 1

Two control variables. The first, NCODE, determines if library data is to be read. If NCODE = 4, data will not be read. This is used if IEND = 4 and the case being run is not the first case in the series. The second variable is NNOD and in the sequence number of the electron which identifies the specie concentration to be punched by Segment SEVEN. Format (2I2).

Card 2

Number of elements and number of species in the library to follow. Symbols are MAXE and MAXS. Format (2I2). Maximum number of elements is 15, maximum number of species is 70.

Card 3

The number of values in the temperature array for which thermochemical data is tabulated. The normal and maximum value is 39. Format (I2).

Card 4

The temperature array for which thermochemical data is tabulated. Format (20X, 5F7.0). Repeat cards until the entire array is entered, normally 8 cards. Units are °K.

Card 5

The chemical symbol, formula vector, and molecular weight of the specie for which data is being tabulated on Cards 6, 7, and 8. See references 2 and 9 for a discussion on how to set up the formula vector. Format (12X, A4, 15I2, E10.0).

Card 6

Free energy array. The values of $-(F - H_o)/T$ for the specie being tabulated. The values correspond to the temperature array of Card 4. Enter five values per card until all values are entered. Format (20X, 5E12.0). Units are cal/mole-°K.

Card 7

Enthalpy array. The values of $(H - H_o)/T$ for the specie being tabulated. Enter as in Card 6. Units are Cal/mole-°K.

Card 8

Heat of formation for the specie being tabulated. Units are kcal/mole. Format (20X, E12.0).

Cards 5 through 8 are repeated for each specie in the library. The number of packets will equal the number of species designated on Card 2. The elements will occupy the last positions in the assembled group of packets.

Card 9

The number of elements and the number of species to be considered in the large chemistry calculation, since the entire library need not be used. Format (2I2).

Card 10

The specie identification numbers for the species considered for the large chemistry calculation. The number of entries must correspond to the number of species specified on Card 9. Format (20I4). Use as many cards as required, maximum of 4.

Card 11

The array of concentration guesses for the elemental species. Required to start the iteration procedure. On each card, enter the specie (element) formula and the guess in format (A4, E12.0). One card is required for each element used in the calculation. A good initial guess is 10^{-15}.

c. Calculation Options

There are four basic options which may be set up:

(1) Complete calculation, including thermochemistry which yields electron concentration and collision frequency. Only one case at a time may be run with this option.

(2) Basic boundary layer calculation, exclusive of electron concentration profiles. Single or multiple cases may be run under this option.

(3) Separate thermochemistry calculations. This calculation is a continuation of the condensed calculations of (2). This option may be used with either a single case or with multiple cases.

(4) Clean air calculation. This can be considered a fourth option, since certain parts of the setup can be eliminated. Only one case at a time may be run with this option.

Table IX describes the necessary card sets to be used in the various options. It should be remembered that for multiple cases the input data for the first case is complete, followed by the case data for subsequent cases.

Table IX

DATA STRUCTURE

Case	Data card sets and order
Full boundary layer with thermochemistry	1, 3, 4, 5,* 6, 7, 8, 9
Basic boundary layer (first case)	1, 3, 4, 5,* 6
Basic boundary layer (additional cases	1, 3, 4
Thermochemistry only	1, 2, 8, 9
Clean air boundary layer	1

*Used only for flat plate option.

d. Sample Input Data

The following examples give the input data for running some of the options of the boundary layer program. The first example is a listing of the case data for the boundary layer portion of the program (designated T716 at the right), the mass loss calculations (designated M367 at the right), and the composition of the two ablators (less contaminants for this portion of the program). A typical packet of the chemical library data for one specie for the thermochemistry calculations is shown in the second example. A more detailed discussion of the setup for this portion of the program may be found in references 2 and 9. The third example shows the data necessary to run only the large version of the thermochemical calculations. Case data is read in (marked by T716) which transfers control to read in the data resulting from the boundary layer calculations and the ablator composition including contaminants. The thermochemistry library is not listed, but is essentially the same as the small versions. Example four is case data for a clean air case. As can be seen, the only data required is that for the boundary layer portion of the program, T716.

The sample case given here is a 7.5° half-angle case with a 0.5 inch nose radius. The boundary layer profile at an axial station of 50 inches is the profile of interest. The flight velocity is 15,000 ft/sec at an altitude of 5000 feet mean sea level. The nose tip is 9 inches long and is composed of graphite; the afterbody is phenolic silica. Contaminants in the graphite are 20 ppm sodium and 100 ppm calcium. In the phenolic silica, the contaminant is 100 ppm sodium.

EXAMPLE 1. BOUNDARY LAYER CASE DATA

```
NNCD          0     1          1                                                  T716
TITLE     1   1     5  FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY              T716
NFIXED        2     1          9                                                  T716
IOPT      6   3     1          2                                                  T716
IROD      7   4     1          1                                                  T716
IEND      8   5     1          3                                                  T716
NARL1     9   6     1          3                                                  T716
NARL2     10  7     1          4                                                  T716
IREAD     11  8     1          1                                                  T716
KODE      12  9     1          3                                                  T716
IHEMIS    13  10    1          0                                                  T716
KODEW     14  10    1          0                                                  T716
NFLOAT        11    1          7                                                  T716
ANTL      15  12    1 7.5                                                         T716
RN        16  13    1 0.50                                                        T716
ALT       17  14    1 5000.0                                                      T716
VEL       18  15    1 15000.                                                      T716
XINCH     19  16    1 50.0                                                        T716
TW        25  22    1 5000.0                                                      T716
YMIN      31  28    1 7.45943                                                     T716
ENDFIL                                                                          T716
NNCD          1     1          0                                                  M367
NFIXED        2     1          2                                                  M367
IHETML    30  3     1          2                                                  M367
NMAT      43  4     1          2                                                  M367
NFLOAT        5     1         14                                                  M367
RAD       44  6     1 0.50                                                        M367
RETMLC    45  7     1 110.0                                                       M367
RF        48  8     3  1.0          50.0          7.5                            M367
XI        52  9     2  9.0          50.0                                         M367
EMO       72  10    2  0.8          0.8                                          M367
EMI       77  11    2  0.0          0.0                                          M367
TI        82  12    2  6400.0       5000.0                                       M367
AB        87  13    2  2000.0       2086.0                                       M367
ETL       92  14    2  2.0          0.517                                        M367
ETT       97  15    2  2.0          0.0                                          M367
ZTI       102 16    2  0.0        0.320                                          M367
ZTT       107 17    2  0.0          0.0                                          M367
TRAD      126 18    2  6400.0       5000.0                                       M367
AMONEL    131 19    1  0.0                                                       M367
ENDFIL                                                                          M367
   N2       8 78.0881    28.016
   O2       9 20.9795    32.0
   A       23  0.9324    39.994
   C       20 99.3892    12.011
   H       24  0.1802     1.008
   O       22  0.4306    16.0
   C       20 19.8066    12.011
   H       24  1.4271     1.008
   O       22  3.7713    16.000
  SIO2     19 74.9950    60.090
```

EXAMPLE 2. THERMOCHEMISTRY LIBRARY PACKET

```
                O2          2                                  32.000
O2  -(F-H0)/T       0.506884E 020.538017E 020.560979E 020.579308E 020.594640E 02
O2  -(F-H0)/T       0.607868E 020.619530E 020.629974E 020.639015E 020.647764E 02
O2  -(F-H0)/T       0.655840E 020.663350E 020.670379E 020.676989E 020.683233E 02
O2  -(F-H0)/T       0.689150E 020.694775E 020.700138E 020.705261E 020.710166E 02
O2  -(F-H0)/T       0.714870E 020.719390E 020.723738E 020.727927E 020.731966E 02
O2  -(F-H0)/T       0.735866E 020.739635E 020.743281E 020.746810E 020.750229E 02
O2  -(F-H0)/T       0.753543E 020.756758E 020.759879E 020.762911E 020.765856E 02
O2  -(F-H0)/T       0.768720E 020.771506E 020.774217E 020.776857E 020.779427E 02
O2  (H-H0)/T        0.750328E 010.785394E 010.811068E 010.831960E 010.850109E 01
O2  (H-H0)/T        0.866200E 010.880473E 010.893084E 010.907500E 010.919900E 01
O2  (H-H0)/T        0.931120E 010.943459E 010.953473E 010.962934E 010.971857E 01
O2  (H-H0)/T        0.980247E 010.988101E 010.995415E 010.100219E 020.100842E 02
O2  (H-H0)/T        0.101411E 020.101928E 020.102394E 020.102809E 020.103175E 02
O2  (H-H0)/T        0.103495E 020.103769E 020.104000E 020.104189E 020.104338E 02
O2  (H-H0)/T        0.104449E 020.104524E 020.104565E 020.104573E 020.104551E 02
O2  (H-H0)/T        0.104501E 020.104424E 020.104321E 020.104196E 020.104196E 02
O2  HEAT FORMATION  0.          E 00
```

EXAMPLE 3. THERMOCHEMISTRY CALCULATION DATA

```
NHCD          1     1          0                                                    T716
NFIXED        2     1          2                                                    T716
IHOD     7    3     1          1                                                    T716
IEND     8    4     1          4                                                    T716
NFLOAT        5     1          0                                                    T716
ENDFIL                                                                              T716
      BOUNDARY LAYER THERMOCHENISTRY CALCULATION                                          1
           0.80476          336.696          4168.999        214.259                    2
           0.03553                3.9497E-07 19.519          14842.1                    3
           4.122          3.905              2778.           4.558                      4
             3    5    5    0                                                           5
           0.11525          7.83278                                                     6
   N2     8  78.0881 28.016
   O2     9  20.9795  32.0
   A     62  0.9324  39.994
   C     58  99.3773  12.011
   H     64  0.1802  1.008
   O     60  0.4305  16.0
   NA    63  0.002  22.991
   CA    69  0.01  40.08
   C     58 19.8041  12.011
   H     64  1.4246  1.008
   O     60  3.7688  16.000
  SIO2   22 74.9925  60.090
   NA    63  0.0100  22.991
```

EXAMPLE 4. CLEAN AIR CASE DATA

NRCD		1	1	1	T716
TITLE	1	2	5	CLEAN AIR CHECK CASE	T716
NFIXED		3	1	3	T716
IBOD	7	4	1	1	T716
IEND	8	5	1	3	T716
IHEMIS	13	6	1	0	T716
NFLOAT		7	1	6	T716
ANTL	15	8	1	7.5	T716
RN	16	9	1	0.5	T716
ALT	17	10	1	5000.	T716
VEL	18	11	1	15000.	T716
XINCH	19	12	1	50.	T716
TW	25	13	1	2800.	T716
ENDFIL					T716

3. OUTPUT

Nominal output sheets for the sample problem given in the preceding paragraph are presented in this section. Unless otherwise indicated, a definition of the symbols appearing on the output is listed following each output set. In the order in which the output is generated, it is

1. Listing of case data input parameters to the boundary layer portion of of the program. (Symbolic input section) (Symbols defined in Section V.2.b)

2. Results of the shock shape calculation.

3. Listing of case data input parameters to the mass loss calculation. (Symbolic input section) (Symbols defined in Section V.2.b)

4. Mass loss calculations. This will consist of 2N-1 pages, where N is the number of ablators. The last page of this set contains, additionally, the mass loss rate, the turbulent heating rate, and the turbulent shear stress. (Symbols defined in reference 1)

5. Inviscid flow field profiles.

6. Preliminary boundary layer profiles, considering ideal gas relationships for the equation of state. (Symbols defined for item 7)

7. Final real gas boundary layer profiles. Values are tabulated at every 10 points (or at 0.1 intervals of the velocity ratio through the boundary layer).

8. Parameters for use in running the thermochemistry portion separately for the electron concentration profile. (Symbols defined for item 7)

9. Thermochemistry calculations from the small version of that routine in the boundary layer program. One page at each of the points tabulated in 7. The output from the thermochemistry section (large version) to give electron concentration is the same, but with more species, and will not be repeated here. (Symbols defined in references 2 and 9)

10. Final tabulation of the real gas boundary layer profiles, including electron concentration and collision frequency. This tabulation results when the large version of the thermochemistry calculations is used.

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TITLE FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

IOPT	= 2	CP10	= 0.
IBOD	= 1	TW	= 5.00000000E+03
IEND	= 3	EM	= 0.
NABL1	= 3	EMDOT	= 0.
NABL2	= 4	WA	= 0.
IREAD	= 1	WABL1	= 0.
KODE	= 3	WABL2	= 0.
IHEMIS	= 0	YBB	= 7.45943000E+00
KODEW	= 0		

ANTL = 7.50000000E+00

KN = 5.00000000E-01

ALT = 5.00000000E+03

VEL = 1.50000000E+04

AINCH = 5.00000000E+01

Y1 = 0.

SMXB = 0.

AMXB = 0.

DYDX1 = 0.

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

SHOCK SHAPE FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY
ALT = 5000.00FT MACH NO = 13.67269 P-LOC = 4.771475 ATM

A1 = .86000 B1 = .40500 CD = .888995

X = 30.0000 IN RN = .500000 IN

X/R	Y/R	ANGLE
	0.	90.000000
.002655	.125103	87.000000
.008552	.200902	84.000000
.017036	.265587	81.000000
.027932	.324467	78.000000
.041223	.379866	75.000000
.056997	.433131	72.000000
.075428	.485178	69.000000
.096778	.536710	66.000000
.121405	.588323	63.000000
.149780	.640557	60.000000
.182515	.693946	57.000000
.220404	.749038	54.000000
.264480	.806435	51.000000
.316096	.866815	48.000000
.377046	.930979	45.000000
.449749	.999891	42.000000
.537522	1.074757	39.000000
.645014	1.157112	36.000000
.778918	1.248975	33.000000
.949155	1.353073	30.000000
1.170989	1.473207	27.000000
1.468964	1.614877	24.000000
1.884760	1.786399	21.000000
2.494245	2.001060	18.000000
3.448665	2.281647	15.000000
5.069670	2.671217	12.000000
8.344795	3.263421	9.000000
15.982729	4.745474	8.212495
23.620662	5.831206	7.993608
31.258596	6.895779	7.889808
38.896530	7.951012	7.850248
46.534464	9.003676	7.848904
54.172397	10.057918	7.871767
61.810331	11.116431	7.910728
69.448265	12.181036	7.960797
77.086199	13.253013	8.018753
84.724132	14.333277	8.082427
92.362066	15.422502	8.150315
100.000000	16.521186	8.221342

EPS = .328592 CPMAX = 1.892027 CP10 = .002726
KC = 1.784644 KS = 2.169794 GAMMA-E = 1.393990
THETA-S = 5.051394 ETA-C = 9.131183 P-SHOCK = 7572.157
THETA = 7.500000 RN = .500000 P-BODY = 10097.299
GSTAG = 1.242627 RHO0 = 2.0482E-03 P-ZERO = 1760.870

SYMBOLS USED ON OUTPUT FROM SHOCK SHAPE

ALT	Altitude, feet
MACH NO	Mach number
P-LOC	Local pressure, atm
A_1, B_1	Coefficients in Mirels-Thornton shock shape correlation
CD	Nose drag coefficient
X	Local body station, inch
RN	Nose radius, inch
X/R	Normalized X-coordinate of shock wave
Y/R	Normalized Y-coordinate of shock wave
ANGLE	Shock wave angle
EPS	Density ratio across normal shock
CPMAX	Stagnation pressure coefficient
CP10	Newtonian bluntness drag
KC	Hypersonic similarity parameter, $M_\infty \sin(\theta_c)$
KS	Mirels-Thornton parameter, K
GAMMA-E	Effective ratio of specific heats behind normal shock
THETA-S	Flow angle behind shock at body station, degrees
ETA-C	Shock angle at body station, degrees
P-SHOCK	Pressure behind shock at body station, psf
THETA	Cone half-angle, degrees
P-BODY	Local pressure, psf
GSTAG	Stagnation value of ratio of specific heats
RHOO	Free stream density, lb/ft^3
P-ZERO	Free stream pressure, psf

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TITLA FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TITLB

TITLC

```
ITRETML=     2     4     0     0     0
 NOK  =      0     0     0     0     0     0
 I20  =      0
 I22  =      0
 NMAT =      2
 RAD  = 5.00000000E-01
RETMLC= 1.10000000E+02
 DTAU = 0.
 TH   = 0.
 RE   = 1.00000000E+00 5.00000000E+01 7.50000000E+00
 UELS = 0.
 AL   = 9.00000000E+00 5.00000000E+01 0.          0.          0.
 CO   = 0.             0.             0.          0.          0.
 CI   = 0.             0.             0.          0.          0.
 HU   = 0.             0.             0.          0.          0.
```

EMO =	8.00000000E-01	8.00000000E-01	0.	0.	0.
EMI =	0.	0.	0.	0.	0.
II =	6.40000000E+03	5.00000000E+03	0.	0.	0.
AB =	2.00000000E+03	2.08600000E+03	0.	0.	0.
ETL =	2.00000000E+00	5.17000000E-01	0.	0.	0.
ETT =	2.00000000E+00	0.	0.	0.	0.
ZTL =	0.	3.20000000E-01	0.	0.	0.
ZTT =	0.	0.	0.	0.	0.
CPO =	0.	0.	0.		
RHOS =	0.	0.	0.	0.	

TURBULENT BOUNDARY LAYER

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TIME 0. SECONDS ALTITUDE 5.000000E+03 FEET

ATMOSPHERE PARAMETERS.

	FREE STREAM	LOCAL	UNITS	LOCAL/FREE STREAM
FLOW SPEED	1.500000E+04	1.482045E+04	FT/SEC	9.880301E-01
TEMPERATURE	5.008615E+02	9.449069E+02	DEGREES R	1.886563E+00
PRESSURE	1.760870E+03	1.046908E+04	LB/FT**2	5.945403E+00
MACH NUMBER	1.365058E+01	9.819411E+00		7.193405E-01
SPECIFIC WEIGHT	6.589845E-02	2.076754E-01	LB/FT**3	3.151446E+00
VISCOSITY	1.181629E-05	1.890715E-05	LB*(SEC/FT)**3	1.600092E+00
ENTHALPY	1.209430E+02	2.267164E+03	BTU/LB	1.874572E+01
COMPRESSIBILITY	1.000000E+00	1.000000E+00		1.000000E+00
PRANDTL NUMBER	7.380000E-01	7.388981E-01		1.001217E+00
REYNOLDS NUMBER PER FOOT	8.365372E+07	1.627873E+08	1./FT	1.945966E+00

LAMINAR

ABLAT(1)	2.000000E+03	BTU/LB	ETA	2.000000E+00
ZETA	0.		Y	5.000000E-01
CK	5.750000E-01		RECOVERY FACTOR	8.595918E-01
RADIATION RATE	6.386522E+02	BTU/SEC/FT**2	ABLATION ENERGY	6.686009E+03 BTU/LB
DQUADT AT XHIGH	6.455871E+02	BTU/SEC-FT**2	TOTAL HEATING RATE	3.689021E+01 BTU/SEC
RADIATION CUTOFF	1.124305E-01	FT	TRANSITION POINT	1.100281E-01 FT
XLOW	0.	FT	XHIGH	1.100281E-01 FT
ADIABATIC WALL TEMP	6.6849 5E+03	DEGREES R	ECKERTS TEMP	6.179742E+03 DEGREES R
ECKERTS PRANDTL NO	6.430644E-01		ECKERTS REYNOLDS NO	7.828543E+06 1/FT
COMPRESSIBILITY	1.047321E+00		ECKERTS VISCOSITY	5.739904E-05 LB*(SEC/FT)**3
ECKERTS SPECIFIC WT	3.031964E-02	LB/FT**3	ADIABATIC ENTHALPY	3.994844E+03 BTU/LB
ECKERTS ENTHALPY	2.076224E+03	BTU/LB	WEIGHT LOSS	5.517524E-03 LB
ABLATION TEMP	6.600000E+03	DEGREES R	HEATING RATE AT X	2.462126E+02 BTU/SEC-FT**2

TURBULENT

ABLAT(1)	2.000000E+03	BTU/LB	ETA	2.000000E+00
ZETA	0.		Y	8.000000E-01
CK	3.404000E-02		RECOVERY FACTOR	9.040550E-01
RADIATION RATE	6.386522E+02	BTU/SEC/FT**2	ABLATION ENERGY	6.686009E+03 BTU/LB
DQUADT AT XHIGH	1.733331E+03	BTU/SEC-FT**2	TOTAL HEATING RATE	7.336937E+02 BTU/SEC
RADIATION CUTOFF	1.113982E+02	FT	TRANSITION POINT	1.100281E-01 FT
XLOW	1.100281E-01	FT	XHIGH	7.564717E-01 FT
ADIABATIC WALL TEMP	8.769383E+03	DEGREES R	ECKERTS TEMP	6.237136E+03 DEGREES R
ECKERTS PRANDTL NO	6.415386E-01		ECKERTS REYNOLDS NO	7.700410E+06 1/FT
COMPRESSIBILITY	1.049759E+00		ECKERTS VISCOSITY	5.768289E-05 LB*(SEC/FT)**3
ECKERTS SPECIFIC WT	2.997088E-02	LB/FT**3	ADIABATIC ENTHALPY	4.189700E+03 BTU/LB
ECKERTS ENTHALPY	2.114092E+03	BTU/LB	WEIGHT LOSS	1.097357E-01 LB
ABLATION TEMP	6.600000E+03	DEGREES R	HEATING RATE AT X	1.733331E+03 BTU/SEC-FT**2

TURBULENT BOUNDARY LAYER

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TIME 0. SECONDS ALTITUDE 5.000000E+03 FEET

ATMOSPHERE PARAMETERS.

	FREE STREAM	LOCAL	UNITS	LOCAL/FREE STREAM
FLOW SPEED	1.500000E+04	1.482045E+04	FT/SEC	9.880301E-01
TEMPERATURE	5.008615E+02	9.449069E+02	DEGREES R	1.886563E+00
PRESSURE	1.760870E+03	1.046908E+04	LB/FT**2	5.945403E+00
MACH NUMBER	1.365058E+01	9.819411E+00		7.193405E-01
SPECIFIC WEIGHT	6.589845E-02	2.076754E-01	LB/FT**3	3.151446E+00
VISCOSITY	1.181629E-05	1.890715E-05	LB*(SEC/FT)**3	1.600092E+00
ENTHALPY	1.209430E+02	1.452741E+03	BTU/LB	1.201178E+01
COMPRESSIBILITY	1.000000E+00	1.000000E+00		1.000000E+00
PRANDTL NUMBER	7.380000E-01	7.388981E-01		1.001217E+00
REYNOLDS NUMBER PER FOOT	8.365372E+07	1.627873E+08	1./FT	1.945966E+00

LAMINAR

ABLAT(1)	2.086000E+03 BTU/LB	ETA	5.170000E-01
ZETA	3.200000E-01	Y	5.000000E-01
CK	5.750000E-01	RECOVERY FACTOR	8.595918E-01
RADIATION RATE	2.379165E+02 BTU/SEC/FT**2	ABLATION ENERGY	5.193644E+03 BTU/LB
DQDADT AT XHIGH	1.014363E+03 BTU/SEC-FT**2	TOTAL HEATING RATE	5.275589E+01 BTU/SEC
RADIATION CUTOFF	1.837665E+00 FT	TRANSITION POINT	1.010947E-01 FT
XLOW	0. FT	XHIGH	1.010947E-01 FT
ADIABATIC WALL TEMP	8.688907E+03 DEGREES R	ECKERTS TEMP	5.479176E+03 DEGREES R
ECKERTS PRANDTL NO	6.916892E-01	ECKERTS REYNOLDS NO	9.692904E+06 1/FT
COMPRESSIBILITY	1.017561E+00	ECKERTS VISCOSITY	5.381530E-05 LB*(SEC/FT)**3
ECKERTS SPECIFIC WT	3.519640E-02 LB/FT**3	ADIABATIC ENTHALPY	3.994844E+03 BTU/LB
ECKERTS ENTHALPY	1.669012E+03 BTU/LB	WEIGHT LOSS	1.015778E-02 LB
ABLATION TEMP	5.000000E+03 DEGREES R	HEATING RATE AT X	1.573250E+02 BTU/SEC-FT**2

TURBULENT

ABLAT(1)	2.086000E+03 BTU/LB	ETA	0.
ZETA	0.	Y	8.000000E-01
CK	3.404000E-02	RECOVERY FACTOR	9.040550E-01
RADIATION RATE	2.379165E+02 BTU/SEC/FT**2	ABLATION ENERGY	2.086000E+03 BTU/LB
DQDADT AT XHIGH	1.904331E+03 BTU/SEC-FT**2	TOTAL HEATING RATE	1.788626E+04 BTU/SEC
RADIATION CUTOFF	1.380732E+05 FT	TRANSITION POINT	1.010947E-01 FT
XLOW	1.010947E-01 FT	XHIGH	4.202621E+00 FT
ADIABATIC WALL TEMP	8.769383E+03 DEGREES R	ECKERTS TEMP	5.556093E+03 DEGREES R
ECKERTS PRANDTL NO	6.596284E-01	ECKERTS REYNOLDS NO	9.455167E+06 1/FT
COMPRESSIBILITY	1.020854E+00	ECKERTS VISCOSITY	5.422333E-05 LB*(SEC/FT)**3
ECKERTS SPECIFIC WT	3.459346E-02 LB/FT**3	ADIABATIC ENTHALPY	4.189700E+03 BTU/LB
ECKERTS ENTHALPY	1.711880E+03 BTU/LB	WEIGHT LOSS	8.382675E+00 LB
ABLATION TEMP	5.000000E+03 DEGREES R	HEATING RATE AT X	1.904331E+03 BTU/SEC-FT**2

TURBULENT BOUNDARY LAYER

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TIME 0. SECONDS ALTITUDE 5.000000E+03 FEET

ATMOSPHERE PARAMETERS.

	FREE STREAM	LOCAL	UNITS	LOCAL/FREE STREAM
FLOW SPEED	1.500000E+04	1.482045E+04	FT/SEC	9.880301E-01
TEMPERATURE	5.008615E+02	9.449069E+02	DEGREES R	1.886563E+00
PRESSURE	1.760870E+03	1.046908E+04	LB/FT**2	5.945403E+00
MACH NUMBER	1.365058E+01	9.819411E+00		7.193405E-01
SPECIFIC WEIGHT	6.589845E-02	2.076754E-01	LB/FT**3	3.151446E+00
VISCOSITY	1.181629E-05	1.890715E-05	LB*(SEC/FT)**3	1.600092E+00
ENTHALPY	1.209430E+02	1.452741E+03	BTU/LB	1.201178E+01
COMPRESSIBILITY	1.000000E+00	1.000000E+00		1.000000E+00
PRANDTL NUMBER	7.380000E-01	7.388981E-01		1.001217E+00
REYNOLDS NUMBER PER FOOT	8.365372E+07	1.627873E+08	1/FT	1.945986E+00

LAMINAR

ABLAT(1)	2.086000E+03	BTU/LB	ETA	5.170000E-01	
ZETA	3.200000E-01		Y	5.000000E-01	
CK	5.750000E-01		RECOVERY FACTOR	8.595918E-01	
RADIATION RATE	2.379165E+02	BTU/SEC/FT**2	ABLATION ENERGY	5.193844E+03	BTU/LB
QUADT AT XHIGH	1.014363E+03	BTU/SEC-FT**2	TOTAL HEATING RATE	5.275589E+01	BTU/SEC
RADIATION CUTOFF	1.837665E+00	FT	TRANSITION POINT	1.010947E-01	FT
XLOW	0.	FT	XHIGH	1.010947E-01	FT
ADIABATIC WALL TEMP	8.688907E+03	DEGREES R	ECKERTS TEMP	5.479176E+03	DEGREES R
ECKERTS PRANDTL NO	6.616892E-01		ECKERTS REYNOLDS NO	9.692704E+06	1/FT
COMPRESSIBILITY	1.017561E+00		ECKERTS VISCOSITY	5.381530E-05	LB*(SEC/FT)**3
ECKERTS SPECIFIC WT	3.519640E-02	LB/FT**3	ADIABATIC ENTHALPY	3.994044E+03	BTU/LB
ECKERTS ENTHALPY	1.669012E+03	BTU/LB	WEIGHT LOSS	1.015778E-02	LB
ABLATION TEMP	5.000000E+03	DEGREES R	HEATING RATE AT X	3.708185E+02	BTU/SEC-FT**2

TURBULENT

ABLAT(1)	2.086000E+03	BTU/LB	ETA	0.	
ZETA	0.		Y	8.000000E-01	
CK	3.404000E-02		RECOVERY FACTOR	9.040550E-01	
RADIATION RATE	2.379165E+02	BTU/SEC/FT**2	ABLATION ENERGY	2.086000E+03	BTU/LB
QUADT AT XHIGH	2.683415E+03	BTU/SEC-FT**2	TOTAL HEATING RATE	1.147089E+03	BTU/SEC
RADIATION CUTOFF	1.380732E+05	FT	TRANSITION POINT	1.010947E-01	FT
XLOW	1.010947E-01	FT	XHIGH	7.564717E-01	FT
ADIABATIC WALL TEMP	8.769383E+03	DEGREES R	ECKERTS TEMP	5.556093E+03	DEGREES R
ECKERTS PRANDTL NO	6.598284E-01		ECKERTS REYNOLDS NO	9.485167E+06	1/FT
COMPRESSIBILITY	1.020854E+00		ECKERTS VISCOSITY	5.422333E-05	LB*(SEC/FT)**3
ECKERTS SPECIFIC WT	3.459346E-02	LB/FT**3	ADIABATIC ENTHALPY	4.189700E+03	BTU/LB
ECKERTS ENTHALPY	1.711880E+03	BTU/LB	WEIGHT LOSS	5.498986E-01	LB
ABLATION TEMP	5.000000E+03	DEGREES R	HEATING RATE AT X	2.683415E+03	BTU/SEC-FT**2

MAXIMUM X 9.000000E+00 MASS LOSS 1.152532E-01

MAXIMUM X 5.000000E+01 MASS LOSS 7.832776E+00

MASS LOSS = 7.948029E+00 LB/SEC ENDOT = 9.129103E-01

TURB. Q = 1.904331E+03 TURB. SHEAR STRESS = 7.813912E+03

TURBULENT BOUNDARY LAYER

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

INVISCID FLOW FIELD PROFILES

Y/	SS FT2/SEC2	RHO LM/CU.FT	H BTU/LM	V FT/SEC	AMU LM/SEC	PSIRH PSF
14.9189	57255.7	2.7664E-02	2303.76	10754.94	0.	10097.294
14.9206	57232.7	2.7715E-02	2297.52	10769.44	8.4389E-02	10097.155
14.9234	57163.8	2.7852E-02	2278.85	10812.76	2.1763E-01	10096.926
14.9267	57049.2	2.8083E-02	2247.60	10884.42	3.8033E-01	10096.647
14.9304	56899.1	2.8410E-02	2204.44	10983.67	5.6766E-01	10096.325
14.9345	56714.1	2.8834E-02	2155.39	11094.92	7.7805E-01	10095.964
14.9390	56434.6	2.9382E-02	2098.06	11223.54	1.0115E+00	10095.564
14.9438	56141.4	3.0003E-02	2030.66	11372.85	1.2693E+00	10095.122
14.9488	55805.3	3.0709E-02	1953.43	11541.64	1.5532E+00	10094.634
14.9542	55427.2	3.1470E-02	1866.52	11728.63	1.8663E+00	10094.097
14.9598	55008.1	3.2322E-02	1770.20	11932.45	2.2124E+00	10093.503
14.9657	54549.2	3.3244E-02	1664.73	12151.72	2.5966E+00	10092.844
14.9719	54051.7	3.4265E-02	1550.71	12372.25	3.0252E+00	10092.109
14.9784	53496.7	3.5435E-02	1441.21	12603.79	3.5066E+00	10091.283
14.9851	52896.6	4.0378E-02	1316.32	12849.46	4.0514E+00	10090.348
14.9922	52260.5	4.3553E-02	1147.84	13078.25	4.6733E+00	10089.281
14.9996	51590.2	4.7430E-02	1007.68	13287.45	5.3908E+00	10088.050
15.0073	50887.6	5.2304E-02	972.19	13503.29	6.2283E+00	10086.613
15.0155	50154.5	5.7014E-02	884.18	13665.50	7.2194E+00	10084.913
15.0244	49393.0	6.2829E-02	793.93	13829.83	8.4112E+00	10082.868
15.0341	48615.9	7.0959E-02	693.06	14011.23	9.8717E+00	10080.362
15.0446	47860.8	8.0176E-02	606.80	14164.53	1.1702E+01	10077.221
15.0564	46620.4	9.2521E-02	518.19	14320.28	1.4061E+01	10073.173
15.0698	45637.3	1.0784E-01	437.52	14460.61	1.7207E+01	10067.776
15.0859	44634.3	1.2188E-01	384.99	14551.28	2.1591E+01	10060.255
15.1067	43624.1	1.4026E-01	332.05	14642.07	2.8070E+01	10049.138
15.1351	42655.0	1.6396E-01	281.20	14728.76	3.8474E+01	10031.288
15.1784	41675.1	1.9772E-01	229.65	14816.12	5.7424E+01	9998.774
15.3074	41416.4	2.0651E-01	216.65	14838.06	1.2142E+02	9888.964
15.4296	41344.4	2.0906E-01	215.21	14840.50	1.8334E+02	9782.727
15.5729	41310.2	2.1038E-01	214.31	14842.01	2.5640E+02	9657.383
15.7386	41297.2	2.1140E-01	213.69	14843.06	3.4087E+02	9512.444
15.9278	41296.8	1.9136E-01	213.20	14843.88	4.3711E+02	9347.331
16.1415	41304.3	1.9480E-01	212.77	14844.60	5.4546E+02	9161.420
16.3811	41317.1	1.9077E-01	212.35	14845.31	6.6631E+02	8954.068
16.6480	41333.6	1.9028E-01	211.91	14846.06	8.0005E+02	8724.611
16.9437	41352.7	1.9132E-01	211.41	14846.90	9.4706E+02	8472.377
17.2700	41373.6	1.7589E-01	210.84	14847.86	1.1077E+03	8196.683
17.6292	41395.9	1.6999E-01	210.18	14848.98	1.2825E+03	7896.841
18.0238	41419.3	1.6360E-01	209.41	14850.28	1.4717E+03	7572.157

F = 6.0000000E-01 L = 1 NTRIG = 1

SYMBOLS USED ON OUTPUT FROM INVISCID FLOW FIELD PROFILES

Y/R	Normalized Y-coordinate through inviscid field
SS	Entropy, ft^2/sec^2
RHO	Density, lb/ft^3
H	Enthalpy, Btu/lb
V	Velocity, ft/sec
XMD	Mass flow between wall and point in flow field, lb/sec
PSTRM	Pressure, psf

TURBULENT BOUNDARY LAYER
FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TAU	RHOE	WMIX	HW
2.88114E+03	2.01403E-01	2.97536E+01	4.74616E+02

FM	USTOEL	XMASS	UE
7.94803E+00	4.51869E-01	2.67374E+02	1.48431E+04

U	HH	RHOOH	F	Y	YD	T
0.	474.616	1.000000	.824658	4.15457E-07	9.94455E-06	2.77778E+03
.010	511.163	.942623	.816411	5.10651E-07	1.22232E-05	2.77778E+03
.020	546.920	.894188	.808165	6.21313E-07	1.48720E-05	2.77778E+03
.030	581.886	.852849	.799918	7.49240E-07	1.79341E-05	2.77778E+03
.040	616.060	.817237	.791671	8.96391E-07	2.14564E-05	2.77778E+03
.050	649.444	.786320	.783425	1.06490E-06	2.54900E-05	2.77778E+03
.060	682.036	.759300	.775178	1.25711E-06	3.00907E-05	2.77778E+03
.070	713.838	.735553	.766932	1.47555E-06	3.53193E-05	2.77778E+03
.080	744.848	.714584	.758685	1.72299E-06	4.12422E-05	2.77778E+03
.090	775.067	.695996	.750439	2.00244E-06	4.79314E-05	2.77778E+03
.100	804.496	.679466	.742192	2.31720E-06	5.54656E-05	2.77778E+03
.110	833.133	.664728	.733945	2.67084E-06	6.39304E-05	2.77778E+03
.120	860.979	.651564	.725699	3.06725E-06	7.34190E-05	2.77778E+03
.130	888.034	.639789	.717452	3.51067E-06	8.40329E-05	2.77778E+03
.140	914.298	.629250	.709206	4.00570E-06	9.58822E-05	2.77778E+03
.150	939.771	.619817	.700959	4.55737E-06	1.09087E-04	2.77778E+03
.160	964.453	.611379	.692713	5.17113E-06	1.23778E-04	2.77778E+03
.170	988.344	.603841	.684466	5.85292E-06	1.40098E-04	2.77778E+03
.180	1011.444	.597121	.676219	6.60920E-06	1.58201E-04	2.77778E+03
.190	1033.753	.591150	.667973	7.44699E-06	1.78254E-04	2.77778E+03
.200	1055.271	.585866	.659726	8.37394E-06	2.00442E-04	2.77778E+03
.210	1075.997	.581217	.651480	9.39836E-06	2.24963E-04	2.77778E+03
.220	1095.933	.577156	.643233	1.05293E-05	2.52034E-04	2.77778E+03
.230	1115.078	.573644	.634986	1.17766E-05	2.81890E-04	2.77778E+03
.240	1133.431	.570645	.626740	1.31510E-05	3.14787E-04	2.77778E+03
.250	1150.994	.568128	.618493	1.46641E-05	3.51005E-04	2.77778E+03
.260	1167.765	.566067	.610247	1.63286E-05	3.90848E-04	2.77778E+03
.270	1183.746	.564437	.602000	1.81583E-05	4.34645E-04	2.77778E+03
.280	1198.935	.563220	.593754	2.01683E-05	4.82758E-04	2.77778E+03
.290	1213.333	.562396	.585507	2.23750E-05	5.35577E-04	2.77778E+03
.300	1226.941	.561950	.577260	2.47961E-05	5.93530E-04	2.77778E+03
.310	1239.757	.561869	.569014	2.74512E-05	6.57083E-04	2.77778E+03
.320	1251.782	.562142	.560767	3.03614E-05	7.26743E-04	2.77778E+03
.330	1263.016	.562759	.552521	3.35498E-05	8.03063E-04	2.77778E+03
.340	1273.459	.563712	.544274	3.70417E-05	8.86646E-04	2.77778E+03
.350	1283.111	.564994	.536028	4.08645E-05	9.78151E-04	2.77778E+03
.360	1291.972	.566601	.527781	4.50484E-05	1.07830E-03	2.77778E+03
.370	1300.042	.568529	.519534	4.96259E-05	1.18787E-03	2.77778E+03
.380	1307.321	.570776	.511288	5.46331E-05	1.30772E-03	2.77778E+03
.390	1313.809	.573339	.503041	6.01089E-05	1.43879E-03	2.77778E+03
.400	1319.505	.576220	.494795	6.60963E-05	1.58211E-03	2.77778E+03
.410	1324.411	.579419	.486548	7.26421E-05	1.73879E-03	2.77778E+03
.420	1328.526	.582939	.478302	7.97976E-05	1.91007E-03	2.77778E+03
.430	1331.849	.586782	.470055	8.76189E-05	2.09728E-03	2.77778E+03
.440	1334.382	.590953	.461808	9.61675E-05	2.30191E-03	2.77778E+03
.450	1336.123	.595458	.453562	1.05511E-04	2.52556F-03	2.77778E+03
.460	1337.074	.600303	.445315	1.15724E-04	2.77001F-03	2.77778E+03
.470	1337.233	.605497	.437069	1.26887E-04	3.03722F-03	2.77778E+03
.480	1336.602	.611048	.428822	1.39090E-04	3.32931E-03	2.77778E+03
.490	1335.179	.616966	.420575	1.52431E-04	3.64865E-03	2.77778E+03
.500	1332.965	.623264	.412329	1.67018E-04	3.99782E-03	2.77778E+03

TURBULENT BOUNDARY LAYER
FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

.510	1329.960	.629955	.404082	1.82972E-04	4.37969E-03	2.77778E+03
.520	1326.184	.637053	.395836	2.00422E-04	4.79739E-03	2.77778E+03
.530	1321.578	.644574	.387589	2.19516E-04	5.25442E-03	2.77778E+03
.540	1316.200	.652537	.379343	2.40412E-04	5.75460E-03	2.77778E+03
.550	1310.031	.660961	.371096	2.63289E-04	6.30219E-03	2.77778E+03
.560	1303.071	.669869	.362849	2.88342E-04	6.90188E-03	2.77778E+03
.570	1295.319	.679283	.354603	3.15790E-04	7.55888E-03	2.77778E+03
.580	1286.777	.689232	.346356	3.45874E-04	8.27898E-03	2.77778E+03
.590	1277.444	.699744	.338110	3.78862E-04	9.06859E-03	2.77778E+03
.600	1267.320	.710851	.329863	4.15052E-04	9.93487E-03	2.77778E+03
.610	1256.404	.722588	.321617	4.54778E-04	1.08858E-02	2.77778E+03
.620	1244.698	.734995	.313370	4.98410E-04	1.19302E-02	2.77778E+03
.630	1232.201	.748114	.305123	5.46362E-04	1.30779E-02	2.77778E+03
.640	1218.912	.761994	.296877	5.99097E-04	1.43402E-02	2.77778E+03
.650	1204.833	.776686	.288630	6.57135E-04	1.57295E-02	2.77778E+03
.660	1189.962	.792248	.280384	7.21058E-04	1.72596E-02	2.77778E+03
.670	1174.301	.808746	.272137	7.91522E-04	1.89462E-02	2.77778E+03
.680	1157.848	.826250	.263890	8.69265E-04	2.08071E-02	2.77778E+03
.690	1140.604	.844842	.255644	9.55122E-04	2.28622E-02	2.77778E+03
.700	1122.569	.864609	.247397	1.05003E-03	2.51340E-02	2.77778E+03
.710	1103.744	.885654	.239151	1.15507E-03	2.76482E-02	2.77778E+03
.720	1084.127	.908087	.230904	1.27145E-03	3.04339E-02	2.77778E+03
.730	1063.719	.932036	.222658	1.40055E-03	3.35241E-02	2.77778E+03
.740	1042.520	.957644	.214411	1.54395E-03	3.69566E-02	2.77778E+03
.750	1020.530	.985075	.206164	1.70347E-03	4.07749E-02	2.77778E+03
.760	997.749	1.014514	.197918	1.88118E-03	4.50288E-02	2.77778E+03
.770	974.177	1.046174	.189671	2.07949E-03	4.97756E-02	2.77778E+03
.780	949.813	1.080299	.181425	2.30116E-03	5.50816E-02	2.77778E+03
.790	924.659	1.117172	.173178	2.54942E-03	6.10240E-02	2.77778E+03
.800	898.714	1.157120	.164932	2.82801E-03	6.76925E-02	2.77778E+03
.810	871.978	1.200528	.156685	3.14133E-03	7.51922E-02	2.77778E+03
.820	844.450	1.247845	.148438	3.49452E-03	8.36463E-02	2.77778E+03
.830	816.132	1.299605	.140192	3.89368E-03	9.32009E-02	2.77778E+03
.840	787.022	1.356443	.131945	4.34605E-03	1.04029E-01	2.77778E+03
.850	757.122	1.419123	.123699	4.86026E-03	1.16337E-01	2.77778E+03
.860	726.430	1.488572	.115452	5.44669E-03	1.30374E-01	2.77778E+03
.870	694.948	1.565924	.107206	6.11791E-03	1.46441E-01	2.77778E+03
.880	662.674	1.652581	.098959	6.88924E-03	1.64904E-01	2.77778E+03
.890	629.609	1.750302	.090712	7.77953E-03	1.86214E-01	2.77778E+03
.900	595.753	1.861318	.082466	8.81217E-03	2.10932E-01	2.77778E+03
.910	561.107	1.988504	.074219	1.00165E-02	2.39760E-01	2.77778E+03
.920	525.669	2.135633	.065973	1.14299E-02	2.73591E-01	2.77778E+03
.930	489.440	2.307749	.057726	1.31003E-02	3.13575E-01	2.77778E+03
.940	452.420	2.511760	.049479	1.50905E-02	3.61214E-01	2.77778E+03
.950	414.609	2.757374	.041233	1.74843E-02	4.18512E-01	2.77778E+03
.960	376.007	3.058694	.032986	2.03955E-02	4.88196E-01	2.77778E+03
.970	336.614	3.437010	.024740	2.39836E-02	5.74082E-01	2.77778E+03
.980	296.429	3.926044	.016493	2.84787E-02	6.81677E-01	2.77778E+03
.990	255.454	4.582591	.008247	3.42268E-02	8.19268E-01	2.77778E+03
1.000	213.688	5.510306	.000000	4.17773E-02	1.00000E+00	2.77778E+03

F = 8.2465777E-01 L = 32 NTRIG = 2

F =	8.2465777E-01	L = 32	NTRIG = 2	0.	2.7777778E+03	4.4951017E+00	3.5045175E+02
F =	7.4219199E-01	L = 32	NTRIG = 2	3.4848813E-02	2.7777778E+03	4.4951017E+00	6.9274816E+02
F =	6.5972621E-01	L = 32	NTRIG = 2	3.0086724E-02	3.2266399E+03	4.4951017E+00	9.5593951E+02
F =	5.7726044E-01	L = 32	NTRIG = 2	2.8088082E-02	3.4507365E+03	4.4951017E+00	1.1400258E+03
F =	4.9479466E-01	L = 32	NTRIG = 2	2.7361849E-02	3.5422743E+03	4.4951017E+00	1.2450070E+03
F =	4.1232888E-01	L = 32	NTRIG = 2	2.7392080E-02	3.5553322E+03	4.4951017E+00	1.2708831E+03
F =	3.2986311E-01	L = 32	NTRIG = 2	2.8057471E-02	3.5082291E+03	4.4951017E+00	1.2176542E+03

TURBULENT BOUNDARY LAYER
FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

TAU	RHOW	RMUE	WMIX	HW
3.28704E+03	3.55329E-02	2.03681E-01	2.84568E+01	3.36696E+02

EM	EMOUT	DSTDEL	XMASS	UE
7.94803E+00	9.12919E-01	5.53789E-01	2.71161E+04	1.48421E+04

U	RH	RHOOW	F	Y	YD	T
0.	326.696	1.000000	.804767	3.94967E-07	7.67403E-06	2.77778E+03
.100	680.379	.863350	.724291	2.24815E-06	4.36806E-05	3.22514E+03
.200	944.967	.806526	.643814	9.22679E-06	1.79273E-04	3.44407E+03
.300	1130.460	.785648	.563337	3.14831E-05	6.11702E-04	3.53296E+03
.400	1236.859	.786149	.482860	9.55427E-05	1.85635E-03	3.54539E+03
.500	1264.162	.804769	.402384	2.68188E-04	5.21077E-03	3.49837E+03

TURBULENT BOUNDARY LAYER
FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

.600	1212.371	.842888	.321907	7.15437E-04	1.39006E-02	3.39349E+03
.700	1081.486	.908063	.241430	1.85473E-03	3.60366E-02	3.21463E+03
.800	871.505	1.027583	.160953	4.79121E-03	9.30912E-02	2.89663E+03
.900	582.430	1.423561	.080477	1.30208E-02	2.52988E-01	2.06939E+03
1.000	214.259	5.732243	.000000	5.14680E-02	1.00000E+00	4.92003E+02

FS = 8.04767268E-01 HW = 3.30695794E+02 HAW = 4.16899905E+03

HE = 2.14259363E+02 RHOW = 3.55324773E-02 C5 = 3.94966841E-07

C6 = 1.95193398E+01 UE = 1.48420952E+04 B = 4.12209192E+00

CIH = 3.90574835E+00

SYMBOLS USED ON OUTPUT FROM BOUNDARY LAYER CALCULATIONS

TAU	Wall shear stress, $lbm/ft\text{-}sec^2$
RHOW	Boundary layer wall density, lb/ft^3
RHOE	Boundary layer edge density, lb/ft^3
WMIX	Molecular weight of gas mixture at wall
HW	Boundary layer wall enthalpy, Btu/lb
EM	Injected gas mass flow, lb/sec
EMDOT	Local injection rate, $lb/sec\text{-}ft^2$
DSTDEL	Ratio of displacement thickness to boundary layer thickness
XMASS	Boundary layer total gas flow, lb/sec
UE	Boundary layer edge velocity
U	Ratio of local velocity to boundary layer edge velocity
HH	Enthalpy, Btu/lb
RHOOW	Ratio of local density to boundary layer wall density
F	Mass fraction
Y	Distance through boundary layer, ft
YD	Ratio of local distance through boundary layer to boundary layer thickness
T	Temperature, °K
FS	Mass fraction of injected gas at wall
HAW	Adiabatic wall enthalpy, Btu/lb
HE	Boundary layer edge enthalpy, Btu/lb
C5	Grouping of constants from equation (5), $\dfrac{\mu_W}{\left(\tau_W/\rho_W\right)^{1/2}}\left(\dfrac{1}{E}\right)$
C6	Grouping of constants from equation (5), $\dfrac{KU_e}{\left(\tau_W/\rho_W\right)^{1/2}}$

B	Injection parameter
CIR	Vehicle circumference at body station, ft

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

CHEMICAL EQUILIBRIUM COMPOSITION AND PROPERTIES OF A COMPLEX MIXTURE

RHO0 = 1.292332E+00 GM/L NUMBER OF ITERATIONS =17

	CONCENTRATION - MOLE/LITER (FINAL-1)	(FINAL)	CONCENTRATION PARTICLES/CC	MOLE FRACTION	-LOGE(KP)
CN	4.342125E-10	4.342125E-10	2.615427E+11	2.171368E-08	-1.699131E+01
CO	7.912383E-03	7.912383E-03	4.765929E+18	3.956748E-01	-3.034679E+01
NO	4.853726E-07	4.853726E-07	2.923584E+14	2.427205E-05	-1.337835E+01
NO2	8.599463E-13	8.599463E-13	5.179783E+08	4.300337E-11	-9.697263E+00
N2O	2.278364E-11	2.278364E-11	1.372345E+10	1.139342E-09	-1.633687E+01
CO2	8.052170E-05	8.052170E-05	4.850128E+16	4.026652E-03	-3.532161E+01
H2	2.866874E-03	2.866874E-03	1.726827E+18	1.433639E-01	-5.168290E+00
N2	2.995601E-03	2.995601E-03	1.804364E+18	1.498011E-01	-2.546877E+01
O2	8.894060E-09	8.894060E-09	5.357230E+12	4.447656E-07	-6.016134E+00
NH	2.021342E-08	2.021342E-08	1.217531E+13	1.010813E-06	-3.434181E+00
NH3	1.549892E-08	1.549892E-08	9.335589E+12	7.750551E-07	-8.762357E+00
HCN	1.591894E-07	1.591894E-07	9.588585E+13	7.960592E-06	-2.569250E+01
H2O	1.975942E-04	1.975942E-04	1.190185E+17	9.881102E-03	-1.205600E+01
OH	5.429416E-06	5.429416E-06	3.270344E+15	2.715090E-04	-5.664741E+00
HCO	5.409066E-07	5.409066E-07	3.258086E+14	2.704914E-05	-2.355297E+01
H2CO	3.058790E-09	3.058790E-09	1.842425E+12	1.529610E-07	-2.117462E+01
SIH	1.064212E-07	1.064212E-07	6.410151E+13	5.321807E-06	-1.804703E+00
SIO	5.601413E-03	5.601413E-03	3.373944E+18	2.801101E-01	-1.944142E+01
SIO2	3.203134E-05	3.203134E-05	1.929370E+16	1.601793E-03	-2.383984E+01
C	7.443758E-12	7.443758E-12	4.483658E+09	3.722403E-10	0.
N	1.068714E-08	1.068714E-08	6.437271E+12	5.344323E-07	0.
O	3.085001E-07	3.085001E-07	1.858213E+14	1.542718E-05	0.
A	3.577268E-05	3.577268E-05	2.154724E+16	1.788886E-03	0.
H	2.676187E-04	2.676187E-04	1.611969E+17	1.338282E-02	0.
SI	2.870273E-07	2.870273E-07	1.728875E+14	1.435339E-05	0.

FINAL CONDITIONS

TEMP DEG-K	RHO/RHO0	P/P0	Z	SUM CON MOL/L
2.777778E+03	4.404251E-01	4.558108E+00	8.020455E-01	1.999718E-02

HMO/RT0	SMO/R
7.866577E+00	2.533951E+01

COLD MIXTURE MOLE WT. = 22.828453

COLLISION FREQUENCY = 5.028181E+11

ENTHALPY = 4.456542E-01JOULES / CC HCALC = 3.366958E+02

DENSITY = 1.000000E+00

TOTAL CONCENTRATION OF	N2	AT STANDARD CONDITIONS	1.20161E+01 PER CENT
TOTAL CONCENTRATION OF	O2	AT STANDARD CONDITIONS	3.22830E+00 PER CENT
TOTAL CONCENTRATION OF	A	AT STANDARD CONDITIONS	1.43477E-01 PER CENT
TOTAL CONCENTRATION OF	C	AT STANDARD CONDITIONS	3.20607E+01 PER CENT
TOTAL CONCENTRATION OF	H	AT STANDARD CONDITIONS	2.56806E+01 PER CENT
TOTAL CONCENTRATION OF	O	AT STANDARD CONDITIONS	4.27469E+00 PER CENT
TOTAL CONCENTRATION OF	SIO2	AT STANDARD CONDITIONS	2.25962E+01 PER CENT

TURBULENT BOUNDARY LAYER

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

BOUNDARY LAYER PROFILES

U	RHOOW	F	T	CF	EL	Y	Y/YO	H
0.	1.000248	.804757	2.77778E+03	5.02818E+11	1.49885E+13	3.94967E-07	8.09590E-06	3.36696E+02
.020	.962439	.788672	2.89057E+03	4.92911E+11	2.24974E+13	5.77134E-07	1.18299E-05	4.11760E+02
.040	.931125	.772577	2.99015E+03	4.84634E+11	3.13323E+13	8.20041E-07	1.69319E-05	4.83660E+02
.060	.904545	.756481	3.07922E+03	4.77573E+11	4.09818E+13	1.16140E-06	2.38059E-05	5.52397E+02
.080	.882213	.740386	3.15741E+03	4.71623E+11	5.10395E+13	1.60775E-06	3.29551E-05	6.17970E+02
.100	.863464	.724291	3.22542E+03	4.66623E+11	6.10541E+13	2.19559E-06	4.50045E-05	6.80379E+02
.120	.847729	.708195	3.28414E+03	4.62433E+11	7.06177E+13	2.96265E-06	6.07274E-05	7.39624E+02
.140	.834538	.692100	3.33446E+03	4.58930E+11	7.94188E+13	3.95538E-06	8.10760E-05	7.95705E+02
.160	.823498	.676005	3.37734E+03	4.56008E+11	8.72562E+13	5.23081E-06	1.07219E-04	8.48623E+02
.180	.814284	.659909	3.41367E+03	4.53575E+11	9.40228E+13	6.85872E-06	1.40588E-04	8.98377E+02
.200	.806633	.643814	3.44427E+03	4.51556E+11	9.96859E+13	8.92419E-06	1.82925E-04	9.44967E+02
.220	.800334	.627718	3.46987E+03	4.49887E+11	1.04259E+14	1.15307E-05	2.36352E-04	9.88393E+02
.240	.795219	.611623	3.49106E+03	4.48519E+11	1.07786E+14	1.48037E-05	3.03442E-04	1.02866E+03
.260	.791139	.595528	3.50844E+03	4.47407E+11	1.10342E+14	1.88953E-05	3.87310E-04	1.06575E+03
.280	.787998	.579432	3.52238E+03	4.46521E+11	1.11986E+14	2.39889E-05	4.91716E-04	1.09969E+03
.300	.785749	.563337	3.53309E+03	4.45844E+11	1.12754E+14	3.03058E-05	6.21198E-04	1.13046E+03
.320	.784329	.547242	3.54082E+03	4.45357E+11	1.12710E+14	3.81128E-05	7.81223E-04	1.15807E+03
.340	.783697	.531146	3.54578E+03	4.45045E+11	1.11913E+14	4.77305E-05	9.78364E-04	1.18251E+03
.360	.783818	.515051	3.54813E+03	4.44898E+11	1.10424E+14	5.95446E-05	1.22052E-03	1.20379E+03
.380	.784671	.498956	3.54799E+03	4.44906E+11	1.08298E+14	7.40177E-05	1.51719E-03	1.22191E+03
.400	.786242	.482860	3.54545E+03	4.45066E+11	1.05588E+14	9.17053E-05	1.87974E-03	1.23686E+03
.420	.788524	.466765	3.54058E+03	4.45372E+11	1.02348E+14	1.13273E-04	2.32184E-03	1.24865E+03
.440	.791518	.450670	3.53342E+03	4.45823E+11	9.86270E+13	1.39520E-04	2.85984E-03	1.25727E+03
.460	.795229	.434574	3.52400E+03	4.46418E+11	9.44762E+13	1.71403E-04	3.51337E-03	1.26273E+03
.480	.799670	.418479	3.51232E+03	4.47160E+11	8.99444E+13	2.10070E-04	4.30594E-03	1.26503E+03

TURBULENT BOUNDARY LAYER

FULL BOUNDARY LAYER CHECK CASE WITH CHEMISTRY

BOUNDARY LAYER PROFILES

U	RHOD#	F	T	CF	EL	Y	Y/YD	H
.500	.804857	.402384	3.49838E+03	4.48050E+11	8.50822E+13	2.56895E-04	5.26575E-03	1.20416E+03
.520	.810781	.386208	3.48228E+03	4.49084E+11	7.99600E+13	3.13528E-04	6.42659E-03	1.20013E+03
.540	.817507	.370143	3.46383E+03	4.50279E+11	7.46022E+13	3.81947E-04	7.82902E-03	1.25294E+03
.560	.825077	.354098	3.44296E+03	4.51642E+11	6.90570E+13	4.64528E-04	9.52173E-03	1.24258E+03
.580	.833543	.338002	3.41955E+03	4.53185E+11	6.33738E+13	5.64126E-04	1.15633E-02	1.22906E+03
.600	.842971	.321907	3.39347E+03	4.54923E+11	5.76025E+13	6.84177E-04	1.40240E-02	1.21237E+03
.620	.853437	.305812	3.36454E+03	4.56875E+11	5.17953E+13	8.28824E-04	1.69890E-02	1.19252E+03
.640	.865041	.289716	3.33253E+03	4.59063E+11	4.60048E+13	1.00307E-03	2.05605E-02	1.16951E+03
.660	.877912	.273621	3.29715E+03	4.61520E+11	4.02845E+13	1.21295E-03	2.48627E-02	1.14333E+03
.680	.892207	.257526	3.25800E+03	4.64283E+11	3.46912E+13	1.46582E-03	3.00460E-02	1.11399E+03
.700	.908140	.241430	3.21457E+03	4.67410E+11	2.92816E+13	1.77062E-03	3.62935E-02	1.08149E+03
.720	.925998	.225335	3.16616E+03	4.70970E+11	2.41151E+13	2.13826E-03	4.38293E-02	1.04582E+03
.740	.946181	.209239	3.11180E+03	4.75066E+11	1.92532E+13	2.58216E-03	5.29283E-02	1.00698E+03
.760	.969260	.193144	3.05011E+03	4.79846E+11	1.47622E+13	3.11896E-03	6.39314E-02	9.64989E+02
.780	.996024	.177049	2.97925E+03	4.85519E+11	1.07003E+13	3.76938E-03	7.72635E-02	9.19829E+02
.800	1.027664	.160953	2.89653E+03	4.92403E+11	7.14749E+12	4.55964E-03	9.34620E-02	8.71505E+02
.820	1.066680	.144858	2.79861E+03	5.01122E+11	4.24054E+12	5.52354E-03	1.13220E-01	8.20017E+02
.840	1.117117	.128763	2.67203E+03	5.12671E+11	2.08239E+12	6.70601E-03	1.37458E-01	7.65366E+02
.860	1.186147	.112667	2.51235E+03	5.28713E+11	7.47584E+11	8.16929E-03	1.67451E-01	7.07551E+02
.880	1.283101	.096572	2.31195E+03	5.51151E+11	1.63470E+11	1.00030E-02	2.05038E-01	6.46572E+02
.900	1.423697	.080477	2.06923E+03	5.82580E+11	1.74673E+10	1.23417E-02	2.52976E-01	5.82430E+02
.920	1.623809	.064381	1.79959E+03	6.24702E+11	6.89379E+08	1.53931E-02	3.15524E-01	5.15123E+02
.940	1.923370	.048286	1.50664E+03	6.82739E+11	5.98649E+06	1.94949E-02	3.99600E-01	4.44653E+02
.960	2.401511	.032191	1.19662E+03	7.66093E+11	4.40668E+03	2.52448E-02	5.17459E-01	3.71019E+02
.980	3.311962	.016095	8.60503E+02	9.03408E+11	2.20824E-02	3.38763E-02	6.94385E-01	2.94221E+02
1.000	5.732243	.000000	4.92003E+02	0.	0.	4.87860E-02	1.00000E+00	2.14259E+02

XMASS = 3.0705862E+02 EMB = 1.2588930E+01

SYMBOLS USED ON OUTPUT FROM BOUNDARY LAYER PROFILE TABULATION

U	Ratio of local velocity to edge velocity
RHOOW	Ratio of local density to wall density
F	Mass fraction
T	Temperature, °K
CF	Collision frequency, encounters/sec
EL	Electron concentration, electrons/cm^3
Y	Distance through boundary layer, ft
Y/YD	Ratio of local distance through boundary layer to boundary layer thickness
H	Enthalpy, Btu/lb
XMASS	Boundary layer total gas flow, lb/sec
EMB	Boundary layer injected gas flow, lb/sec

APPENDIX

MIXING PROGRAM*

The following sections describe the program which has been developed to perform boundary layer mixing calculations. Section I presents the differential equations which describe the situation and defines the boundary layer parameters needed for the solution to the problem. In Section II the differential equations from Section I are converted to difference equation form as they are used in the program. Section III consists of a listing of the program and a sample run to illustrate the form and content of the input and output.

*The program described in this Appendix has not been checked out by AFWL. It is included here to document work performed under the contract, but AFWL assumes no responsibility for the accuracy or completeness of this Appendix.

SECTION I

MIXING EQUATIONS--DIFFERENTIAL FORM

Diffusion equation in point form is

$$0 = \rho V_x \frac{\partial f}{\partial x} + \rho V_y \frac{\partial f}{\partial y} - \frac{\partial}{\partial y}\left[\rho D \frac{\partial f}{\partial y}\right] \qquad \text{(A-1)}$$

where

$f = f(x,y)$ = mass fraction of ablator B

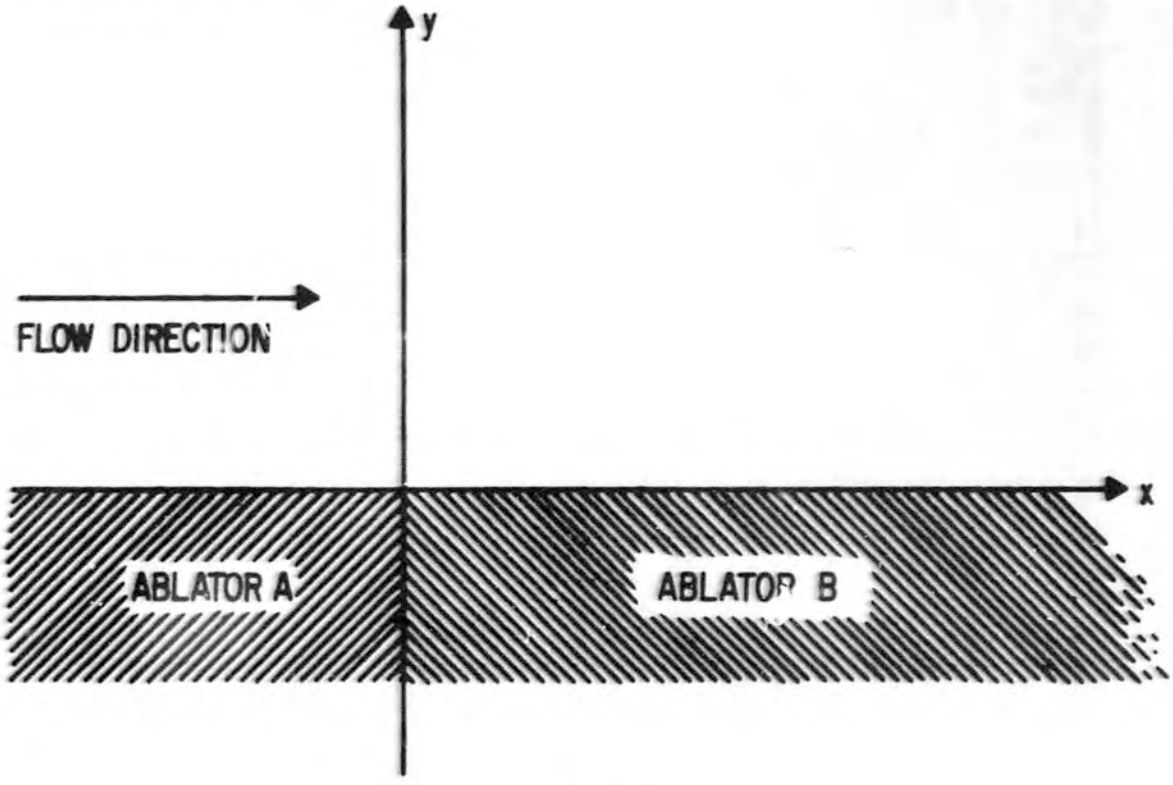
$\rho = \rho(y)$ = density (gm/cm^3)

$V_x = V_x(y)$ = velocity in x direction (cm/sec)

$V_y = V_y(y)$ = velocity in y direction (cm/sec)

$D = D(y)$ = diffusion coefficient (cm^2/sec)

Coordinate System



Equation (A-1) can be expanded as follows:

$$0 = \rho V_x \frac{\partial f}{\partial x} + \rho V_y \frac{\partial f}{\partial y} - \frac{\partial f}{\partial y} \cdot \frac{\partial}{\partial y} (\rho D) - \rho D \frac{\partial^2 f}{\partial y^2}$$

$$0 = \rho V_x \frac{\partial f}{\partial x} - \left[\frac{\partial}{\partial y} (\rho D) - \rho V_y\right] \frac{\partial f}{\partial y} - \rho D \frac{\partial^2 f}{\partial y^2} \qquad \text{(A-2)}$$

SECTION II

CONVERT EQUATION A-2 TO FINITE DIFFERENCE FORM

Using a forward difference expansion for

$$\frac{\partial(\rho D)}{\partial y}$$

equation (2) at f(I,J) becomes

J = Y coordinate = Y_J

I = X coordinate = X_I

$$0 = \rho_c(J)\, V_{xc}(J) \left[\frac{f(I,J) - f(I-1,J)}{\Delta x}\right]$$

$$- \left[\frac{\rho(J+1)D(J+1) - \rho(J)D(J)}{Y(J+1) - Y(J)} - \rho_c(J)\, V_{yc}(J)\right] \left.\frac{\partial f}{\partial y}\right|_{Y_c(J)}$$

$$- \left[\rho_c(J) D_c(J)\right] \left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(J)}$$

Note that a backward difference is used to determine $\partial f/\partial x$. An expression is now needed for

$$\left.\frac{\partial f}{\partial y}\right|_{Y_c(J)}$$

and

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(J)}$$

These expressions are developed next using the three-point Lagrangian interpolation formula:

$$f(X) = \sum_{i=0}^{n} \ell_i(X)\, f_i + R_n(X) \qquad i = [0,n] \tag{A-3}$$

where

$$\ell_i(X) = \frac{\pi_n(X)}{(X-X_i)\pi_n(X_i)} = \frac{(X-X_o)(X-X_1)\cdots\cancel{(X-X_i)}\cdots(X-X_n)}{\cancel{(X-X_i)}(X_i-X_o)\cdots(X_i-X_{i-1})(X_i-X_{i+1})\cdots(X_i-X_n)}$$

$R_n(X)$ = remainder

For the three-point formula - n = 2

$$f(X) = \frac{(X-X_1)(X-X_2)}{(X_o-X_1)(X_o-X_2)} f_o + \frac{(X-X_o)(X-X_2)}{(X_1-X_o)(X_1-X_2)} f_1 + \frac{(X-X_o)(X-X_1)}{(X_2-X_o)(X_2-X_1)} f_2 + R_2(X)$$

Now differentiate to determine $\partial f/\partial x$.

$$\frac{\partial f}{\partial x} = \left[\frac{(X-X_2)}{(X_o-X_1)(X_o-X_2)} + \frac{(X-X_1)}{(X_o-X_1)(X_o-X_2)}\right] f_o$$

$$+ \left[\frac{(X-X_2)}{(X_1-X_o)(X_1-X_2)} + \frac{(X-X_o)}{(X_1-X_o)(X_1-X_2)}\right] f_1$$

$$+ \left[\frac{(X-X_1)}{(X_2-X_o)(X_2-X_1)} + \frac{(X-X_o)}{(X_2-X_o)(X_2-X_1)}\right] f_2 + R_2'(X) \tag{A-4}$$

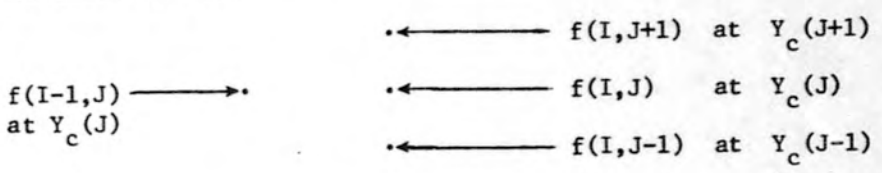
Differentiate $\partial f/\partial x$ to get $\partial^2 f/\partial y^2$.

$$\frac{\partial^2 f}{\partial x^2} = \frac{2}{(X_o-X_1)(X_o-X_2)} f_o + \frac{2}{(X_1-X_o)(X_1-X_2)} f_1$$

$$+ \frac{2}{(X_2-X_o)(X_2-X_1)} f_2 + R_2''(X) \qquad \text{(A-5)}$$

Note that $R_2''(X) = 0$. Now, using the notation which is used in the program, the complete finite difference form of the diffusion equation is derived.

Typical Location



The three points used in the Lagrangian interpolation formula are

$$Y_c(J-1),\ Y_c(J),\ Y_c(J+1)$$

Then $\partial f/\partial y$ becomes the following:

$$\left.\frac{\partial f}{\partial y}\right|_{Y_c(J)} = \left[\frac{Y_c(J)-Y_c(J+1)}{\left[Y_c(J-1)-Y_c(J)\right]\left[Y_c(J-1)-Y_c(J+1)\right]}\right] f(I,J-1)$$

$$+ \left[\frac{Y_c(J)-Y_c(J+1)}{\left[Y_c(J)-Y_c(J-1)\right]\left[Y_c(J)-Y_c(J+1)\right]}\right.$$

$$\left. + \frac{Y_c(J)-Y_c(J-1)}{\left[Y_c(J)-Y_c(J-1)\right]\left[Y_c(J)-Y_c(J+1)\right]}\right] f(I,J)$$

$$+ \left[\frac{Y_c(J)-Y_c(J-1)}{\left[Y_c(J+1)-Y_c(J-1)\right]\left[Y_c(J+1)-Y_c(J)\right]}\right] f(I,J+1) \qquad \text{(A-6)}$$

and

$$\frac{\partial^2 f}{\partial y^2} = \frac{2}{\left[Y_c(J-1)-Y_c(J)\right]\left[Y_c(J-1)-Y_c(J+1)\right]} f(I,J-1)$$

$$+ \frac{2}{\left[Y_c(J)-Y_c(J-1)\right]\left[Y_c(J)-Y_c(J+1)\right]} f(I,J)$$

$$+ \frac{2}{\left[Y_c(J+1)-Y_c(J-1)\right]\left[Y_c(J+1)-Y_c(J)\right]} f(I,J+1) \qquad \text{(A-7)}$$

To simplify the notation in the program, the following terms will be defined at point (I,J):

$$C1 = \rho(J+1)D(J+1) - \rho(J)D(J)$$

$$C2 = -C14 = Y_c(J+1) - Y_c(J-1)$$

$$C3 = C13 = Y_c(J) - Y_c(J-1)$$

$$C4 = C12 = Y_c(J+1) - Y_c(J)$$

$$C5 = Y(J+1) - Y(J)$$

$$C6 = \rho_c(J) \cdot D_c(J)$$

$$C7 = \rho_c(J) \cdot V_{xc}(J)$$

$$C8 = \rho_c(J) \cdot V_{yc}(J)$$

Therefore equation (A-2) becomes (at point I,J)

$$0 = C7\left[\frac{f(I,J)-f(I,J-1)}{\Delta x}\right] - \left[\frac{C1}{C5} - C8\right]\left.\frac{\partial f}{\partial y}\right|_{Y_c(J)} - C6\left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(J)} \qquad \text{(A-2a)}$$

Equation (A-6) becomes

$$\left.\frac{\partial f}{\partial y}\right|_{Y_c(J)} = \frac{C12}{C13 \cdot C14} f(I,J-1) + \left[\frac{1}{C12} - \frac{1}{C13}\right] f(I,J) - \left[\frac{C13}{C12 \cdot C14}\right] f(I,J+1)$$

(A-7a)

and equation (A-7) becomes

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(J)} = \frac{2}{C13 \cdot C14} f(I,J-1) - \frac{2}{C13 \cdot C12} f(I,J)$$

$$+ \frac{2}{C12 \cdot C14} f(I,J+1) \qquad \text{(A-7a)}$$

To simplify the notation still further, define

$$B1 = C12/(C13 \cdot C14)$$

$$B2 = 1/C12 - 1/C13$$

$$B3 = -C13/(C12 \cdot C14)$$

$$B4 = 1/(C13 \cdot C14)$$

$$B5 = -1/(C12 \cdot C13)$$

$$B6 = 1/(C12 \cdot C14)$$

Using this notation the derivatives become

$$\left.\frac{\partial f}{\partial y}\right|_{Y_c(J)} = B1 \cdot f(I,J-1) + B2 \cdot f(I,J) + B3 \cdot f(I,J+1) \qquad \text{(A-6b)}$$

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(J)} = 2B4 \cdot f(I,J-1) + 2B5 \cdot f(I,J) + 2B6 \cdot f(I,J+1) \qquad \text{(A-7b)}$$

Now substitute the derivatives into (A-2a)

$$0 = C7 \left[\frac{f(I,J) - f(I-1,J)}{\Delta x}\right]$$

$$- \left[\frac{C1}{C5} - C8\right]\left[B1\ f(I,J-1) + B2\ f(I,J) + B3\ f(I,J+1)\right]$$

$$- C6\ \left[2B4\ f(I,J-1) + 2B5\ f(I,J) + 2B6\ f(I,J+1)\right]$$

The unknowns are f(I,J-1), f(I,J), and f(I,J+1), therefore the equation is written as a function of the unknowns as follows:

$$\left\{-B1\left[\frac{C1}{C5} - C8\right] - 2B4 \cdot C6\right\} f(1,J-1)$$

$$+ \left\{\frac{C7}{\Delta x} - B2\left[\frac{C1}{C5} - C8\right] - 2B5 \cdot C6\right\} f(I,J)$$

$$+ \left\{-B3\left[\frac{C1}{C5} - C8\right] - 2B6 \cdot C6\right\} f(I,J+1)$$

$$= \frac{C7 \cdot f(I-1,J)}{\Delta x} \qquad \text{(A-2b)}$$

This is the diffusion equation in discrete form as it appears in the program.

In a typical case utilizing 50 unknowns (50 pt. values of mass fraction) the diffusion equation would be written in this form for 48 of the unknown points. The remaining two equations would also be the diffusion equation at the edge of the boundary layer and at the wall. Each of these points results in a slightly different form of the diffusion equation because of the boundary conditions which are assumed. The boundary conditions are as follows:

1. BOUNDARY CONDITION AT THE WALL

The following two equations are solved to give f_W = the mass fraction at the wall

$$\frac{\partial f}{\partial y} = \frac{-\dot{m}\left(f_W\right)}{\rho D}$$

$$\frac{\partial f}{\partial y} = \frac{-f_W}{V_e}\frac{\partial f}{\partial y}$$

or

$$f_W = \frac{1}{1 + \frac{\tau}{\dot{m}V_e}} \qquad \text{(A-8)}$$

where

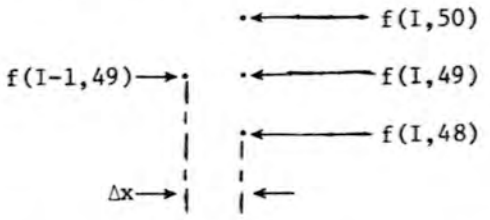
τ = shear stress at the wall (gm/cm^2-sec)

$\dot{m}$ = ablation rate (gm/cm^2-sec)

V_e = velocity of flow at the edge (cm/sec)

At the wall, therefore, the diffusion equation is written exactly as described for the interior points except that the mass fraction at one of the points in the three-point Lagrangian expansion is now known and hence the diffusion equation involves only two unknowns.

2. BOUNDARY CONDITION AT THE EDGE OF THE "BOUNDARY LAYER"



The equation written for the point at $Y_c(50)$ is the original diffusion equation, but the expansion which is is used for $\partial f/\partial y$ and $\partial^2 f/\partial y^2$ is the Lagrangian three-point formula (points $Y_c(48)$, $Y_c(49)$, $Y_c(50)$) evaluated at $Y_c(50)$. The derivation of the finite difference form is as follows:

$$0 = \rho V_x \frac{\partial f}{\partial y} - \left[\frac{\partial}{\partial y}(\rho D) - \rho V_y\right] \frac{\partial f}{\partial y} - \rho D \frac{\partial^2 f}{\partial y^2}$$

or, in finite difference form,

$$0 = \rho_c(50)\, V_{xc}(50) \left[\frac{f(I,50) - f(I-1,50)}{\Delta x}\right]$$

$$- \left[\frac{\rho(51)D(51) - \rho(50)D(50)}{Y(51) - Y(50)} - \rho_c(50)\, V_{yc}(50)\right] \left.\frac{\partial f}{\partial y}\right|_{Y_c(50)}$$

$$- \left[\rho_c(50) D_c(50)\right] \left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(50)}$$

With J = 50 and using the notation of pages 434 and 435, this becomes

$$0 = \frac{C7}{\Delta x}\left[f(I,19) - f(I-1,19)\right] - \left[\frac{C1}{C5} - C8\right] \left.\frac{\partial f}{\partial y}\right|_{Y_c(50)}$$

$$- C6 \left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(50)}$$

Let

$$\left.\frac{\partial f}{\partial y}\right|_{Y_c(50)} = \frac{f(I,50) - f(I,49)}{Y_c(50) - Y_c(49)}$$

and

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{Y_c(50)} = 0$$

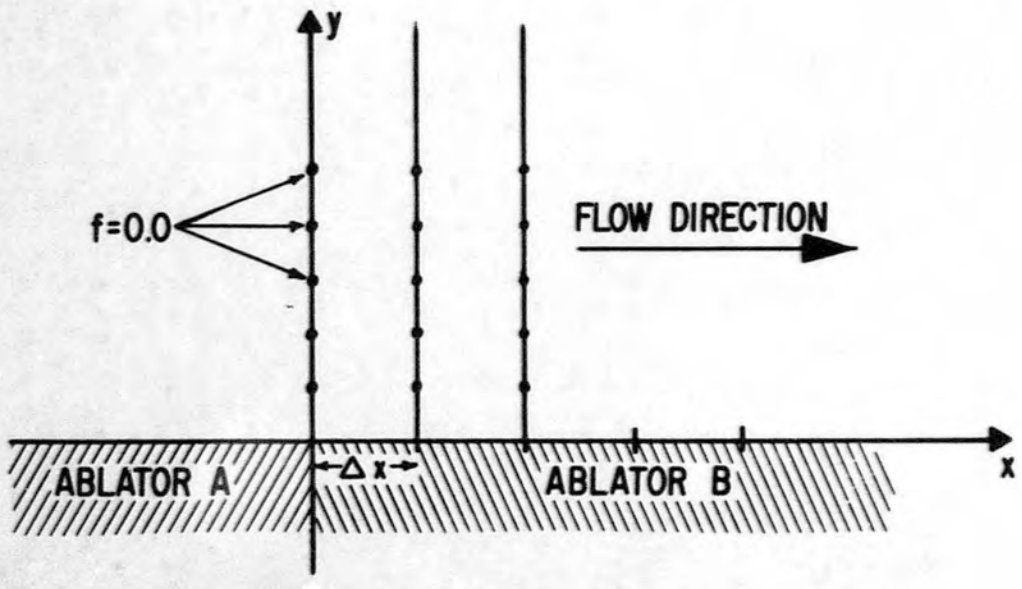
Thereafter substituting the derivatives into the diffusion equation it becomes

$$\left[\frac{C1}{C5} - C8\right]\frac{1}{C3} f(I,49) + \left[\frac{C7}{\Delta x} - \left\{\frac{C1}{C5} - C8\right\}\frac{1}{C3}\right] f(I,50)$$

$$= \frac{C7}{\Delta x} f(I-1,50) \qquad \text{(A-9)}$$

SUMMARY

The operations described above result in N linear equations in N unknowns and are solved for the values of the mass fraction of ablator B at each of the N points in the boundary layer. This set of equations is first written at the point X = 0 and Y = 0 in the illustration below.



with the value of mass fraction at X = 0 assumed to be zero for all y. This set of equations is then solved and the resulting mass fractions at X = ΔX are used as boundary conditions in solving for the mass fractions at X = $2\Delta X$. This procedure is then repeated along the ablator to any specified station.

INPUT

The first input card to be read contains the following information:

MYMAX	= 51	=	The number of cells in the boundary layer (Col. 9-10)
MDOT	= 102	=	Number of stations along the ablator at which the mass fractions are calculated with mass addition (in $\neq$ 0) (Col. 18-20)
NTIMES	= 1	=	Number of times the above on-off process is repeated (Col. 30)
XMASS(L)	= 194820	=	Ablation rate in gm/cm^2-sec (Col. 50-55)
DELX	= 0.10	=	Step size along the ablator in cm (Col. 67-70)

The remainder of the inputs to the program consists of three sets of cards giving the following information:

1. A set of cards giving the distances (in cm) out from the ablator wall at which the diffusion equation is written. There are MYMAX of these values and they are read into the Y(.) array on a 4E15.8 format.

2. A set of cards giving the density profile at the Y(.) stations read in above. The density is in gm/cm^3 and is read into the RHO(.) array on a 4E15.8 format.

3. A set of cards giving the Velocity/Velocity (EDGE) ratio at the stations above. Format is 14F5.2.

```
      PROGRAM SUYO(INPUT,TAPE5=INPUT,OUTPUT,TAPE6=OUTPUT)
      DIMENSION AX(51,51),VY(51),VYC(51),Y(51),RHO(51),D(51),VX(51)
      DIMENSION BX(51,1),YC(51),RHOC(51),VXC(51),XMASS(200),DC(51)
      DIMENSION A(51,51),F(105,51),FX(51,51)
      LOGICAL IPRNT
      IPRNT=.FALSE.
C     READ IN THE FOLLOWING CONTROL VARIABLES
C       MYMAX= NUMBER OF CELLS IN BOUNDARY LAYER PLUS ONE
C       MYDOT= NUMBER OF CELLS OVER WHICH ABLATION OCCURS
C       MDOTNO=NUMBER OF CELLS OVER WHICH ABLATION DOES NOT OCCUR
C       NTIMES=THE NUMBER OF --ON-OFF- SECTIONS ALONG THE ABLATION
C       XMASS(1)= ABLATION RATE IN GM/CM**2-SEC
C       DELX= LENGTH OF EACH CELL ALONG THE ABLATING SURFACE  (CM.)
      READ(5,1000)MYMAX,MDOT,MDOTNO,NTIMES,XMASS(1),DELX
 1000 FORMAT(4I10,2E15.8)
      WRITE(6,1014)MYMAX
 1014 FORMAT(1H135H THE BOUNDARY LAYER IS DIVIDED INTO,I4,2X5HCELLS)
      WRITE(6,1015)MDOT,MDOTNO,NTIMES
 1015 FORMAT(1H010H THERE ARE,I4,2X,12HCELLS ON AND,I4,2X,
     11HCELLS OFF REPEATED,I4,2X,5HTIMES)
      XMASS(1)= XMASS(1)/((12.0*2.54)**2)/2.2046E-03
      WRITE(6,1016)XMASS(1),DELX
 1016 FORMAT(1H0,21H THE ABLATION RATE IS,E15.4,2X,3HAND THE CELL WIDTH
     1 ALONG THE ABLATOR IS,E15.4,2X,3HCM.)
      IT=1
 1002 FORMAT(4E15.8)
      READ(5,1002)(Y(I),I=1,MYMAX)
      READ(5,1002)(RHO(I),I=1,MYMAX)
      READ(5,1003)(VX(I),I=1,MYMAX)
 1003 FORMAT(14F5.2)
C     THE NEXT LOOP WILL CONVERT THE INPUT INTO CONSISTENT UNITS
      DO 10 I=1,MYMAX
      Y(I)=(Y(I)-1.10E-06)*12.0*2.54
      RHO(I)=RHO(I)*39.0255E-02/((12.0*2.54)**3)/2.2046E-03
      VX(I)=VX(I)*15.4118E+03*12.0*2.54
   10 CONTINUE
C     GENERATE THE VELOCITIES PERPENDICULAR TO THE FLOW  (VY)
      DO 20 K=1,MYMAX
      VY(K)= XMASS(1)/RHO(K)
   20 CONTINUE
C     CALCULATE THE DIFFUSION COEFFICIENTS
      WALL=.483891
      INDEX= MYMAX-1
      TAU=2338.56/12.0/2.54/2.2046E-03
      DO 30 M=2,INDEX
      C2= Y(M+1)-Y(M-1)
      C3= Y(M)-Y(M-1)
      C9= Y(M)+Y(M-1)
      C10= Y(M+1)+Y(M-1)
      D1=((C2/C3-1.0)*(2.0*Y(M)-C9)/(C2*(C10-C9))-1.0/C3)*VX(M-1)
      D2=(1.0/C3-(2.0*Y(M)-C9)/(C3*(C10-C9)))*VX(M)
      D3=((2.0*Y(M)-C9)/(C2*(C10-C9)))*VX(M+1)
      DERIV=D1+D2+D3
      D(M)=(TAU*XMASS(1)*VX(M))/RHO(M)/DERIV
   30 CONTINUE
      D(1)=D(2)-(D(3)-D(2))*(Y(2)-Y(1))/(Y(3)-Y(2))
      D(MYMAX)=D(50)+(D(50)-D(49))*(Y(51)-Y(50))/(Y(50)-Y(49))
      WRITE(6,1004)
```

```
 1004 FORMAT(52H1ONE DIMENSIONAL DIFFUSION WITH TWO DIMENSIONAL FLOW)
      WRITE(6,1005)
 1005 FORMAT(21HUEDGE CELL PARAMETERS)
      WRITE(6,1006)
 1006 FORMAT(6HO CELL16X4HY CM5X15HDENSITY G/CM**320H  DIFFUSION CONSTAN
     1T20H X FLOW SPEED CM/SEC20H Y FLOW SPEED CM/SEC)
      WRITE(6,1007) (I,Y(I),RHO(I),D(I),VX(I),VY(I),I=1,MYMAX)
 1007 FORMAT(I6,5E20.8)
C     COMPUTE THE CENTER CELL PARAMETERS
      WRITE(6,1004)
      WRITE(6,1008)
 1008 FORMAT(23HOCENTER CELL PARAMETERS)
      WRITE(6,1006)
      DO 40 I=1,INDEX
      YC(I)=0.5*(Y(I+1)+Y(I))
      RHOC(I)=0.5*(RHO(I+1)+RHO(I))
      DC(I)=0.5*(D(I+1)+D(I))
      VXC(I)=0.5*(VX(I+1)+VX(I))
      VYC(I)=0.5*(VY(I+1)+VY(I))
      WRITE(6,1007) I,YC(I),RHOC(I),DC(I),VXC(I),VYC(I)
 40   CONTINUE
C     CALCULATE THE ABLATION RATES ALONG THE VECHICLE
      DO 50 NIX=1,NTIMES
      MAX=MDOT+1
      DO 51 I=2,MAX
      IA= I + (NIX-1)*(MDOT+MDOTNO)
 51   XMASS(IA)=XMASS(1)
      MIN=MDOT+2
      MAX=MDOT+MDOTNO+1
      DO 52 I=MIN,MAX
      IA= I + (NIX-1)*(MDOT+MDOTNO)
 52   XMASS(IA)=0.0
 50   CONTINUE
C     BEGIN CALCULATION OF THE COEFFICIENT MATRIX (A)
      DO 60 NTIM=1,NTIMES
      MAX=MDOT+MDOTNO+1
 240  CONTINUE
      IF(NTIM.GT.1)GO TO 70
      DO 80 M1=1,MYMAX
      DO 80 M2=1,INDEX
 80   A(M2,M1)=0.0
      DO 200 K=1,INDEX
      J=K+1
      C1=RHO(J+1)*D(J+1)-RHO(J)*D(J)
      C2=YC(J+1)-YC(J-1)
      C3=YC(J)-YC(J-1)
      C4=YC(J+1)-YC(J)
      C5=Y(J+1)-Y(J)
      C6=RHOC(J)*DC(J)
      C7=RHOC(J)*VXC(J)
      C8=RHOC(J)*VYC(J)
      C9=YC(J)*YC(J-1)
      C10=YC(J+1)*YC(J-1)
      C11=0.5*(YC(J+1)-YC(J-1))
      IF(K-MYMAX+2)110,120,130
 110  CONTINUE
C     LAGRANGIAN FORMULATION
      C12=YC(J)-YC(J+1)
      C13=YC(J-1)-YC(J)
      C14=YC(J-1)-YC(J+1)
      B1=C12/(C13*C14)
      B2= 1.0/C12-1.0/C13
```

```
      B4=-C13/(C12*C14)
      B4=1.0/(C13*C14)
      B5= -1.0/(C12*C13)
      B6=1.0/(C12*C14)
      A(K,J-1)=B1*(C1/C5-C8)-2.0*C6*B4
      A(K,J)=C7/DELX-B2*(C1/C5-C8)-2.0*B5*C6
      A(K,J+1)=-B3*(C1/C5-C8)-2.0*B6*C6
      GO TO 200
  120 J=K+1
      A(K,J-1)=(C1/C5-C8)/C3
      A(K,J)=RHOC(J)*VXC(J)/DELX-(C1/C5-C8)/C3
      GO TO 200
  130 ICE=MYMAX-2
      GO TO 500
      DO 131 K1=1,ICE
  131 A(K,K1)=RHOC(K1)*VXC(K1)*(Y(K1+1)-Y(K1))
      A(K,K)=RHOC(K)*VXC(K)*(YC(K)-Y(K))+RHOC(K)*VYC(K)*DELX*0.5
     1-RHOC(K)*DC(K)*DELX/(2.0*(YC(K)-YC(K-1)))
      A(K,K-1)=RHOC(K-1)*VXC(K-1)*(Y(K)-Y(K-1))+RHOC(K)*DC(K)*DELX/
     1(2.0*(YC(K)-YC(K-1)))
  500 CONTINUE
      YCM1=0.0
      C1=RHO(2)*D(2)-RHO(1)*D(1)
      C2=YC(2)-YCM1
      C14=-C2
      C4=YC(2)-YC(1)
      C12=-C4
      C5=Y(2)-Y(1)
      C8=RHOC(1)*DC(1)
      C7=RHOC(1)*VXC(1)
      C3=YC(1)-YCM1
      C13=-C3
      C8=RHOC(1)*VYC(1)
      B1=C12/(C13*C14)
      B2=1.0/C12-1.0/C13
      B3=-C13/(C12*C14)
      B4=1.0/(C13*C14)
      B5=-1.0/(C12*C13)
      B6=1.0/(C12*C14)
      A(50,1)=C7/DELX-B2*(C1/C5-C8)-2.0*B5*C6
      A(50,2)=-B3*(C1/C5-C8)-2.0*B6*C6
      CONST=-WALL*B1*(C1/C5-C8)-2.0*B4*C6*WALL
  200 CONTINUE
      I=2
      A(INDEX,MYMAX)=XMASS(1)*DELX
      DO 90 IP=1,MYMAX
      DO 90 IQ=1,INDEX
   90 A(IQ,IP)=A(IQ,IP)*1.0E-06
      IF(I.EQ.1) GO TO 92
      WRITE(6,1010)((A(L,J),J=1,MYMAX),L=1,INDEX)
 1010 FORMAT(10E12.3)
   92 CONTINUE
C     INVERT THE A MATRIX
      CALL MATRIX(10,INDEX,MYMAX,0,A,51,IND)
C     IF IT EQUALS 2,THIS IS THE OFF MATRIX--AX
C     IF IT EQUALS 1,THIS IS THE ON MATRIX--A
      IF(IT.EQ.2) GO TO 100
      GO TO 70
  100 CONTINUE
      DO 102 L=1,INDEX
      DO 102 L1=1,MYMAX
  102 AX(L,L1)=A(L,L1)
```

```
 70   CONTINUE
      GO TO(105,106),IT
105   MIN=2
      MAX=MDOT+1
      GO TO 107
106   MIN=MDOT+2
      MAX=MDOT+MDOTNO+1
107   CONTINUE
      DO 300 I=MIN,MAX
      XPRNT= (I-1)*DELX + (NTIM-1)*(MDOT+MDOTNO)*DELX
      IF(XPRNT.EQ.2.5)GO TO 1017
      IF(.NOT.IPRNT)GO TO 264
1017  CONTINUE
      WRITE(6,1011)XPRNT
1011  FORMAT(53H1 ONE DIMENSIONAL DIFFUSION WITH TWO DIMENSIONAL FLOW/
     111HOSTATION=---,E11.3,4H CM.)
      WRITE(6,1012)
1012  FORMAT(1H03X6HY CELL4X12HV(Y)/V(EDGE)2X15HDENS.*VEL.*M.F.
     12X13HMASS FRACTION2X13HFLUX PER CELL16H DENSITY G/CM**36X10HX VELO
     2CITY)
264   CONTINUE
      IC=INDEX-1
      DO 150 K=1,IC
      J=K+1
      C7= RHOC(J)*VXC(J)
      BX(K,1) = C7*F(I-1,J)/DELX
150   CONTINUE
      DO 160 K1=1,IC
160   SU=SU+RHOC(K1)*VXC(K1)*F(I-1,K1)*(Y(K1+1)-Y(K1))
      IC=IC+1
      SU=SU+RHOC(IC)*VXC(IC)*(YC(IC)-Y(IC))*F(I-1,IC)+XMASS(I)*DELX
     1+RHOC(IC)*VYC(IC)*DELX*0.5*F(I-1,IC)
     2-(RHOC(IC)*DC(IC)*DELX/(2.0*(YC(IC)-YC(IC-1))))*
     3(F(I-1,IC-1)-F(I-1,IC))
      BX(IC,1)=SU
      BX(IC,1)=C7*F(I-1,J)/DELX-CONST
      SU=0.0
      DO 165 K=1,INDEX
165   BX(K,1)= BX(K,1)*1.0E-06
      GO TO(170,175),IT
175   CALL MATRIX(20,INDEX,INDEX,1,AX,51,BX,1,FX,51)
      GO TO 180
170   CALL MATRIX(20,INDEX,INDEX,1,A,51,BX,1,FX,51)
180   CONTINUE
      DO 190 I1=1,INDEX
      RATIO= VXC(I1)/VX(MYMAX)
      F(I,I1) = FX(I1,1)
      RUTS=RHOC(I1)*VXC(I1)*F(I,I1)
      XFLUX= RHOC(I1)*VXC(I1)*F(I,I1)*(Y(I)+1)-Y(I1))
      IF(XPRNT.EQ.2.5) GO TO 1018
      IF(.NOT.IPRNT) GO TO 190
1018  CONTINUE
      WRITE(6,1013)I1,RATIO,RUTS,F(I,I1),XFLUX,RHOC(I1),VXC(I1)
1013  FORMAT(I10,6E16.6)
190   CONTINUE
      LAST=RHOC(50)*VXC(50)*(YC(50)-Y(50))*F(I,50)
      IF(IT.EQ.2) GO TO 1102
      DO 501 NN=1,49
      AL=(ALOG(WALL)-ALOG(F(I,NN)))/YC(NN)
      TERM=EXP(-1.0*AL*Y(NN+1))-EXP(-1.0*AL*Y(NN))
      TERM=RHOC(NN)*VXC(NN)*(-1.0/AL)*TERM
      TERM=WALL*TERM
```

```
501   SUM=SUM+TERM
      SUM=SUM+LAST
      WRITE(6,502)XPRNT,SUM
502   FORMAT(14H TOTAL MASS AT,E11.3,7H CM. IS,E11.3,4H GM.)
1102  CONTINUE
      SUM=0.0
      IF(IPRNT)GO TO 261
      IPRNT=.TRUE.
      GO TO 263
261   IPRNT=.FALSE.
263   CONTINUE
300   CONTINUE
      GO TO (210,220),IT
210   IT=2
      DO 230 I=1,INDEX
230   VYC(I)=0.0
      XALL=0.0
      GO TO 240
220   CONTINUE
      IT=1
      DO 250 NU=1,INDEX
250   F(1,NU)=F(MDOT+MDOTNU+1,NU)
60    CONTINUE
      STOP
      END
```

REFERENCES

1. Cowperthwaite, R. L., McEwan, M. D., User's Manual, Vol II, Ablation Mass Loss Program, AFWL-TR-67-141, Vol II, Air Force Weapons Laboratory, Kirtland AFB, NM, June 1968.

2. Fivel, H. J., Schaffer, M. J., Buggeln, R. C., User's Manual, Vol I, Chemical Composition Program, AFWL-TR-67-141, Vol I, Air Force Weapons Laboratory, Kirtland AFB, NM, January 1968.

3. Cheng, H. K., Hypersonic Flow with Combined Leading Edge Bluntness and Boundary Layer Displacement Effect, Cornell Aeronautical Laboratory Report AF-1285-A-4, August 1960.

4. Mirels, H., Thornton, P. R., Effect of Body Perturbations on Hypersonic Flow over Slender Power Law Bodies, NASA TR-R-45, National Aeronautics and Space Administration, 1959.

5. Cowperthwaite, R. L., Aungier, R. H., Ferrell, J. E., User's Manual, Vol IV, Vehicle Deformation Program, AFWL-TR-67-141, Vol IV, Air Force Weapons Laboratory, Kirtland AFB, NM, To Be Published.

6. Hudgins, H. E., Supersonic Flow about Right Circular Cones at Zero Yaw in Air at Thermodynamic Equilibrium, Parts II and III, Tables of Data, USA (AMUCOM) Report PA-TM-1493, Parts II and III, September 1964.

7. Spalding, D. B., Auslander, D. M., Sundaram, T. R., "The Calculation of Heat and Mass Transfer through the Turbulent Boundary Layer on a Flat Plate at High Mach Numbers, With and Without Chemical Reaction," In: Supersonic Flow, Chemical Processes and Radiative Transfer, Pergamon Press, New York, 1964.

8. Spalding D. B., Chi, S. W., "The Drag of a Compressible Turbulent Boundary Layer on a Smooth Flat Plate With and Without Heat Transfer," Journal of Fluid Mechanics, Vol 18, No. 1, January 1964.

9. Horton, T. E., The JPL Thermochemistry and Normal Shock Computer Program, JPL TR-32-660, Jet Propulsion Laboratory, Pasadena, California, November 1964.

10. Wagner, R. D., Watson, R., Induced Pressure and Shock Shapes on Blunt Cones in Hypersonic Flow, NASA TN-D-2182, National Aeronautics and Space Administration, 1964.

11. Hayes, W. D., Probstein, R. F., Hypersonic Flow Theory, 2nd Ed., Academic Press, New York, 1966.

12. Wittliff, C. E., Curtis, J. T., Normal Shock Wave Parameters in Equilibrium Air, Cornell Aeronautical Laboratory Report CAL-111, November 1961.

13. Marrone, P. V., Normal Shock Waves in Air: Equilibrium Composition and Flow Parameters for Velocities from 26,000 to 50,000 Ft/Sec, Cornell Aeronautical Laboratory Report AG-1729-A-2, August 1962.

REFERENCES (cont'd)

14. Hansen, C. F., Approximations for the Thermodynamic and Transport Properties of High-Temperature Air, NASA TR-R-50, National Aeronautics and Space Administration, 1959.

15. Romig, M. F., Conical Flow Parameters for Air in Dissociation Equilibrium, Convair Research Report 7, Convair Scientific Research Laboratory, May 1960.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Air Force Weapons Laboratory (WLDE) Kirtland Air Force Base, New Mexico 87117	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

USER'S MANUAL, Vol V, Turbulent Boundary Layer Program (REBOUND)

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

June 1966-October 1967

5. AUTHOR(S) (First name, middle initial, last name)

H. J. Fivel; M. J. Schaffer; Capt Jimmie D. Young

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May 1969	460	16

8a. CONTRACT OR GRANT NO. AF 29(601)-7225 b. PROJECT NO. 5791 c. Task No. 27 d.	9a. ORIGINATOR'S REPORT NUMBER(S) AFWL-TR-67-141, Vol V 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of AFWL (WLDE), Kirtland AFB, NM, 87117. Distribution is limited because of the technology discussed in the report.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY AFWL (WLDE) Kirtland AFB, NM 87117

13. ABSTRACT

(Distribution Limitation Statement No. 2)

This report is one of six volumes providing user's manuals for a series of computer programs developed for the Air Force Weapons Laboratory, ~~by McDonnell Douglas Corporation~~. The program described in this volume treats the inviscid flow field, ablation mass loss, and turbulent boundary layer on a spherically blunted conical body. The principal advantage of this formulation is the use of a real gas equation of state for the ablation products air ~~mixture~~. ~~This~~ report describes the theory and calculation procedure for the program and presents sample cases for clarifying the data input. The program is written in FORTRAN VI computer language and has been run on both the CDC 6400 and CDC 6600 computers.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Computer program						
Turbulent boundary layer						
Viscous flow						
Plasma sheath						
Ablation						
Missile reentry						