N73-11263

ANALYTICAL STUDY OF STRIATED NOZZLE FLOW WITH SMALL RADIUS OF CURVATURE RATIO THROATS

TEES-1092-TR-72-01

by

DAVID J. NORTON AND ROBERT E. WHITE

TEES

TEXAS ENGINEERING EXPERIMENT STATION
TEXAS A&M UNIVERSITY
COLLEGE STATION TEXAS 77843

August 1972

Prepared for
NASA Manned Spacecraft Center
R&D Contracts Branch
Houston, Texas 77058
Under Contract NAS 9-11658

CR-128619

Analytical Study of Striated Nozzle Flow with Small Radius of Curvature Ratio Throats

to the

National Aeronautics and Space Administration
Manned Spacecraft Center

research performed under

Contract No. NAS 9-11658

by

David J. Norton, Principal
Investigator and Assistant Professor
of Aerospace Engineering

and

Robert E. White Research Assistant
of Aerospace Engineering

September, 1972

Texas A&M University
Texas Engineering Experiment Station
Space Technology Division

FORWARD

This report describes the work performed at Texas A&M University under NASA Contract No. NAS 9-11658, "Analytical Study of Striated Nozzle Flow with Small Radius of Curvature Throats," The authors would like to thank the technical monitor for the above contract, Lonnie Jenkins, NASA Manned Spacecraft Center, Houston, Texas. The authors would also like to thank Chester Vaughn of the Auxililary Propulsion and Pyrotechnics branch for his assistance.

CONTENTS

page

page

LIST OF TABLES

LIST OF FIGURES

SUMMARY

An analytical method has been developed which is capable of estimating the chamber and throat conditions in a nozzle with a low radius of curvature throat. The method has been programmed using standard FORTRAN IV language and includes chemical equilibrium calculation subprograms (modified NASA Lewis program CEC71) as an integral part. Program CEC71 is described in Reference 1. The method determines detailed and gross rocket motor characteristics in the presence of striated flows and gives detailed results for the motor chamber and throat plane with as many as 20 discrete zones. The method employs a simultaneous solution of the mass, momentum, and energy equations and allows propellant types, O/F ratios, propellant distribution, nozzle geometry, and injection schemes to be varied so to predict spatial velocity, density, pressure, and other thermodynamic variable distributions in the chamber as well as the throat. Results for small radius of curvature (down to R_C/R_T=.25) have shown good comparison to experimental results. Both gaseous and liquid injection may be considered with frozen or equilibrium flow calculations. For parametric studies it is also persible to elect frozen flow with calorically perfect gases in order to conserve computational time. In this case it is possible to study the effects of complete mixing compared to striated flow.

INTRODUCTION

Objective

The objective of this study has been to develop a computer program for determining the performance of rocket engines in the presence of striated flows and small radius of curvature throats and to perform parametric studies to determine the influence of engine parameters on mass flow rate, specific impulse, and chamber pressure. The program has been equipped with subprograms (a modified form of CEC71 computer program of Reference 1) to determine the thermochemical composition for steady, inviscid flow in discrete zones. The program has the capability of analyzing the chamber flow field for specified injector schemes with gaseous or liquid propellant injection and of analyzing the inviscid, transonic throat plane flow field. Predictions are made of the chamber operating conditions and the spatial distribution of velocity and thermodynamic variables for the reactants and products of combustion. Parameters which are considered for studies include the propellant types (especially gaseous O_2 and H_2), O/F ratios, propellant mass distributions, nozzle geometry, and injection schemes. Approximate two dimensional performance parameters at the throat plane are determined and compaired to one dimensional performance results. The philosophy used to develop this computer program is discussed in Appendix A.

Considerations and Assumptions

In small rocket motors heat transfer can be significant in terms of engine efficiency as well as thermal protection requirements. One method of reducing heat transfer is to reduce the heat transfer coefficient. This can sometimes be accomplished by laminarization of the boundary layer by using a highly convergent subsonic section of a nozzle. Experimental work[2,3] has shown that there is a tendency with a rapidly converging subsonic section to stablize the boundary layer as a result of the rapid acceleration of the working fluid. This tendency toward laminarization of the boundary layer through the transonic throat region may be continued by employing a low throat radius of curvature ratio throat. In addition to the effect of reducing the heat transfer a decrease in nozzle surface area can be achieved. Further reductions in overall heat transfer to the nozzle walls may be effected by use of barrier cooling. By surrounding the hot core flow with fuel or ozidizer rich flow, temperature gradients to the motor walls are reduced thus the heat transfer is reduced. Striating therefore permits maximum temperature in the core zone and yet permits near maximum overall specific impuse.

In the treatment of striated flow with throat radius of curvature effects no closed form relationship exists between the chamber pressure and the mass flow. This is due to variations of total enthalpy and velocity in the chamber because of striation, as well as, radial pressure gradients at the physical throat due to radius of curvature effects. Thus choking with a given throat radius can only be assured and the flow parameters determined with a coupled solution of the throat and the chamber.

Even for unstriated nozzle flow the solution for the two dimensional flow field with small radius of curvature ratio throats is difficult and time consuming[4]. With inclusion of striation the problem becomes more complex and determination of performance parameters for a parametric analysis may require considerable machine computations. Radial velocities at the physical throat plane are thus assumed to be small and the throat is a revolved circular arc section. Figures 1a and 1b present the flow regime of interest in which mass, momentum, and energy are conserved along streamlines. This analysis is for inviscid, compressible, axially symmetric flow. Each annular flow zone is assumed to flow isentropically after leaving the chamber and satisfy conservation equations of mass and momentum at the throat. Losses are limited to curvature and striation effects and to the degree of equilibrium flow attained (kinetic or energy conversion losses). The energy conversion losses are dependent on the degree of freezing considered for the solution and is restricted to limiting cases. In this analysis it was not necessary to average the mass flow nor to require the isentropic coefficient to be constant in a flow zone or stream tube at the throat.

The chamber static pressure is the pressure at which chamber combustion occurs. This pressure is used for thermochemical calculations for each flow zone. In calculating an individual flow zone total temperature the difference between the kinetic energies of the burned gas and the injected reactants in the chamber zone is not considered since the difference is small compared to the overall energy of the material in the combustion zone. Thus, even for low contraction ratios the stagnation temperature in a flow zone determined with the chamber static pressure differs very little from the exact chamber total temperature for the zone.

THEORY

1.0 Chamber Solutions

The chamber solution depends on the method of mass generation and on the properties of the injected and combustion materials. To obtain choked conditions at the throat one may have to alter the chamber combustion static pressure or the total mass flow and recalcualte throat plane parameters. By this technique the coupling is maintained between the chamber solution and the throat plane solution. By multiple iteration the complete solution can be made to simultaneously satisfy chamber and throat requirements.

For a discrete number of stream tube zones as shown in Figure 1a, the mass, momentum, and energy equations for one dimensional, inviscid, compressible flow are:

$$\rho_{I_{fu_i}} w_{I_{fu_i}} A_{I_{fu_i}} + \rho_{I_{ox_i}} w_{I_{ox_i}} A_{I_{ox_i}} = \dot{m}_{I_{fu_i}} + \dot{m}_{I_{ox_i}} = \dot{m}_i \qquad (1.1)$$

$$= \rho_{g_i} w_{g_i} A_{g_i}$$

$$\dot{m}_{I_{fu_i}} w_{I_{fu_i}} + \dot{m}_{I_{ox_i}} w_{I_{ox_i}} + A_{g_i} P_I = P_g A_{g_i} + \dot{m}_i w_{g_i} \qquad (1.2)$$

$$H_{o_{I_i}} = H_{o_{g_i}} = h_{g_i} + w_{g_i}^2/2 \qquad (1.3)$$

where

$$h_{g_i} = \int_{T_o}^{T_{g_i}} C_{p_{g_i}} dT + h^{\circ}_{g_i}$$

In addition, it is necessary that

$$\Sigma A_{g_i} = A_c \tag{1.4}$$

$$\dot{m}_{I_{ox_i}} = (O/F)_i \, \dot{m}_{I_{fu_i}} \tag{1.5}$$

Further, if the injected propellants are gaseous and perfect gases are assumed

$$\rho_{I_{fu_i}} = P_I / [R_{I_{fu_i}} T_{I_{fu_i}}] \tag{1.6}$$

and

$$\rho_{I_{ox_i}} = P_I / [R_{I_{ox_i}} T_{I_{ox_i}}] \tag{1.7}$$

For a constant chamber static pressure, the propellant distributtion in each zone is specified as the fraction of the total mass flow. This case is of interest when it is necessary to compare engine performance at a fixed pressure. The total mass flow is determined by iteration and is not known apriori. An initial estimation of the total mass flow or an adjustment by iteration of the total mass flow permits intermediate calculations and allows determination of parameters in the chamber. Thus, iteration between the chamber solution and the throat plane solution drives the intermediate results to the solution values of parameters. An analysis with a constant chamber static pressure is a practical mode to consider for parametric studies since the chamber static pressure, propellant reactants inputs, and O/F in each zone, once specified, do not change results of chemical equilibrium calculations for the mass flow adjustments required to determine the solution. The solution being that for the chamber and the throat plane with a specified throat radius of curvature. Equations (1.1) through (1.4) provide a system of equations to which the unknowns T_{g_i}, W_{g_i}, A_{g_i} and P_I may be determined.

For <u>specified mass flow in each zone</u>, there exists a chamber static pressure which simultaneously satisfies Equations (1.1) through (1.4). By application of the governing equations at the throat plane, the throat radius determined is the prescribed throat radius. For this scheme of mass generation, chemical equilibrium calculations would be required at each iteration step. Initial estimates of thermodynamic variables are altered once to provide data for intermediate calculations and a final solution.

For <u>specified injector reservior pressures</u>, injection equations for mass generation due to the reservior pressure for the fuel mass flow injected into each zone is determined by

$$\dot{m}_{I_{fu_i}} = C_{D_{I_i}} A_{I_{fu_i}} [2\rho_{I_{fu_i}} (P_{R_{fu_i}} - P_I)]^{1/2} \tag{1.8}$$

and thus

$$\dot{m}_i = \dot{m}_{I_{fu_i}} (1 + O/F_i) \tag{1.9}$$

and

$$\dot{m}_{I_{ox_i}} = \dot{m}_i - \dot{m}_{I_{fu_i}} \tag{1.10}$$

This scheme is limited in the program by the requirement that injector fuel areas are required to be specified. Intermediate results are determined by a double iteration since the injector pressure, P_I, is also determined within the chamber solution iteration loop. Equation (1.8) is not strictly applicable to gaseous injection but is operational for liquid or gas injection in the computer program .

For cases where frozen, calorically perfect flow is considered, a solution for <u>fully mixed flow</u> may be included. The results give a limiting case for comparison to striated flow. Appropriate thermodynamic and flow variables may be calculated as follows:

$$C_{P_{g_{mix}}} = \Sigma\, C_{P_{g_i}} \dot{m}_i/\dot{m}_{total} \tag{1.11}$$

$$MW_{g_{mix}} = 1/\Sigma\, \frac{\dot{m}_i}{\dot{m}_{total}} \cdot \frac{1}{MW_{g_i}} \tag{1.12}$$

$$R_{g_{mix}} = \Re\, g_c/MW_{g_{mix}} \tag{1.13}$$

$$\gamma_{g_{mix}} = C_{P_{g_{mix}}}/(C_{P_{g_{mix}}} - R_{g_{mix}}) \tag{1.14}$$

$$T_{o_{g_{mix}}} = \frac{\Sigma\, C_{P_{g_i}} T_{o_{g_i}} \dot{m}_i}{C_{P_{g_{mix}}} \dot{m}_{total}} \tag{1.15}$$

Conservation of energy for this case requires

$$C_{P_{g_{mix}}} T_{o_{g_{mix}}} = h_{g_{mix}} + w_{g_{mix}}^{2}/2 \tag{1.16}$$

but

$$h_{g_{mix}} = C_{P_{g_{mix}}} T_{g_{mix}} = C_{P_{g_{mix}}} \frac{P_{g_{mix}}}{\rho_{g_{mix}} R_{g_{mix}}}$$

$$= C_{P_{g_{mix}}} \frac{P_{g_{mix}} w_{g_{mix}} A_g}{\dot{m}_{total} R_{g_{mix}}} \tag{1.17}$$

so

$$C_{p_{g_{mix}}} T_{o_{g_{mix}}} = C_{p_{g_{mix}}} \frac{P_{g_{mix}} w_{g_{mix}} A_g}{\dot{m}_{total} R_{g_{mix}}} + w_{g_{mix}}^2/2 \tag{1.18}$$

Conservation of momentum requires

$$P_{g_{mix}} A_g + \dot{m}_{total} w_{g_{mix}} = P_g A_g + \Sigma \dot{m}_i w_{g_i} \tag{1.19}$$

or

$$P_{g_{mix}} = P_g + \frac{\Sigma \dot{m}_i w_{g_i}}{A_g} - \frac{\dot{m}_{total} w_{g_{mix}}}{A_g} \tag{1.20}$$

Elimination of $P_{g_{mix}}$ from Equation (1.18) with Equation (1.20) gives a quadratic in $w_{g_{mix}}$. Equation (1.20) gives $P_{g_{mix}}$ and $\rho_{g_{mix}}$ may be determined by

$$\rho_{g_{mix}} = \dot{m}_{total}/A_g w_{g_{mix}} \tag{1.21}$$

Also,

$$P_{o_{g_{mix}}} = P_{g_{mix}} \left(\frac{T_{o_{g_{mix}}}}{T_{g_{mix}}} \right)^{\frac{\gamma_{g_{mix}}}{\gamma_{g_{mix}} - 1}} \tag{1.22}$$

If only <u>one zone of gas flow for a low temperature or nonreacting gas</u> is the subject of investigation, a simplified calculation option in the program may be used. Injector details are not determined and thermochemical data are input as opposed to determined by chemical equilibrium calculations.

The energy equation, mass equation, and equation of state are simultaneously solved. By elimination of the injector pressure, P_I, from the list of unknowns, the momentum equation is not required. This option in the program serves to predict radius of curvature effects at the throat plane for low temperature single zone gas flow.

2.0 Throat Solution

The properties of the combustor flow field are determined with the combustion chamber calculations prior to the throat solution. Initially, estimates are made followed by iterative corrections until the converged solution results for desired fixed parameters. The momentum and mass equations describe the flow at the throat. These equations together with the requirement that energy be conserved in stream tubes defines the flow. A schematic representation of the geometry of a typical nozzle throat is given in Figure 1b.

The radial momentum equation is:

$$\partial P/\partial r = - \rho[u\ \partial u/\partial r + w\ \partial u/\partial z - v^2/r] \tag{2.1}$$

Considering the throat plane, we expect that the radial velocity will be small everywhere since it must be zero at the throat wall and at the nozzle centerline. Thus, for axially symmetric flow

$$dP/dr = -\rho w\ du/dz \tag{2.2}$$

Since $dr/dz = u/w$, then on the throat plane between the circular arc throat wall and the centerline

$$dP/dr = -\rho w^2 d^2 r/dz^2 = -\rho w^2/R_c \tag{2.3}$$

In the above equation R_c refers to the radius of curvature of the streamlines at the throat. The energy equation applied at the throat where $u = 0$ requires

$$w^2 = 2(H_o - h) \tag{2.4}$$

For frozen, calorically perfect flow, the isentropic coefficient, γ, is constant in each stream tube and it is convenient to express the enthalpy in stream tubes as:

$$h = \frac{\gamma}{\gamma-1}[P]^{\frac{\gamma-1}{\gamma}}[C]^{1/\gamma} \tag{2.5}$$

Where the constant C for each stream tube is defined by the isentropic process equation as:

$$C = C(\psi) = P/[\rho]^{\gamma} \tag{2.6}$$

The equation for mass conservation is:

$$d\psi/dr = d\dot{m}/dr\left(1/\dot{m}_{total}\right) = \frac{2\pi r\rho w}{\dot{m}_{total}} \tag{2.7}$$

Applying Equation (2.4) to Equations (2.3) and (2.7) it can be seen that a set of two nonlinear total differential equations result. The solution of these equations can be obtained at the throat providing C, Ho, γ, R_c can be specified as a parametric function of ψ. Of course the variation of C, H_o, and γ with ψ is known for the current iteration from the chamber solution (at least for frozen, calorically perfect flow: for equilibrium or frozen flow these parameters are determined as a function of pressure for a given chamber solution). It remains only to specify a relationship for R_v to close the formulation at the throat plane. It is clear from experimental and analytical studies that the main functional form for R_v should be $R_v = R_v(R_C, R_T, r)$. This formulation for R_v neglects effects due to inlet geometry between the chamber and the throat region; however, experimental evidience indicates this to be a minor variation except for contraction section wall slopes near perpendicular to the nozzle axis and/or $R_C/R_T \cong 0$.[5,6,7] The appropriate boundary conditions for R_v regardless of striation effects are:

$$R_v = R_C \quad \text{at} \quad r = R_T$$

$$R_v = \infty \quad \text{at} \quad r = 0$$

A plausible form for R_v which satisfies these conditions is

$$R_v = R_C \left[\frac{R_T(1 + R_C/r) - r}{R_C} \right]^n \tag{2.8}$$

Figure 2 presents a number of calculations for C_D with respect to R_C/R_T for various values of n. Based on these results a value of $n = 3/2$ is indicated. The representation chosen for R_v can be further evaluated by comparing results for the variation of throat plane Mach number with radius. This will be presented later in the report.

Equations (2.3) and (2.7) thus comprise the necessary equations, expressed in a more general form than previously considered[8], since $H_o = H_o(\psi)$, $h = h(P, \psi)$, $\rho = \rho(P, \psi)$, and $w = w(H_o, h) = w(P, \psi)$. In general, $H_o(\psi)$ is determined from chamber conditions and is calculated before equations at the throat are solved. If required, $C(\psi)$ is also determined from chamber conditions. Throat plane variables are determined so that the throat radius which just chokes the flow must be the prescribed radius. An iteration between the chamber solution and the throat plane solution allows starting estimations to be corrected to assure the final iteration predicts choked flow for the desired geometry and fixed parameters. For simultaneous solution of the throat plane equations, Hamming's prediction-corrector technique[9] was employed. This method proved to be efficient and stable and is initiated with the Range-Kutta method so to be self starting.

3.0 Performance Solution

Two types of performance calculations are made. The first type includes the effects of throat radius of curvature and striation. The total mass flow is given by

$$\dot{m}_{total} = \Sigma\, \dot{m}_i \tag{3.1}$$

The throat vacuum specific impulse is determined by integrating the velocity and pressure profiles across the throat plane so that

$$I_{sp_{vac}} = \int_{r=0}^{r=R_T} w d\psi + \int_{r=0}^{r=R_T} PdA/\dot{m}_{total} \tag{3.2}$$

The characteristic velocity is

$$c^* = \frac{A_T\, \Sigma\{\dot{m}_i\, P_{o_{g_i}} / \dot{m}_{total}\}}{\dot{m}_{total}} \tag{3.3}$$

Within the program, intergrations are actually represented as finite summations based on the fraction of the mass or area over which the required thermodynamic variable is representative. In Equation (3.2) the velocity at a point on the throat plane is assigned a mass fraction determined by adjacent values of ψ and pressure is assigned the area over which it is representative. Small independent variable step sizes input to the program permit the integration approximation technique used to be accurate.

For comparison, the second type of performance calculations are based on the ideal or optimum distribution of Mach number uniformly equal to one on the throat plane. Although this distribution does not in general occur for one dimensional striated flow[8], it serves as a limit to which striated flow with radius of curvature effects may be compaired. If frozen, calorically perfect flow is considered, the one dimensional mass flow is:

$$\dot{m}_{total_{1-D}} = \Sigma\, A_i^* \, P_{o_{g_i}} \, (\Gamma'_{g_i}/T_{o_{g_i}})^{1/2} \tag{3.4}$$

where

$$\Gamma'_{g_i} \equiv \frac{\gamma_{g_i}}{R_{g_i}} \left[\frac{2}{\gamma_{g_i}+1}\right]^{\frac{\gamma_{g_i}+1}{\gamma_{g_i}-1}} \tag{3.5}$$

If frozen or equilibrium flow is considered, the one dimensional mass flow is determined by

$$\dot{m}_{total} = \Sigma\, A_i^* \, P_g/c^*_{1-D_i} \tag{3.6}$$

where $c^*_{1-D_i}$ is determined by the chemical equilibrium calculations and is based on P_g. A_i^* in either case is determined by the throat solution and is the area of a particular flow zone on the throat plane. In a similar manner the one dimensional vacuum specific impulse at the throat is

$$I_{sp_{vac_{1-D}}} = \Sigma \frac{\dot{m}_i}{\dot{m}_{total}} \left[\gamma_{g_i} R_{g_i} \frac{T_{o_{g_i}}}{(T_{o_{g_i}}/T_i)_{1-D}}\right]^{1/2} + \frac{\Sigma\, P_{o_{g_i}} (P_i/P_{o_{g_i}})_{1-D} A_i^*}{\dot{m}_{total_{1-D}}} \tag{3.7}$$

where

$$(T_{o_{g_i}}/T_i)_{1-D} \equiv \frac{1+\gamma_{g_i}}{2} \tag{3.8}$$

and

$$(P_i/P_{o_{g_i}})_{1-D} \equiv P_{o_{g_i}} \left[T_i/T_{o_{g_i}}\right]^{\frac{\gamma_{g_i}}{\gamma_{g_i}-1}} \tag{3.9}$$

for frozen, calorically perfect flow. For frozen or equilbirum flow

$$I_{sp_{vac_{1-D}}} = \Sigma I'_{sp_{vac}} \dot{m}_i/\dot{m}_{total} \tag{3.10}$$

where $I'_{sp_{vac}}$ is determined by the chemical equilibrium claculations.

The characteristic velocity for one dimensional calculations is

$$c^*_{1-D} = \frac{A_T \, \Sigma(\dot{m}_i/\dot{m}_{total})(P_{o_{g_i}})}{\dot{m}_{total_{1-D}}} \tag{3.11}$$

4.0 Chamber and Throat Iteration Techniques

A chamber solution iteration is necessary to assure that P_I, A_{g_i}, T_{g_i}, and w_{g_i} satisfy conservation equations. The injector pressure, P_I, converges after an initial estimate is made by the program. The iteration equation employed is as follows:

$$P_I = P_g + (P_I - P_g) \left(\frac{\Sigma A_{g_i}}{A_C}\right)^2 \tag{4.1}$$

A double iteration was employed to determine T_{g_i}, A_{g_i}, and w_{g_i}. This technique employes a simultaneous solution of the energy, mass, and momentum equations with the assumed value of P_I as initially estimated or, after the first chamber solution iteration loop is completed, as determined by

Equation (4.1). Convergence is achieved after the above procedure is complete and the following is determined to be true

$$\Sigma A_{g_i} = A_C \tag{4.2}$$

The injector pressure, P_I, is the pressure at each injector surface in the chamber and is the same for each zone. Within the program P_I is a dimensioned variable and is identical with the variable PIAVG. If convergence fails for some reason, variables P_g, PIAVG, ΣA_{g_i}, and A_C are printed in an error message. Convergence difficults may be encountered for a single zone of flow of low temperature or non reacting gas. (This is because w_g becomes large for this case). Special subroutine SPPG has been provided in the program to avoid convergence difficulties should they occur for the above mentioned single zone flow case.

To initiate the throat plane solution an estimate of the centerline pressure is required. The estimation is based on a small perturbation method[10]

$$P_{CL} = P_{o_{g_1}} \left[\frac{2}{\gamma_{g_1} + 1}\right]^{\frac{\gamma_{g_1}}{\gamma_{g_1} - 1}} \left(1 + \frac{1}{\frac{4}{\gamma_{g_1}} \quad R_C/R_T + 1} \right) \tag{4.3}$$

With the initial estimate, throat equations are integrated by Hamming's method[9] to determine the throat wall radius corresponding to $\psi = 1.0$. The pressure, P_{CL}, given by Equation (4.3) is increased and a new wall radius is determined by integration. Following this, dr_w/dP_{CL} is used to alter P_{CL} until the minimum radius is determined. The iteration process by which $dr_w/dP_{CL} = 0$ is sought requires that dP_{CL} be sufficiently small and that

$r_{w_{min}} = R_T$. Only seven iterations are permitted by the program once $r_{w_{min}}$ is bracketed. A convergence message is provided by the program if seven iterations are not sufficient; however, calculations are not halted but procede to alter P_g or $\dot{m}_{total}$ such that $r_{w_{min}} = R_T$. The minimum wall radius as determined by the above procedure is used to alter the chamber static pressure by

$$P_g = P_g(r_{w_{min}}/R_T)^2 \tag{4.4}$$

for fixed mass flow or fixed reservior pressure solutions. For fixed chamber static pressure, the total mass flow is altered to obtain $r_{w_{min}} = R_T$ by

$$\dot{m}_{total} = \dot{m}_{total}\ (R_T/r_{w_{min}})^2 \tag{4.5}$$

If attempts to determine P_g or $\dot{m}_{total}$ for the solution exceed seven, another convergence message is provided and parameters as described in the program output are, nevertheless, determined. Results may be erroneous but sufficient for parametric studies. The most flagrant error occurs if the P_{CL} estimation is very inaccurate. This error may occur for $R_c/R_T \lesssim .25$ so are seldom encountered. This usually causes the pressure, P, to become zero and internally generated messages occur and machine generated messages may occur. Calculations are not necessarily halted and corrections provided in the program may allow the solution to be determined.

5.0 Results and Conclusions

In this section equations which were presented under Throat Solution and Performance Solution will be more thoroughly qualified. Results of the present analysis are discussed and where possible compared with experimental data and with other methods of analysis. The analytical form used for the flow radius of curvature at the throat plane is presented and verified by comparison to experimental data. Results of the many options available in the program will not all be presented; however, several typical results will be presented and discussed. Results of parametric studies are presented as well as performance calculation results.

5.1 Effect of Radial Velocity at the Throat Plane

Consider an axisymmetric converging-diverging nozzle shown in Figure 2. An element of mass flow, $d\dot{m}$, passes through surface dl inclined to the axis at angle ϕ. The element of mass has velocity V inclined at angle σ to the nozzle axis and is da wide. The elemental mass flow is

$$d\dot{m} = 2\pi r \rho V da \tag{5.1}$$

Since

$$da = \sin(\phi-\sigma)\ dl \tag{5.2}$$

and

$$dl = dr/\sin\phi \tag{5.3}$$

then,

$$d\dot{m} = \frac{2\pi\rho V \sin(\phi-\sigma)\ rdr}{\sin\phi} \tag{5.4}$$

On the throat plane $\phi = \pi/2$ so that

$$\dot{m}_{total} = \int_{r=0}^{r=R_T} 2\pi\rho V\cos\sigma\ rdr \tag{5.5}$$

Thus, the discharge coefficient is

$$C_D = \frac{\int_{r=0}^{r=R_T} 2\pi\rho w \, r dr}{\dot{m}_{total_{1-D}}} \tag{5.6}$$

The effect of radial velocity on mass flow at the throat plane is exhibited in the density as shown by Equation (5.5). For single flow zone of a frozen, calorically perfect gas the density may be written as

$$\rho = \rho_{o_g} \left[1 - \frac{\gamma_g - 1}{2} \frac{w^2 + u^2}{(\gamma_g R_g T_{o_g})^2} \right]^{1/\{\gamma_g - 1\}} \tag{5.7}$$

The radial velocity component, u, may be assumed to be negligible at the throat plane compared to w^2 since $u = 0$ at $r = 0$ and at $r = R_T$. Using two dimensional methods of Reference 4 this assumption may be verified for nonstriated flow for values of R_C/R_T down to .35. It is therefore concluded that this assumption will be useful for the present analysis. Recall, however, $\partial u/\partial z \neq 0$ on the throat plane.

5.2 Streamline Radius of Curvature at the Throat Plane

In the discussion of the throat plane solution the radius of curvature of the streamlines was given by Equation (2.8) as

$$R_r = R_C \left[\frac{R_T(1 + R_C/r) - r}{R_C} \right]^n \tag{5.8}$$

This form satisfies the boundary conditions at $r = 0$ and at $r = R_T$ for any n greater than zero. Figure 3 presents calculation results for n = 1.0, 1.5, and 2.0. Based on these results a value of $n = 1.5$ is indicated as the appropriate exponent. In addition, Figure 4 shows good agreement between the present theory throat plane Mach number distribution and experimental data of Reference 5 for R_c/R_T = .625. Other theories are also shown for comparison. Figure 5 presents the distribution of throat plane static pressure to chamber pressure ratio of the present method. Results compare favorably to the data taken by Cuffel, Back, and Massier.[5]

5.3 Results of Analytical Method

The effects of striating mixtures of gaseous hydrogen and oxygen for nozzles with small throat radius of curvature to throat radius ratio having various amounts of individual flow zone mass flow have been investigated. Figure 6 illustrates the dependence of the mass discharge coefficient with respect to the cooling mass flow ratio for R_c/R_T = .5 and P_g = 200 psia. Results are for equilibrium; frozen; frozen, calorically perfect; and frozen, calorically perfect fully mixed flows. Figure 7 shows similar results for R_c/R_T = 2.0. Figures 6 and 7 indicate that if striation and equilibrium are maintained, an improved discharge coefficient may be obtained. Improvement in discharge coefficient results for equilibrium flow compared to fully mixed flow expecially for a large mass of fuel rich cooling. Both Figures 6 and 7 show that little change in discharge coefficient with fraction of cooling mass flow occurs when chamber flow zones burned gases are fully mixed in the chamber.

Figure 8 presents discharge coefficient, C_D, versus radius of curvature ratio, R_c/R_T, for various amounts of cooling with unmixed and fully mixed flows. Input thermochemical data used to calculate results shown in Figure 8 for frozen, calorically perfect flow at the throat plane were determined at 50 psia chamber pressure with constant chamber total enthalpy in flow zones. The core and cooling zones had O/F ratios of 6.0 and 0.5, respectively. As in Figures 6 and 7 striated flow has higher discharge coefficient than uncooled or fully mixed flow for these propellants.

Figure 9 shows the spatial distribution of Mach number and static pressure on the throat plane for R_c/R_T = 0.5, 1.0, 2.0, and 10.0 with four flow zones. The Mach number discontinuities between flow zones appears for all R_c/R_T resulting in a discharge coefficient less than 1.0. A discharge coefficient of unity may be obtained only for a Mach number at the throat plane uniformly equal to 1.0. This point will be expanded later in this section. It also appears that striation may improve the discharge coefficient. This is verified by the hydrogen and oxygen propellant combinations used to determine Figures 6, 7, and 8.

5.4 Effect of Throat Plane Mach Number Distribution on Discharge Coefficient

Based on Equation (2.3) the axial velocity may be found by

$$w^2 = -\frac{R_v}{\rho}\, dP/dr \tag{5.9}$$

Since $u \cong 0$, the density is $\rho = \rho(P,\psi)$ and thus mass flow rate given by Equation (5.5) and discharge coefficient given by Equation (5.6) depend on pressure and the radial pressure gradient on the throat plane, on thermochemical parameters of chamber flow zone gases (on ψ), and on the flow radius of curvature, R_v. For frozen, calorically perfect flow, the mass flow with radius of curvature and striation effects by the present analysis is

$$\dot{m}_{total} = \frac{2\pi}{\sqrt{\mathcal{R}}} \int_{r=0}^{r=R_T} \frac{PM\sqrt{MW}\ \sqrt{\gamma}}{\sqrt{T}}\, rdr \tag{5.10}$$

In the present analysis the one dimensional mass flow for frozen, calorically perfect flow was determined by,

$$\dot{m}_{total_{1-D}} = \frac{\pi R_T^2 / \sqrt{\mathcal{R}}}{\sum \left\{ \left(\frac{\dot{m}_i}{\dot{m}_{total}} \right) \frac{\sqrt{T_{o_{g_i}}}}{P_{o_{g_i}} \sqrt{MW_{g_i}} \sqrt{\gamma_{g_i}} \left[\frac{2}{\gamma_{g_i}+1} \right]^{(\gamma_{g_i}+1)/(\gamma_{g_i}-1)}} \right\}} \tag{5.11}$$

Thus, by taking differential elements of radius with associated thermodynamic variables the discharge coefficient may be written as

$$C_D = \left\{ \sum_i \left[\left(\frac{\dot{m}_i}{\dot{m}_{total}} \right) \frac{\sqrt{T_{o_{g_i}}}}{P_{o_{g_i}} \sqrt{MW_{g_i}} \sqrt{\gamma_{g_i} \left(\frac{2}{\gamma_{g_i}+1} \right)^{(\gamma_{g_i}+1)/(\gamma_{g_i}-1)}}} \right] \right\}$$

$$\cdot \left\{ \sum_j \left[\frac{P_j \sqrt{MW_{g_i}} \sqrt{\gamma_{g_i}} \; M_j}{\sqrt{T_i}} \cdot \frac{r_j^2 - r_{j-1}^2}{R_T^2} \right] \right\} \tag{5.12}$$

For unstraited flow considerable simplification results and the discharge coefficient for frozen, calorically perfect isentropic flow is

$$C_D = \frac{2}{R_T^2} \int_{r=0}^{r=R_T} M \left(1 + \frac{\gamma_{g_1} - 1}{2} M^2 \right)^{-\frac{\gamma_{g_1}+1}{2(\gamma_{g_1}-1)}} \left(\frac{\gamma_{g_1}+1}{2} \right)^{\frac{\gamma_{g_1}+1}{2(\gamma_{g_1}-1)}} r \, dr \tag{5.13}$$

In Equation (5.13) the coefficient is one if the Mach number is uniformly equal to one at the throat plane. For all other Mach number distributions consistent with conservation of mass, momentum, and energy the discharge coefficient is less than one.

That the discharge coefficient of striated flows in the presence of small radius of curvature throats tends to be greater than the discharge coefficient of unstriated flow is not necessarily true for all fuel-oxidizer combinations. Figure 10 presents R_c/R_T versus discharge coefficient examples for comparison. In Figure 10 the hydrogen-oxygen propellant combination with a core O/F = 7.0 and a sheath O/F = 0.0 shows an improved discharge coefficient for 20% striation over unstriated flow. However, for the hydrazine-nitrogen tetroxide combination, which also assumes frozen, calorically perfect flow, the unstriated flow discharge coefficient is greater than the striated flow discharge coefficient. As in Figure 8, Figure 10 employes thermochemical data determined for a chamber static pressure of 200 psia and constant total enthalpy. For Figures 8 and 10 the isentropic exponent in the chamber core flow zone is not equal to the ratio of specific heats. The significant difference between the flows used to calculate results for Figure 10 is that for striated flow using N_2H_4 and N_2O_4 propellant combinations less throat area is required to accommodate N_2H_4 sheath zone mass flux than is required for the H_2 and O_2 propellant combinations using H_2 as a coolant given the same fraction of coolant mass flow to total mass flow.

5.5 Parametric Study Results

Extensive parametric studies have consisted of determining the effect on discharge coefficient due to striation and radius of curvature ratio with frozen, calorically perfect flow. By the direct imput of thermochemical data and the use of a frozen, calorically perfect flow analysis, particular sets of thermodynamic variables were fixed and used to generate parametric curves with thermodynamic parameter combinations. Ratios of specific heats, stagnation temperatures, and molecular weights as well as cooling mass flow to total mass flow appear to be the main parameters required to develop a parametric relationship between striated and unstriated flow discharge coefficients for given throat radius of curvature ratio. Studies have yielded no general parameter combination which correlates striated to unstriated flow discharge coefficient without knowledge of the results of the throat plane solution. However, trends have been observed, namely, there is a significant effect of combustion gas ratios of specific heats in striated flow. Stagnation temperatures and molecular weights of the combustion gasses have only a secondary effect on discharge coefficient. Even for unstriated flow there is a difference between discharge coefficients with different ratios of specific heats. Figure 11 shows that discharge coefficient increases with decreasing ratio of specific heats. In Figure 11 the molecular weight and total temperature of the burned gas remained constant for all curves of R_C/R_T versus C_D.

5.6 Performance Results

The characteristic velocity, c*, versus cooling to total mass flow ratio is shown in Figure 12. Curves are for fully mixed and unmixed flow for radius of curvature ratios, R_C/R_T, equal to 0.5 and 2.0. Under the same flow mixing conditions, i.e. unmixed or fully mixed, and for the same fraction of cooling to total mass flow the characteristic velocity is greater for $R_C/R_T = 0.5$ than for $R_C/R_T = 2.0$. This is due to the decrease in total mass flow at specified chamber pressure for $R_C/R_T = 0.5$ compared to the total mass flow for $R_C/R_T = 2.0$. Recall that

$$c^* = \frac{\pi R_T^2}{\dot{m}_{total}} \sum_i P_{o_{g_i}} \left(\dot{m}_i / \dot{m}_{total} \right) \tag{5.14}$$

In Figure 12, fully mixed flow at a given R_c/R_T has a characteristic velocity greater than unmixed flow for a fixed chamber pressure for the same reason even though total pressure decreases for fully mixed flow from unmixed flow.

Figure 13 presents the characteristic velocity, c*, versus cooling to total mass flow ratio. Curves are for two zone striated and two zone, fully mixed flow with R_c/R_T = 2.0. Also shown is the resulting chamber static pressure, P_g , which, for this case, was the same for the striated and mixed flow. Note that the static pressure for this engine drops off as the cooling flow is increased. Results for Figure 13 were obtained by calculations with constant reservior pressures and may not be compared on the same basis with results shown in Figure 12. Figure 12 and Figure 13 do agree, however, in indicating an improved characteristic velocity of mixed flow over striated flow. Results for curves of Figure 13 show a decreasing total mass flow with increasing cooling to total mass flow ratio which agrees with a decreasing chamber static pressure. At the same mass ratio, the total mass flow for striated flow is greater than the total mass flow for fully mixed flow.

5.7 Conclusions

The theory presented here permits an extension to an analysis of one dimensional compressible striated nozzle flow. The throat plane solution gives an approximation to two dimensional axisymmetric flow and has successfully been used for analyzing the throat plane in converging-diverging type nozzles with striated flow for throat radius of curvature to throat radius ratios down to .25. The method has been programmed using standard FORTRAN IV and is fully documented in this report. The program is capable of analyzing the nozzle throat plane and rocket motor chamber with frozen, calorically perfect; equilibrium; or frozen striated flows as well as fully mixed frozen, calorically perfect flow. By coupling the chamber and throat plane solutions, chamber and throat plane spatial distributions of velocity and other thermodynamic parameters are uniquely determined for specified geometry. Several modes of injector description and mass generation are possible with the program.

Results of the program agree favorably with experimental data and results of other computational methods, especially in predicting the mass discharge coefficient. No experimental results are presently available which permit distinct annular zone or striated flow to be compared to results obtained by this method. The dependence of discharge coefficient is primarily on the radial pressure gradient, the adiabatic or isentropic coefficient, and the amount of mass in each flow zone. Radial velocity is insignificant especially if streamlines on the throat plane have radii of curvature centered on the throat plane as proposed by the present analysis.

It can be concluded that results obtained by this method may offer an advantage over other methods by being in agreement with experimental data and by being in agreement with but requiring less computational time than finite difference methods used to obtain the same data. Further, multizone striated flow solutions may be obtained with the program in only seconds of computer time which permits rapid calculation of parametric performance calculations for candidate engine designs.

DESCRIPTION OF PROGRAM INPUT

Program input information or data consists of six types. Three of the types are required for all problems and three are optional. The optional inputs allow two kinds of problems. The required types consist of the following:

(a) Comment Cards (two cards)

(b) Interger control card (program variable name LETOUT)

(c) Namelist data which includes input parameters and option integers.

The optional input data include (1) thermochemical data for frozen, calorically perfect flow problems input under nameslist EQU or (2 and 3) modified input data to the chemical equilibrium calculations with formats identical to the formats required for the corresponding in NASA program CEC71.[1] Table I and II are schematics of input cards containing namelist and other input information or variables required for the two kinds of problems which may be considered. Input format information may be seen in Appendix D and examples of input cards for several typical problems are given in Appendix F.

Comment Cards

Two comment cards are required for each problem case. Any information may appear on these cards anywhere in the eighty column fields.

Control Card

One integer control card is required for each problem case. The integer, read by an I5 format provides control for the following:

(a) An end to all computations

(b) Printing of variables in namelist NUM

(c) Supression of printing of namelist NUM variables

(d) Stacking cases behind an initial case where thermochemical data determined by chemical equilibrium calculations does not change from the initial case.

If the control integer is zero computations are ended and the program ends. For example, if the last three cards on a set of stacked data are blank, the program ends after the next to last data set (the last case) has been used for calculations. Three blank cards should be used to provide a normal exit since each case requires two comment cards. No further printed output results if the integer control card is blank or the integer control variable, LETOUT, is zero. If the integer control is greater than one but not equal to 71, an initial case is considered but variables in namelist NUM are not printed. If thermochemical calculations occured for a preceding initial case and these data will be used again in the present case, an integer control equal to 71 is input. If this is so (LETOUT equal 71) and thermochemical calculations have been made in a previous initial case, input data in THERMO and REACTANTS sets are not to be input again. If thermochemical data are to be input under namelist EQU and not calculated, LETOUT may be equal to 71 with the only effect being that parameters in namelist NUM will not be output under the namelist write.

Namelist NUM

The third type of input data is required for all cases. This data is read under a namelist read input and includes the following:

(a) Throat plane integration data (required by Subprogram HPCG)

(b) Geometry data

(c) Chamber operation mode data or mass generation data (three modes are possible)

(d) Initial estimation data

(e) Option control data

(f) Injector data

A more complete description of the above is given in Appendix D.

Throat plane integration data are required by subroutine HPCG. These include the arrays or vectors PRMT and ERWT. PRMT are required for the first or an initial case where the throat radius is changed from a preceding initial case. ERWT are optional and if not included in namelist NUM, ERWT are determined internally. PRMT may only be changed by input with namelist NUM where the dependent variable error weights, ERWT, unless specifically designated, are optimized for throat solution iterations. Generally, ERWT are not input.

Geometry data include the throat radius (RT), the throat radius of curvature ratio (RCRT), and the chamber to throat area contraction ratio (EPSC). Unless changed, geometry data for stacked cases remain the same as initial case data.

Chamber operation mode data consist of only one of the three following sets.

(a) PG, PCTMO (PCTMzero)

(b) MDTI, AGAMGU, AMWGU, ATOGU (ATzeroGU)

(c) PRF, ESPG, CDINJ, PRO

The chamber static pressure and the fraction of total mass flow in each flow zone starting with the inner most zone in all cases comprise the first chamber mode data set listed above. The sum of all PCTMO must equal 1.0.

When specified mass flow is considered, not only is the mass flow in each flow zone required but also initial estimations of the isentropic coefficient, the molecular weight, and the total temperature in each flow zone are required. Initial estimations are replaced by equilibrium calculations or data input under namelist EQU during the first chamber solution iteration.

When specified reservior pressures are considered, the fuel and oxidizer reservior pressures are required. The O/F ratio in each zone permits the oxidizer injector areas to therefore be determined. An initial estimation of chamber static pressure is required to start the chamber solution and the throat solution iterations. The initial estimate of the chamber static pressure is determined by a fraction of the fuel reservior pressure in zone one. This estimate, ESPG, is optional in namelist NUM input but ESPG = .7 is set before data is read under namelist NUM. ESPG must be less than one and should be based on experience with rocket motor calculations to avoid extensive iteration calculations in the chamber and throat solutions. Suitable values of ESPG have been determined to be .9 for gaseous injection and .5 for liquid injection to speed convergence some.

Option control data determine the type of problem solved and must be considered carefully to assure proper program operation. Integer variables as opposed to logical variables have been used to give more flexibility. Six integer variables are included in this category and are as follows:

(a) MODECH

(b) NCEC71

(c) NAORW

(d) MGORLI

(e) MTOTMX

(f) NSPPG

MODECH must be input for each initial case and need not be input again for stacked cases where it remains the same. This control allows specified mass flow for MODECH = 1, specified reservior pressures for MODECH = 2, and fixed chamber static pressure for MODECH = 3. NCEC71 has the value 0 before read by namelist NUM. If NCEC71 = 0, frozen, calorically perfect thermochemical data is read by namelist EQU. NCEC71 = 1 permits frozen, calorically perfect thermochemical data to be determined by chemical equilibrium calculations. NCEC71 = 2 permits equilibrium data to be determined and shifting equilibrium calculations are used for the throat solution. NCEC71 = 3 permits frozen flow data to be determined and used for the throat solution. The program does not presently allow recalculation of thermochemical data during each chamber static pressure iteration loop for MODECH = 1 or 2. Modification of subprograms REACT and EQULIB would be necessary to reconsider reactants data and solution calculation time would be increased considerably. When the chamber static pressure is fixed, reactants input data cards used for thermochemical data calculations need only be used once by the thermochemical calculation subroutines since mass flow iterations do not affect thermochemical calculations. If thermochemical frozen, calorically perfect data is input under namelist EQU, the data remains unchanged for iterations for the case. NAORW = 2 before being read by namelist NUM and remains unchanged for stacked cases unless read by a namelist NUM read.

NAORW = 1 is used when it is desired to specify fixed injector areas and NAORW = 2 is used to specify fixed injector velocities. MGORLI = 2 before being read by namelist NUM. If MGORLI = 1 is input, the gas constants of gaseous fuel and oxidizer reactants in each zone must be read in an initial case by namelist NUM. This option is useful for gaseous injection where reactant molecular weights are known. If MGORLI = 2 is input or if MGORLI does not appear in an initial case namelist NUM read, fuel and oxidizer reactants densities in each zone must be read in an initial case namelist NUM read. MTOTMX = 0 provides that the burned gas in the chamber is not to be fully mixed for the throat solution calculations. MTOTMX = 1 provides that fully mixed flow of the chamber burned gases by used for the throat solution calculations. For MTOTMX = 1, one flow zone is used for throat solutions and the options NCEC71 = 0 or 1 may be considered. No fully mixed solutions may be considered for NCEC71 = 2 or 3. If only one injector zone is considered for calculations, MTOTMX = 0 must be used. NSPPG = 0 before being read by namelist NUM. If NSPPG = 0, special chamber routine SPPG is not used. If NSPPG = 1, routine SPPG will be used. NSPPG = 1 may only be used for single zone flow of low temperature nonreacting gases. For striated flow analysis of rocket motors or for single zone reacting gas flows, routine SPPG may not be used. Routine SPPG provides a method for cold flow, single zone model analysis found in the laboratory.

Injector data includes the following:

(a) Number of flow zones; NCHA1 and NCHA2

(b) Oxidizer to fuel mass ratios; OXFULE

(c) Injector reactants temperatures; TIFU and TIOX

(d) Injector area data; AITAC, AIPCT, and AIPCTF

(e) Injector velocities; WIFU and WIOX

(f) Injected reactants densities; RHOIFU and RHOIOX

(g) Injected gaseous reactants gas constants; RIFU and RIOX

Either injector area data or injector velocities and either densities or gas constants are input by namelist NUM. Other injector data required for initial cases and these also need not be changed for stacked cases unless required.

NCHA1 is required for an initial case read and must be greater than zero. NCHA2 = 0 before being read by namelist NUM initially and the sum of NCHA1 and NCHA2 cannot exceed twenty. Two regions of flow zones designated by NCHA1 and NCHA2 have been used to eliminate or change outer flow zones in stacked cases with few changes or to enable the user to modify the throat integration technique in routine HPCG for future adaptations.

Oxidizer to fuel mass ratio is required with all cases and indices accounting starts for the inner most flow zone. It may be necessary to avoid OXFULE = 0.0 or other very small fractions for a flow zone when using chemical equilibrium calculations due to the lower temperature limit on combustion products allowed by modified program CEC71.

Injector fuel temperature, TIFU, and oxidizer temperature, TIOX, must be input in each initial data case with namelist NUM. If OXFULE = 0.0 for any zone, special handling is not required; however, if oxidizer only is injected into a flow zone, the injected material must be designated a fuel and OXFULE is correspondingly set equal to zero for that zone.

If NAORW = 1, AITAC, AIPCT, and AIPCTF are required as initial case data by namelist NUM. AITAC is the ratio of total injector areas to the chamber area. AIPCT are the ratios of fuel plus oxidizer areas for each flow zone to the total injector area and AIPCTF are the ratios of fuel injector area to the fuel plus oxidizer injector area in each flow zone.[a] If AITAC is not input or is zero, AIPCT then must be the injector areas for each zone. Thus, injector areas as opposed to area ratios may be input directly.

As an alternative to specified injector areas, fuel injector velocities, WIFU, and oxidizer injector velocities, WIOX, may be input. This option is used when NAORW = 2. NAORW = 2 before being read by namelist NUM for the first case being considered.

Injector reactant densities, RHOIFU for the fuels, and RHOIOX for the oxidizers, are required for MGORLI = 2. RHOIFU (1) and RHOIOX (1) correspond to the fuel and oxidizer densities in zone one, etc. MGORLI = 2 before the first case data are read by namelist NUM.

As an alternative to injector densities, fuel gas constants, RIFU, and oxidizer gas constants, RIOX, may be input for gaseous injection. The option is exercised by setting MGORLI = 1 in namelist NUM.

Refer to Appendix D for more information on data read by namelist NUM and to Appendix F for examples.

[a]If MODECH = 2 an exception to injector area input must be made. AITAC must be the ratio of fuel injector areas to the total chamber area and AIPCT is the ratio of fuel injector area for the ith zone to the total fuel injector area. For MODECH = 2, AIPCTF is not required input but is internally set equal to 1.0.

Namelist EQU

If NCEC71 = 0, frozen, calorically perfect thermochemical data is used for chamber and throat calculations. The data required for each flow zone are the total temperature, TOG, the isentropic coefficient, GAMG, the burned gas molecular weight, MWG, and an integer variable NCHANG. Each case requires one set of data read by namelist EQU when NCEC71 = 0 and the data set follows variables read by namelist NUM. Each initial case requires all thermochemical variables described above and NCHANG must be greater than zero. For stacked cases where thermochemical data does not change from the previous case, it is necessary only to input NCHANG = 0. Table I is a schematic of the input and namelist variables required for all problems when NCEC71 = 0. Operation of the program with this input consititutes the first kind of option mentioned at the beginning of this section.

THERMO Data

When thermochemical data are to be determined by chemical equilibrium calculations, NCEC71 = 1, 2, or 3, two types of data are required. The first type of chemical data required for this kind of problem as indicated by Table II, are THERMO Data. This consists of a THERMO code card, temperature range data, and a library of coefficients for calculating thermodynamic parameters of chemical species. The data used by this program is identical to that used by NASA SP-273.[1] The data may be read by card, tape, or disk and is written on unit 4 for use during a run. A permanent tape or disk containing the data may be made during a run (requiring a THERMO code card) or prior to a run (without THERMO code card) by proper job control cards. Options and changes to the THERMO Data of NASA SP-273[1] are possible with this program also. The authors have found it convenient to permanently store THERMO data on a disk and use the disk with this program or with NASA program CEC71.

REACTANTS Data

If NCEC71 = 1, 2, or 3, complete sets of REACTANTS Data must be input for each initial case. There must be as many REACTANTS data sets as there are flow zones; however, for a stacked case after an initial case where the thermochemical data will not change, the REACTANTS data sets need not be input. If cases are to be stacked as indicated above, the Control Card, LETOUT, must have the integer 71 starting in column 4 for each stacked case. The REACTANTS data set consists of three types of cards. The first type in this set has REACTANTS punched in columns one through nine. The second type of cards are reactant cards coded with the same format as the reactant card coding found in NASA SP-273[1] except that only O/F ratios may be used. A blank card follows the reactants cards. The third type of card in each REACTANTS set has NAMELISTS punched in columns one through nine. This card initializes chemical equilibrium calculations.

Additional information on program inputs and options may be seen in Appendixes D and F.

Units

In this program the English Technical system of units is used with the exception of thermo data and the reactant card inputs which are used by the chemical equilibrium calculations. Thermo data and reactant card inputs form an optional kind of input and do not use the English Technical system so as to be compatible with the inputs currently used by NASA program CEC71. The English Technical system is as follows:

Physical Quantity	Unit	Symbol
Length	feet	ft
Time	second	sec
Temperature	Rankine	R°
Force	pound	lbf
Pressure	pounds per square foot	psf
Mass Flow	pound seconds per foot	lbf-sec/ft

The Universal gas constant used is 1545.43 lbf-ft/lbmole-°R. Density has the units of lbf-sec^2/ft^4 and specific impulse has the units of lbf-sec/lbmass. Where required, a conversion constant of 32.174 lbmass-ft/lbf-sec^2 is used. The oxidant to fuel weight ratio is used exclusively if reactant cards are used. Symbols used to represent parameters may be seen in Appendix B and program variable names are given in Appendix C.

DESCRIPTION OF PROGRAM OUTPUT

The final output of the program is in the form of printed tables that are designed to be self-explanatory. Except where fully mixed flow is considered all output are the same.

Namelist NUM

For an initial case, parameters in namelist NUM may be printed by a namelist write. The printout option is exercised by setting LETOUT = 1. For LETOUT $\geq$ 2 namelist NUM variables are not printed.

Heading and Comments

The case number is printed for each case. The numbers are internally generated and not user controlled. Information on the input comment card are printed following the case number.

Option Data

Option data includes MODECH, MGORLI, NAORW, NCEC71, NSPPG, MTOTMX, LETOUT, and NCHA1 + NCHA2 or the number of zones designated by NINCH. Also included are the number of throat solution iteration loops required for convergence (LOOP) and R_C/R_T.

Geometry and Pressure Output

This output includes $R_{\bar{T}}$, the converged value of the wall radius at the throat as determined by the throat solution, and the chamber contraction ratio, EPSC. R_T in this output is the specified throat radius. Pressure data includes the chamber static pressure, P_g, and the centerline static pressure at the throat, P_{CL}.

Injector Parameters

Output includes the zone stream function limits, the flow zone index, P_I, $T_{I_{fu_i}}$, $T_{I_{ox_i}}$, $A_{I_{fu_i}}$, $A_{I_{ox_i}}$, $w_{I_{fu_i}}$, $w_{I_{ox_i}}$, $\rho_{I_{fu_i}}$, $\rho_{I_{ox_i}}$, $R_{I_{fu_i}}$, $R_{I_{ox_i}}$, $\dot{m}_{I_{fu_i}}$, $\dot{m}_{I_{ox_i}}$, $P_{R_{fu_i}}$ and $P_{R_{ox_i}}$.

If MODECH = 3, reservior pressures are set equal to zero and if MGORLI = 2 injector gas constants are set equal to zero. If O/F_i is zero, oxidizer parameters are zero.

Burned Gas Parameters

When MTOTMX = 0, striated flow at the thraot is considered and combustion product output parameters for the chamber included $P_{o_{g_i}}$, $T_{o_{g_i}}$, $C_{P_{g_i}}$, γ_{g_i}, T_{g_i}, A_{g_i}, w_{g_i}, ρ_{g_i}, and $\dot{m}_i$ for each flow zone.

Mixed Gas Parameters

For fully mixed flow in the chamber (MTOTMX > 0) only one set of burned gas parameters are output. These parameters are designated by the same symbols as were zone parameters in burned gas outputs. In addition, however, the chamber static pressure of the fully mixed gases and the total pressure of the burned gases in each flow zone prior to mixing are included.

Throat Plane Output

Spatial distributions of r, P, ψ, T, w, M, and γ are included in throat plane output. Values at the throat wall for the converged solution are also given in the output.

Performance Output

Vacuum specific impulse at the throat, characteristic velocity, total mass flow rate for radius of curvature effects and one dimensional flow are performance outputs. In addition, the ratios of mass flows (discharge coefficient), vacuum specific impulses, and characteristic velocities are printed. Output units are given.

Units for all outputs which appear in the other outputs are found in the UNITS section of this report.

PROGRAM MODULES

The program consists of two modules as depicted in Appendix G. The first module was developed specifically for striated nozzle flow with throat radius of curvature effects. The second module is an adaption of NASA Lewis program CEC71 documented in NASA SP-273[1]. The original version of CEC71 has been modified to be compatable with requirements of the first module.

Striated Nozzle Flow

This module consists of subprograms which read comments, control data, and namelist data. If subprogram DRIVCE is made a dummy subprogram, the striated nozzle flow module may be operated without chemical equilibrium subprograms for frozen, calorically perfect gas flows. In this form, the program would consist of a main program and fourteen subprograms: two of which, LOAD1 and TAB, may also be made dummy subprograms. The purpose of this module is to determine and write injector, chamber burned gas, and throat parameters.

Chemical Equilibrium Calculations

This module, depicted in Appendix G, is used to determine thermochemical parameters for use by the first module. Linkage between the first module and this module is by common block and conversions within subprogram LOAD1 and entry to LOAD1, LOAD2. This module consists of sixteen subprograms and a block data subprogram. General output of program CEC71 has been suppressed; however, error message output has been retained.

The chemical equilibrium calculations module reads THERMO data and REACTANTS data inputs with formats and units of the unmodified form of program CEC71. Removal of comments from data output cards in modified CEC71 in this program would cause data output to appear after the heading and comments unless a stacked case with LETOUT = 71 is being run.

The first module of this program will be described in this section whereas Reference 1 describes in detail routines of module two (modified CEC71). Limitations on inputs to module two are described in the DESCRIPTION OF PROGRAM INPUT section.

Main Program

A flow chart of the main program is given in Appendix H. Generally, the routine performs the following functions:

(1) Initializes inputs and constants.

(2) Reads namelist NUM. Writes NUM if LETOUT = 1.

(3) Calculates chamber geometry.

(4) Initializes P_g for MODECH = 1 or 2.

One dimensional equations are used to estimate initial values of P_g or $\dot{m}_i$. For fixed mass flow in zones, initial estimates of γ_{g_i}, MW_{g_i}, and $T_{o_{g_i}}$ are required. For fixed reservior pressures, the chamber pressure is initially estimated to be a fraction of the fuel reservior pressure in the first flow zone. The chamber static pressure is initially estimated if required in the main program.

(5) Calls subroutine THROAT.

The purpose of the main program is to serve as a driver for the remainder of the program.

Subroutine THROAT

The purpose of subroutine THROAT is to control the throat plane iteration technqiue, determine performance parameters, and write results. A flow chart of subroutine THROAT is given in Appendix H. This subroutine performs throat plane calculations prior to performance parameter calculations and so the two types of calculations may be discussed separately.

Initially, the number of iterations performed to satisfy choking at the throat is checked to determine whether more than seven iterations have been performed to determine the total mass flow or chamber static pressure required. If LOOP = 8 then results of the seventh loop are used to determine output parameters. If mass flows are specified or if reservoir pressures are specified, P_g is being sought in the throat solution. For these modes, MODECH = 1 or 2, alterations of P_g on the seventh loop is made by interpolation when P_g has been bracketed by previous iteration values of the chamber static pressure. If it has not been bracketed, the sixth iteration is used for output parameters. If specified reservoir pressures are considered, an additional requirement is that P_g must be less than one percent smaller than any fuel reservoir pressure. Very seldom more than four iteration loops are required for a converged throat solution.

The chamber solution is determined and then throat plane centerline static pressure estimated and altered to determine the throat radius corresponding to maximum mass flux. Intergrations performed by subroutine HPCG require boundary conditions for both dependent variables at the nozzle centerline. These are $P = P_{CL}$ and ψ = PSI01(1) = 0.0. After itegration to ψ=1.0 is complete, the corresponding wall radius is determined by subroutine AITKEN. Radii are ordered monatonically upward by subroutine UPORD. Successful determination of a minimum radius consists of two criteria which permit checks to be made as to (1) whether the radius is a minimum based on $(dr/r)_w < .05\%$ and $dP_{CL}/P_{CL} < 1\%$ and (2) whether the minimum radius determined is within .05% of R_T.

If the first of the above is satisfied but not the second, P_g or $\dot{m}_{total}$ is altered and iteration starts again by indexing LOOP. Subroutine UPORD is used to order the P_{CL} during a loop once the minimum radius is bracketed for that loop. A second order curve fit by subroutine PLYCF is made to determine P_{CL} for $(dr_w/dP_{CL}) = 0.0$. The first criteria stated above must be satisfied or the process is repeated with the three smallest radii of the loop.

With the first and second criteria stated above satisfied, remaining parameters except throat plane parameters and performance are output. Performance calculations are made as described in Performance Solution. Throat plane and performance parameters are printed as described in Throat Plane Output and Performance Output, respectively.

Subroutine CHAMBR

The purpose of subroutine CHAMBR is to determine chamber parameters for striated or fully mixed flow. A flow chart for subroutine CHAMBR is given in Appendix H. This routine performs calculations shown in Chamber Solutions and calls other subroutines which cause thermochemical data to be calculated or read.

Upon being called by subroutine THROAT, subroutine CHAMBR initializes the iteration indexes NOAI and LOOPC and begins iteration loops by initializing the injector static pressure, P_I. For MODECH = 2, subroutine PRSPFD is called to determine zone mass flows and oxidizer injector areas. Both fuel and oxidizer reservoir pressures are input and zone O/F ratios must be specified. If MODECH = 3, subroutine PGFIX is called to determine thermochemical data and to determine zone mass flows from the total mass flow. Chamber parameters T_{g_i}, A_{g_i}, and w_{g_i} are next intialized for the iterations and, for MODECH = 1 or 2, thermochemical data are determined by a direct call to subprogram EQULIB. Special routine SPPG is called to provide the chamber parameters if option parameter NSPPG = 1 is in effect. Injector parameters are determined and chamber burned gas parameters T_{g_i}, ρ_{g_i}, w_{g_i}, and A_{g_i} for iteration loop LOOPC are calculated. The primary iteration with index NOAI is used for MODECH=2 which, after a complete iteration with index LOOPC, alters the mass flow by repeated calls to PRSPFD. Index NOAI also serves to control the injector pressure iteration calculations and a convergence error message is provided when NOAI > 20. For MODECH = 2, iteration is complicated by the fact that mass flow must satisfy reservoir equations and chamber equations. Thus, for specified reservoir pressures, one equation set is added and one set of unknowns, $\dot{m}_i$, must exist to determine a chamber solution.

Convergence of the chamber solution permits dividing streamlines to be determined and corresponding radii in the chamber and total pressures to be determined. If MTOTMX > 0, fully mixed flow parameters are determined by equations given in the section Fully Mixed Flow.

Subroutine PGFIX

This subroutine is called by CHAMBR when MODECH = 3. The purpose of this routine is to call subrotine EQULIB for thermochemical parameters. It also serves to provide an initial estimate of the total and zone mass flows and to calculate zone mass flows during iterations with index LOOP.

Subroutine PRSPFD

This routine provides an initial estimate for P_I for NOAI = 1. It also provides intermediate calculations for zone mass flows and oxidizer injector areas. PRSPFD is used when MODECH = 2.

Subroutine SPPG

Subroutine SPPG is used for single zone low temperature or non-reacting gas flow chamber calculations when MODECH = 1 or 3. Its use is optional and occurs when NSPPG = 1. The iteration index LOOPSP permits simultaneous solution of chamber equations to satisfy convergence requirements. Convergence requirements are that burned gas static temperature for loop LOOPSP + 1 be within .05% of the temperature of loop LOOPSP.

Subroutine EQULIB

This routine serves three purposes. These are:

(1) Read frozen, calorically perfect thermochemical data under namelist NUM when NCEC71 = 0.

(2) Call subprogram DRIVCE (formerly the main program for CEC71) for NCEC71 = 1, 2, or 3.

(3) Store chamber thermochemical data for reuse during iterations. This is necessary when MTOTMX = 1.

Logical IF statements in subprogram EQULIB determine the order of operations so to satisfy option and iteration loop requirements.

Subroutines LOAD1 and LOAD2

Subroutine LOAD1 (LOAD2 is an ENTRY to LOAD1) contain COMMON blocks POINTS and PERF which permit data transfer. These two blocks appear in NASA Lewis program CEC71. In addition, COMMON blocks ZONE, ONE, and ACSTAR have been included to permit data transferral between the two calculation modules which make up this program. LOAD1 and LOAD2 apply conversion units and store thermodynamic parameters for chamber and throat calculations. Each flow zone is assigned on index, KZONE, and two dimensional arrays permit interpolation in a flow zone at the throat for an internally generated pressure schedule. Ten pressure ratios are generated for each flow zone in subprogram ROCKET when NCEC71 = 2 or 3. In addition, one dimensional sonic pressure and chamber pressure are stored. Thus, thermochemical calculations are performed for twelve pressures for each flow zone when NCEC71 = 2 or 3. When NCEC71 = 1, only data corresponding to the chamber and sonic pressure are determined; however, data corresponding to sonic pressure are not used for calculations ince NCEC71 = 1 is for frozen, calorically perfect flow.

Subroutine HPCG

This subroutine uses Hamming's modified predictor-corrector method for the solution of throat plane equations which are of the initial-value type. This subroutine is discussed and listed with operating instructions in Reference 9. HPCG uses two external subprograms, FCT and OUTP. FCT provides the right hand side of two first order equations and OUTP stores the solution vectors of HPCG. Few changes have been made to the original version of HPCG. Comment cards have been removed and statements which have been added are so indicated by alphabetic characters in card sequences replacing last character zeros. Look to the program listing of HPCG to see the changes.

Subroutine FCT

Subroutine FCT provides the right hand side of the two first order ordinary differential equations for P and ψ with r as the independent variable. In the call to FCT from HPCG three parameters are transferred; R is the independent variable, Y are the solutions P and ψ, and DERY are the first two error weights for the dependent variables which are saved in

THROAT under variable name ERWT. FCT provides a check of option NCEC71 to determine if frozen, calorically perfect flow is being considered or if frozen or equilibrium flow is being considered. Determination is made as to which flow zone integrations are being performed in and provides the corresponding thermochemical variables. For NCEC71 = 0 or 1, the index K determines the thermochemical variables to be used. For NCEC71 = 2 or 3, subroutine TAB is called to interpolate on the current value of K and Y(1). Y(1) is the current pressure, P, and K corresponds to KZONE. KZONE is generated in DRIVCE and is the index on two dimensional arrays developed in LOAD1. Equations used for dependent variables are described in the THEORY section under Throat Solution.

Subroutine OUTP

Subroutine OUTP is called by HPCG. This routine stores dependent variables with index J. If NCEC71 = 0 or 1, temperature (T(J)) Mach number (YPSAV(J,4)), and velocity (YSAVE(J,3)) are determined in OUTP. OUTP also causes PRMT(5) to be set equal to 1.0 when ψ (YSAVE(J,2)) becomes $\geq$ 1.0. PRMT(5) $\neq$ 0.0 causes return to subroutine THROAT from subroutine HPCG.

Subroutine TAB

TAB is an interpolation routine used by subroutine FCT for determination of thermochemical data when NCEC71 = 2 or 3.

Subroutine AITKEN

This subroutine is called by subroutine THROAT to determine the throat wall radius after return to THROAT from HPCG. It also provides wall values of throat plane parameters which are determined in the performance section of subroutine THROAT.

Subroutine UPORD

This subroutine is called by subroutine THROAT to order a pair of real array variables. The parameter list for this subroutine contains two array variables and two integers. The first integer gives the number of pairs and the second integer, MORDER, determines which array will be ordered up. MORDER $\neq$ 0 orders the first array upward with corresponding variables in the second carried along. MORDER = 0 orders the second array. MORDER = 1 for this program.

Subroutine PLYCF

Subroutine PLYCF is a least squares polynomial curve fit routine called by subroutine THROAT to predict P_{CL} for $r_{w_{min}}$.

Subroutine DRIVCE is the modified main program of NASA Lewis program CEC71. Subroutines REACT, SEARCH, HCALC, SAVE, EQLBRM, CPHS, MATRIX, GAUSS, OUT1, VARFMT, EFMT, THERMP, ROCKET, RKTOUT, FROZEN, and BLOCK DATA are subprograms used by the program CEC71 and are described in NASA SP=273[1]. Differences between these routines and the routines of CEC71 may be seen by card sequence numbers. A / symbol after the sequence name indicates a comment card has been made and an alphabetic character and a repeated sequence number after the sequence name indicates a statement has been added. Changes to CEC71 occur in DRIVCE (CEC71 program Main), ROCKET, and RKTOUT. Subroutines SHCK and DETON of CEC71 have been removed for this modification.

APPENDIX A

PHILOSOPHY

The philosophy used in developing this program has been to define where possible, subprograms or variables with names which are appropriate and representative of the purpose they serve or the commonly used symbol they represent, respectively. Use of the same variable name to represent different variables has been avoided. Subprograms which have been written and documented by others which serve as utility subprograms may have changes; however, changes which have been necessary have been made so as to be easily detectable. Individual card sequence numbers expose the changes made. The program has been written to be as machine independent as possible. The FORTRAN IV language (FORTRAN G level 20) has been used and the program is constructed so that user modifications are possible, particularly where a change in the units for input and output is desired. Transferral of variables between subprograms is by common block where possible and algorithms have been used which are logical and easy to follow. Computation time for typical cases are small (approximately one minute on the IBM 360/65). The main input data set is by namelist to simplify checking input cards and to simplify stacked case input data sets.

The program is intended to provide a capability for rapid analysis with detailed chamber operating conditions and spatial distributions of thermodynamic flow properties on the nozzle throat plane. An important aspect of the program is that it permits throat plane caluclations with a variable adiabatic or isentropic coefficient and is not limited to a constant stagnation pressure in the chamber. The computer program for calculation of chemical equilibrium compositions as found in NASA SP-273[1] has been incorporated as subprograms. Changes to this program, CEC71, may easily be seen in the program listing and compaired to the original version of the program by the sequence code found in the last eight columns of the computer card. The SHCK and DETON subprograms of CEC71 were removed and references to them have been made comment cards. Input data to the modified form of CEC71 in this program are simplified somewhat but, where required, have the same form and units used by program CEC71. Standard output of the modified CEC71 calculations have been suppressed by commenting output in the manner described above; however, error messages from this program have not been suppressed.

APPENDIX B

SYMBOLS

A	area, ft^2
A_c	chamber area, ft^2
A_T	throat area, ft^2
C	$P/(\rho)^\gamma$, polytropic equation of state constant (isentropic)
C_D	discharge coefficient
C_p	constant pressure specific heat, ft^2/sec^2-°R
c^*	characteristic velocity, ft/sec
g_c	32.174 lbmass-ft/lbf-sec^2
H_o	total enthalpy, ft^2/sec^2
h	enthalpy, ft^2/sec^2
$I_{sp_{vac}}$	throat vacuum specific impulse, lbf-sec/lbmass
M	Mach number
MW	molecular weight, lbmass/lbmole
$\dot{m}$	mass flow rate, lbf-sec/ft
O/F	oxidant to fuel mass ratio
P_o	stagnation pressure, lbf/ft^2
P	pressure, lbf/ft^2
R	gas constant, ft^2/sec^2-°R
R_c	throat wall radius of curvature, ft
R_T	throat radius, ft.
R_v	flow radius of curvature at throat, ft
$\mathcal{R}$	universal gas constant, 1545.43 lbf-ft/lbmole-°R
r	radius; radial coordinate, ft
T	temperature, °R

T_o	stagnation temperature, °R
u	radial velocity component, ft/sec
v	tangential velocity component, ft/sec
w	axial velocity component, ft/sec
z	axial coordinate, ft
γ	isentropic exponent or adiabatic coefficient
π	3.14159265
ρ	density, $lbf\text{-}sec^2/ft^4$
ψ	stream function = $\int_0^{\dot{m}(r)} [d\dot{m}/\dot{m}_{total}]$

Subscripts:

CL	centerline value on or near the throat plane
fu	fuel
g	combustion product or burned gas
I	injector
min	minimum value
mix	fully mixed
ox	oxidizer
R	reservoir
total	total or whole amount
w	wall value on or near the throat plane
1-D	one dimensional

Superscripts:

'	modified
*	critical throat condition

Indices:

i	striation zone (inner most zone is 1)
j	local value on the throat plane

APPENDIX C

COMMON VARIABLES

All common variables are given in this appendix except those developed in the original unmodified program CEC71, which is documented in NASA SP-273.

Variable	Dimension	Common Label	Description and Comments
AC	1	THR	A_C .
AG	20	THR	A_g .
AIFU	20	THR	$A_{I_{fu_i}}$.
AIOX	20	THR	$A_{I_{ox_i}}$.
AIPCTF	20	THR	Fraction of injector fuel area assigned to each zone.[a]
AMAC	20,13	LOTAB	Mach number array in each zone corresponding to APRES values.
APRES	13	LOTAB	P_g/P array used for interpolation at the throat plane for frozen or equilibrium flow.
ARHO	20,13	LOTAB	Density array in each zone corresponding to APRES values.
ASON	20,13	LOTAB	Sonic velocity array in each zone corresponding to APRES values.
AUX	16,4	blank	Auxiliary storage array (See HPCG subprogram description).
AVEL	20,13	LOTAB	Velocity array in each zone corresponding to APRES values.
C	1	CALC	$P_g/\{\rho_{g_i}\}^{\gamma_{g_i}}$, used for frozen, calorically perfect flow only.
CDINJ	20	THR	Injector discharge coefficients for MODECH = 2.

[a]For MODECH = 2, AIPCT becomes the ratio of fuel injector areas to chamber area and AIPCTF is not required.

CLP	10	blank	P_{CL}.
CPG	20	TWO	$C_{p_{g_i}}$.
CST	20	LOAD	c^*_i from chemical equilibrium calculations.
CSTAR	13	ACSTAR	c* corresponding to P_g for each zone
DERY	4	blank	Input error weights, dependent variable derivatives (see FCT and HPCG subprograms descriptions).
DIACH	1	THR	Chamber diameter.
EPSC	1	ONE	Chamber contraction ratio.
ERWT	4	blank	Error weights for dependent variables in subprogram HPCG.
GAM	1	CALC	Local value of γ on the throat plane.
GAMG	20	TWO	γ_{g_i}.
GAMR	20,13	LOTAB	Isentropic coefficient array in each zone corresponding APRES values
GAMSAV	200	CALC	γ array on the throat plane.
HO (Hzero)	1	CALC	Local value of total enthalpy, $H_o = C_{p_{g_i}} T_{o_{g_i}}$.
ICASE	1	ONE	Solution case index.
IHLF	1	ONE	Step size bisection parameters used by subroutine HPCG.
J	1	TWO	Step index for local throat plane parameters
K	1	CALC	Index parameter for zone designation.
KZONE	1	ZONE	Zone index.

LETOUT	1	ZONE	Integer control constant. See input data for LETOUT options.
LOOP	1	blank	Index counter designating the number of throat solution iteration loops.
MDTI	20	THR	$\dot{m}_i$.
MDTIFU	20	THR	$\dot{m}_{I_{fu_i}}$.
MDTIOX	20	THR	$\dot{m}_{I_{ox_i}}$.
MGORLI	1	ONE	Input option parameter indicating injector gas constant input (=1) or injector density input (=2). MGORLI = 2 initially.
MODECH	1	blank	Input option parameter indicating the mass injector scheme. See input data for MODECH options.
MORDER	1	blank	= 1, describes the method for ordering variables in subroutine UPORD.
MTOTMX	1	THR	Input option parameter. MTOTMX = 0 for striated flow, MTOTMX=1 for fully mixed flow. MTOTMX=0 initially.
MWG	20	THR	MW_{g_i}.
MO (Mzero)	1	TWO	$\Sigma\dot{m}_i$, total mass flow.
MOCH MzeroCH)	7	blank	$\Sigma\dot{m}_i$ corresponding to LOOP.
NAORW	1	THR	Input option parameter indicating whether injector areas are fixed (=1) or injector velocities are fixed (=2). Initially NAORW = 2.
NCEC71	1	ONE	Input option indicating the degree of chemical equilibrium. See Input data for NCEC71 options.

NCHA1	1	TWO	The number of flow zones in an arbitrary chamber region ($1 \leq NCHA1 \leq 10$ NCHA1 is an input option.
NCHA2	1	TWO	The number of flow zones in an arbitrary chamber region ($0 \leq NCHA2 \leq 10$ NCHA2 is an input option.
NDIM	1	ONE	= 2. See HPCG subprogram description
NSPPG	1	ONE	Special input option for single zone nonreacting flow calculations. Initially NSPPG = 0. See input list.
OXFUL	20	ZONE	O/F_i for chemical equilibrium calculations.
OXFULE	20	THR	O/F_i.
PCTMO (PCTMzero)	20	THR	Fraction of total flow in the ith flow zone.
PG	1	THR	P_g.
PGCH	7	blank	P_g corresponding to LOOP.
PGPSIA	1	ZONE	P_g internally converted to psia for chemical equilibrium calculations.
PI	20	THR	P_I
PMIX	1	MIXED	$P_{g_{mix}}$
PRF	20	THR	$P_{R_{fu_i}}$.
PRO	20	THR	$P_{R_{ox_i}}$.
PRMT	6	blank	Input and output array for HPCG (see input desription).
PSIO1	10	TWO	ψ_i in the first flow region.
PSIO2	10	TWO	ψ_i in the second flow region.

PSI12	1	TWO	ψ corresponding to the streamline between the first and second flow regions.
PO (Pzero)	20	MIXED	Total pressure in each zone before full mixing occurs.
POCL (PzeroCL)	7	blank	$P_{o_{g_i}}$ corresponding to LOOP.
POG (PzeroG)	20	TWO	$P_{o_{g_i}}$.
RAD	20	THR	Radius corresponding ψ_i in the chamber.
RCRT	1	ONE	R_c/R_T.
RCWALL	1	TWO	R_c.
RGAS	20	THR	R_{g_i}.
RHOG	20	THR	ρ_{g_i}.
RHOIFU	20	THR	$\rho_{I_{fu_i}}$.
RHOIOX	20	THR	$\rho_{I_{ox_i}}$.
RIFU	20	THR	$R_{I_{fu_i}}$.
RIOX	20	THR	$R_{I_{ox_i}}$.
RLIM	1	blank	not used.
RMIN	7	blank	r_w corresponding to LOOP.
RMINSQ	7	blank	r^2_w corresponding to LOOP.
RT	1	TWO	R_T.
RW	10	blank	Wall radius vector corresponding to LOOP.

T	200	ONE	T on the throat plane.
TEMPE	20,13	LOTAB	Temperature array in each zone corresponding to APRES values.
TG	20	THR	T_{g_i} .
TIFU	20	THR	$T_{I_{fu_i}}$.
TIOX	20	THR	$T_{I_{ox_i}}$.
TWOPI	1	TWO	2π
TOG (TzeroG)	20	THR	$T_{o_{g_i}}$.
UNIVR	1	THR	$\mathcal{R}$
URMNSQ	7	blank	$1/r^2_w$ corresponding to LOOP.
VAC1DI	20	LOAD	$I'_{sp_{vac}}$.
VACSPI	20,13	LOTAB	Vacuum specific impulse array in each zone corresponding to APRES values.
WG	20	THR	w_{g_i} .
WIFU	20	THR	$w_{I_{fu_i}}$.
WIOX	20	THR	$w_{I_{ox_i}}$.
WMR	20,13	LOTAB	Molecular weight array in each zone corresponding to APRES values.
XSAVE	200	TWO	r on the throat plane. The index is J.
Y	4	blank	Y(1)=P and Y(2)=ψ on the throat plane corresponding to XSAVE. Y(3) and Y(4) are not used.

YPSAV	200,4	TWO	YPSAV(J,1) is the pressure derivative array on the throat plane. YPSAV (J,2) is the derivative of ψ. YPSAV(J,3) is $T_{o_{g_i}}/T$ on the throat plane. YPSAV(J,4) is the Mach number array on the throat plane.
YSAVE	200,4	TWO	YSAVE(J,1) = p on the throat plane. YSAVE(J,2) = ψ on the throat plane. YSAVE(J,3) = w on the throat plane. YSAVE(J,4) is dA corresponding to YSAVE(J,1).
ZCH	1	THR	not used.

DIMENSIONED VARIABLES

Variable	Dimension	Routines where used	Description and Comments
A	11	PLYCF	polynomial coefficients.
ACO	11	THROAT	polynomial coefficients.
AGAMGU	20	MAIN	initial γ_{g_i} estimates.
AIPCT	20	MAIN	fraction of total injector area for zone i.[a]
AMACH	200	THROAT	Mach number on the throat plane.
AMWGU	20	MAIN	initial MW_{g_i} estimates.
ATOGU (ATzeroGU)	20	MAIN	initial $T_{o_{g_i}}$ estimates.
AUX	16,2	HPCG	Auxiliary storage array (see HPCG subprogram description).
CLP	10	UPORD	P_{CL}.
DERY	2	FCT, OUTP, HPCG	input error weights, dependent variable derivatives (see FCT subprogram description).
DRDPSV	10	THROAT	dr_w/dP_{CL}.
DUM1	3	THROAT	independent dummy variable for AITKEN interpolation.
DUM2	3	THROAT	dependent dummy variable for AITKEN interpolation.
PII	20	CHAMBR	stored P_I array during CHAMBR iteration.
PRMT	6	OUTP, HPCG	input and output array for HPCG (See input description).
PSI	200	THROAT	ψ on throat plane.

[a]For MODECH = 2, AIPCT is the ratio of fuel injector area to total fuel injector area for the ith zone.

RADT	20	THROAT	radius of stream tube shell boundary at the throatplane.
RW	10	UPORD	r_w.
SAVGAM	20	EQULIB	γ_{g_i}, values stored to prevent destruction in fully mixed flow cases.
SAVMWG	20	EQULIB	MW_{g_i} values stored to prevent destruction in mixed flow cases.
SAVTOG (SAVTzeroG)	20	EQULIB	$T_{o_{g_i}}$ values stored to prevent destruction in fully mixed flow cases.
SUMAGG	20	CHAMBR	ΣA_{g_i} .
TGA	15	SPPG	T_{g_1} .
TGA	20,10	CHAMBR	T_{g_i} .
WHAT1	20	MAIN	first input comment card required for each case (20A4).
WHAT2	20	MAIN	second input comment card required for each case (20A4).
X	variable	AITKEN, PLYCF	general independent varialbe.
Y	2	FCT,OUTP, HPCG	dependent variable in HPCG (see FCT subroutine description).
Y	variable	AITKEN, PLYCF	general dependent variable.

APPENDIX D

INPUTS AND OPTIONS

Comment Cards (2):

Order	Contents	Format	Card Columns
First[a]	Any information pertinent to the case being run.	20A4	1 to 80

Control Card (Variable name LETOUT):

Order	Contents	Format	Card Columns
Second[b]	Integer Control (LETOUT in Common label ZONE)	I5	1 to 5

(1) If LETOUT = 0, program computations stop and program output begins.

(2) If LETOUT = 1, any initial case input follows and variables in namelist NUM will be output by namelist write.

(3) If LETOUT > 1, but ≠ 71, any initial case[c] or a stacked case[d] using input thermochemical data follows. Variables in namelist NUM will not be output.

(4) Stacked cases which require thermochemical calculation results be used for solutions as opposed to input thermochemical data must have LETOUT = 71 in columns 4 to 5. This stacked case option is used as when the chamber static pressure, flow zone O/F ratios, and reactants data for each flow zone will not change for the stacked cases. Nozzle geometry and/or propellant distributions may be changed when LETOUT=71 for stacked cases which meet requirements stated above.

[a]Two comment cards are required for all cases.

[b]The control card must be preceded by two comment cards.

[c]More than one initial case may be run at one time. Any initial case is a case where a complete set of thermochemical input data is required.

[d]Stacked cases refer to cases where thermochemical data need not be calculated (NCEC71 = 1, 2, or 3) or read (NCEC71 = 0).

Namelist NUM Variables:

Variable	Dimension	Type	Common Label	Value before read by first case
PRMT	6	R	blank	undefined
ERWT	4	R	blank	0.
PG	1	R	THR	0.
MDTI	20	R	THR	0.
RT	1	R	TWO	undefined
RCRT	1	R	ONE	undefined
EPSC	1	R	ONE	undefined
OXFULE	20	R	THR	0.
PRF;PRO	20;20	R	THR	0.
ESPG	1	R	-	.7
AITAC	1	R	-	0.
AIPCT	20	R	-	0.
AIPCTF	20	R	THR	0.
RHOIFU	20	R	THR	0.
RIFU	20	R	THR	0.
TIFU	20	R	THR	0.
WIFU	20	R	THR	0.
RHOIOX	20	R	THR	0.
RIOX	20	R	THR	0.
TIOX	20	R	THR	0.
WIOX	20	R	THR	0.
PCTMO	20	R	THR	0.
AGAMGU	20	R	-	0.
AMWGU	20	R	-	0.
ATOGU	20	R	-	0.
CDINJ	20	R	THR	.7

Namelist NUM Variables (cont'd.)

Variable	Dimension	Type	Common label	Value before read by first case
MTOTMX	1	I	THR	0
MODECH	1	I	blank	undefined
NAORW	1	I	THR	2
MGORLI	1	I	ONE	2
NCHA1	1	I	TWO	undefined
NCHA2	1	I	TWO	2
NSPPG	1	I	ONE	0
NCEC71	1	I	ONE	0

Variable in NUM	Option exercised
MTOTMX = 0	Striated flow solution.
MTOTMX = 1	Fully mixed flow solution (only for NCEC71 = 0 or 1)
MODECH = 1	Input zone mass flows.
MODECH = 2	Input zone fuel and oxidizer reservoir pressures.
MODECH = 3	Input chamber static pressure.
NAORW = 1	Input injector areas.
NAORW = 2	Input injector velocities (only for MODECH=1 or 3).
MGORLI = 1	Input injector gas constants.
MGORLI = 2	Input injector densities.
NSPPG = 0	Complete chamber solution.
NSPPG = 1	Special chamber solution (only for NCHA1=1, NCHA2=0)
NCEC71 = 0	Input thermochemical data by namelist EQU.
NCEC71 = 1	Calculates thermochemical data for frozen, calorical perfect flow.
NCEC71 = 2	Calculates thermochemical data for equilibrium flow.
NCEC71 = 3	Calculates thermochemical data for frozen flow.

Namelist EQU Variables (used only when NCEC71 = 0):

Variable	Dimension	Type	Common label	Value before read by first-initial case
TOG[a]	20	R	THR	undefined
GAMG	20	R	TWO	undefined
MWG	20	R	THR	undefined
NCHANG	1	I	-	0

NCHANG must be read for each case under namelist EQU when NCEC71=0. If an initial case is being considered NCHANG must be greater than zero and thermochemical data variables for each flow zone starting from the innermost flow zone must be input under namelist EQU. For stacked cases when NCEC71=0 and new thermochemical data are not to change from the preceding initial case, only NCHANG=0 need be read under namelist EQU. For stacked cases when NCEC71=0, control card LETOUT may be input with any integer greater than one in columns 1 to 5 and parameters in namelist NUM will not be output by a namelist write.[b]

[a]TzeroG

[b]Namelist NUM print out for cases after the first case will not be completely applicable to the cases since variable parameters in NUM from the previous case will be seen.

Thermo Data:

Card Order	Contents	Format	Card Column
1	THERMO	3A4	1 to 6
2	Temperature range for 2 sets of coefficients: lowest, common, and highest T, °K	3F10.3	1 to 30
3	Species name, Date	3A4, 2A3	1 to 12, 19 to 24
	Atomic symbols and formula	4(A2,F3.0)	25 to 44
	Phase of species (S,L, or G for solid, liquid, or gas respectively)	A1	45
	Temperature range	2F10.3	46 to 65
	Integer 1	I15	80
4	Coefficients a_i (i=1 to 5) (for upper temperature interval)	5(E15.8)	1 to 75
	Integer 2	I5	80
5	Coefficients a_i (i=6 and 7) (for upper temperature interval) and a_i (i=1 to 3) (for lower temperature interval)	5(E15.8)	1 to 75
	Integer 3	I5	80
6	Coefficients a_i (i=4 to 7) (for lower temperature interval)	4(E15.8)	1 to 60
	Integer 4	I20	80

(a) repeat cards 3 to 6 for each different species

(Final Card)	END	3A4	1 to 3

Reactants Data:

Order	Contents	Format	Card Columns
1	REACTANTS	3A4	1 to 9
2	One card for each reactant species (maximum 15). Each card contains:		
	(1) Atomic symbols and formula numbers (maximum of 5 sets)[a]	5(A2,F7.5)	1 to 45
	(2) Relative weight[b]	F7.5	46 to 52
	(3) Enthalpy[a], cal/mole	F9.5	54 to 62
	(4) State: ,L, or G for solid, liquid, or gas, respectively	A1	63
	(5) Temperature associated with enthalpy in (3), $^{\circ}K$	F8.5	64 to 71
	(6) F if fuel or 0 if oxidant	A1	72
	(7) Density in g/cm^3 (optional)	F8.5	73 to 80
3	blank		
4	NAMELISTS	3A4	1 to 9
(c)	repeat cards 1 to 4 for each flow zone		

[a] Program will calculate the enthalpy (3) for species in THERMO data at tne temperature (5) if zeros are punched in card columns 37 and 38.

[b] Relative weight of fuel in total fuel or oxidant in total oxidant.

APPENDIX E

PROGRAM LISTING

```
C
C     ALL INPUT DATA HAVING UNITS EXCEPT INPUTS IN THE REACTANTS AND
C     THERMO DATA CARDS TO THE CHEMICAL EQULIBRIUM SUBPROGRAMS (FORMERLY
C     CEC71) MUST USE AN ENGLISH TECHNICAL SYSTEM OF UNITS.
C     THE ENGLISH TECHNICAL SYSTEM OF UNITS REQUIRED EMPLOYS
C     ONLY FOUR BASIC UNITS.  THESE UNITS ARE POUNDS FORCE, SECONDS,
C     FEET, AND DEGREES RANKINE.
C     ALL OUTPUT DATA HAVING UNITS ARE IN THE ENGLISH TECHNICAL SYSTEM
C     OF UNITS DESCRIBED ABOVE UNLESS INDICATED.
C
      REAL MDTI,MO,MOCH,MDTIFU,MDTIOX,MWG
      DIMENSION WHAT1(20),WHAT2(20),AGAMGU(20),AMWGU(20),ATOGU(20)
     1,AIPCT(20)
      COMMON    PRMT(6),DERY(4),AUX(16,4),Y(4),RW(10)
     1          ,POCL(7),ERWT(4),RMIN(7),MODECH,RMINSQ(7)
     2          ,PGCH(7),CLP(10),RLIM,MORDER,LOOP,MOCH(7),URMNSQ(7)
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSI01(10),PSI02(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRU(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      COMMON/ZONE/KZONE,PGPSIA,OXFUL(20),LETOUT
      NAMELIST/NUM/PRMT,ERWT,PG,MDTI,RT,RCRT,EPSC,OXFULE,PRF,PRO,ESPG,
     1 AITAC,AIPCT,AIPCTF,RHOIFU,RIFU,TIFU,WIFU,RHOIOX,RIOX,TIOX,WIOX,
     2 PCTMO,    CDINJ,    AGAMGU,AMWGU,ATOGU,
     3 MTOTMX,MODECH,NAORW,MGORLI,NCHA1,NCHA2,NSPPG,NCEC71
C
C     LETOUT=0 STOPS CALCULATIONS.
C     LETOUT=1 WRITES VARIABLES IN NAMELIST NUM.
C     LETOUT=71 DOES NOT REQUIRE NEW INPUTS FOR DRIVCE WHICH IS USEFUL
C     WITH STACKED CASES AS WHERE CHAMBER STATIC PRESSURE REMAINS THE
C     SAME AS THE PREVIOUS CASE.  LETOUT.GT.1 ALLOWS STACKED CASES AND
C     VARIABLES IN NUM ARE NOT PRINTED.
C
C     NCHA1 PLUS NCHA2 ARE THE NUMBER OF FLOW ZONES CONSIDERED.
C
C     MODECH=1 CORRESPONDS TO FIXED MASS INJECTION.
C     MODECH=2 CORRESPONDS TO SPECIFIED RESERVOIR PRESSURES.
C     MODECH=3 CORRESPONDS TO FIXED CHAMBER STATIC PRESSURE.
C
C     MGORLI=1 IS FOR GAS INJECTION.  MGORLI=2 IS FOR LIQUID OR GAS
C     INJECTION.  INJECTOR GAS CONSTANTS ARE REQUIRED WHEN MGORLI=1.
C     FUEL AND OXIDIZER DENSITIES ARE REQUIRED WHEN MGORLI=2.
C
C     NAORW=1 IS FOR FIXED AREAS.  NAORW=2 IS FOR FIXED VELOCITIES.
C
```

```
      TWOPI=6.2831853
C
C     NCEC71=1,2,OR 3 INITIATES MODIFIED EQULIBRIUM PROGRAM CEC71.
C     NCEC71=0 USES A SPECIAL INPUT DATA SET AND ASSUMES FROZEN,
C     CALORICALLY PERFECT FLOW.
C     FOR NCEC71=0, THE SPECIAL INPUT DATA SET UNDER NAMELIST EQU IS
C     REQUIRED AND CONSISTS OF ONE SET FOR EACH ZONE OF TOG, GAMG, MWG
C     STARTING WITH THE INNER ZONE.  IN ADDITION, AN INTEGER OPTION
C     NUMBER, NCHANG, WHICH IS GREATER THAN ZERO MUST BE READ IN
C     UNDER EQU.  HOWEVER, IF CASES ARE TO BE STACKED BEHIND THE CASE
C     DESCRIBED ABOVE AND TOG, GAMG, AND MWG DO NOT CHANGE, ONLY
C     NCHANG EQUAL ZERO NEED BE READ IN FOR THE STACKED CASES
C     WITH NCEC71=0.
C     NCEC71=0  INPUT FROZEN, CALORICALLY PERFECT DATA.
C     NCEC71=1  CEC71 CALCULATES FROZEN, CALORICALLY PERFECT DATA.
C     NCEC71=2  CEC71 CALCULATES EQULIBRIUM DATA.
C     NCEC71=3  CEC71 CALCULATES FROZEN DATA.
C
C
C
C
C
      UNIVR=1545.43
      NDIM=2
      ICASE=0
      DO 111 I=1,20
      PRF(I)=0.
      PRO(I)=0.
      OXFULE(I)=0.
      MDTI(I)=0.
      MDTIFU(I)=0.
      MDTIOX(I)=0.
      AIPCT(I)=0.
      AIPCTF(I)=0.
      RHOIFU(I)=0.
      RHOIOX(I)=0.
      RIFU(I)=0.
      RIOX(I)=0.
      TIFU(I)=0.
      TIOX(I)=0.
      WIFU(I)=0.
      WIOX(I)=0.
      AGAMGU(I)=0.
      AMWGU(I)=0.
      ATOGU(I)=0.
      CDINJ(I)=.7
  111 PCTMO(I)=0.
      ERWT(1)=0.
      ERWT(2)=0.
      ERWT(3)=0.
```

```
      ERWT(4)=0.
      PG=0.
      ESPG=.7
      AITAC=0.
      NCHA2=0
      NSPPG=0
      NAORW=2
      MGORLI=2
      NCEC71=0
      MTOTMX=0
    1 READ(5,101)WHAT1,WHAT2,LETOUT
  101 FORMAT(20A4,/,20A4,/,I5)
      IF(LETOUT.EQ.0)GOTO 999
      ICASE=ICASE+1
      READ(5,NUM)
      IF(LETOUT.GE.2) GOTO 110
      WRITE(6,666)
      WRITE(6,100)
  100 FORMAT(37H PARAMETERS IN NAMELIST NUM VARIABLES)
      WRITE(6,NUM)
  110 CONTINUE
      DIACH=SQRT(EPSC)*2.*RT
      AC=EPSC*TWOPI*RT*RT/2.
      NINCH=NCHA1+NCHA2
      IF(NAORW.EQ.2) GOTO 108
      IF(AITAC.EQ.0.) AITAC=1./AC
      DO 109 I=1,NINCH
      IF(MODECH.EQ.2) AIPCTF(I)=1.0
      AIFU(I)=AC*AITAC*AIPCT(I)*AIPCTF(I)
  109 AIOX(I)=AC*AITAC*AIPCT(I)*(1.-AIPCTF(I))
  108 CONTINUE
      RCWALL=RCRT*RT
      WRITE(6,102)ICASE,WHAT1,WHAT2
  102 FORMAT(1H1,60X,6H CASE ,I2,///,20X,20A4,/,20X,20A4,/////)
      MORDER = 1
      LOOP=0
      IF( ESPG.NE.1..AND.MODECH.EQ.2)PG= ESPG*PRF(1)
      IF(ESPG.EQ.1.) PG=.9*PRF(NINCH)
      IF(MODECH.EQ.1) GOTO 103
      GOTO 888
  103 CONTINUE
      WRITE(6,105)
  105 FORMAT(1X,4HZONE,3X,9HEST.GAMMA,6X,6HEST.MW,9X,6HEST.TO,/)
      WRITE(6,106) (I,AGAMGU(I),AMWGU(I),ATOGU(I),I=1,NINCH)
  106 FORMAT(2X,I2,1X,3E15.6)
      WRITE(6,107)
  107 FORMAT(/////)
      ASTAR=TWOPI*RT*RT/2.
      UNIV=SQRT(UNIVR*32.174)
      SUM=0.
```

```
      DO 104  I=1,NINCH
      MDTIFU(I)=MDTI(I)/(1.+OXFULE(I))
      MDTIOX(I)=MDTI(I)-MDTIFU(I)
      FOFGAM=SQRT(AGAMGU(I))*(2./(AGAMGU(I)+1.))**((AGAMGU(I)+1.)/
     1(2.*AGAMGU(I)-2.))
  104 SUM=SUM+MDTI(I)/FOFGAM*SQRT(ATOGU(I)/AMWGU(I))
      PG=UNIV*SUM/ASTAR
  888 CONTINUE
      IF(NINCH.EQ.1) MTOTMX=0
      CALL THROAT
      GOTO 1
  999 CONTINUE
      WRITE(6,666)
  666 FORMAT(1H1)
      STOP
      END
```

```
      SUBROUTINE THROAT                                                 THRO  10
      REAL MO1,MO2,MO,MDTI,MDTIFU,MDTIOX,ISPVAC,ISPV1D,MO1D,MOCH,MWG    THRO  20
      DOUBLE PRECISION ACO(11)                                          THRO  30
      DIMENSION PSI(200),AMACH(200),RADT(20),DUM1(3),DUM2(3),DRDPSV(10) THRO  40
      COMMON    PRMT(6),DERY(4),AUX(16,4),Y(4),RW(10)                   THRO  50
     1          ,POCL(7),ERWT(4),RMIN(7),MODECH,RMINSQ(7)               THRO  60
     2          ,PGCH(7),CLP(10),RLIM,MORDER,LOOP,MOCH(7),URMNSQ(7)     THRO  70
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)   THRO  80
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),THRO  90
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSI01(10),PSI02(10),TWOPI,PSI12,MO THRO 100
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),         THRO 110
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),         THRO 120
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,        THRO 130
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),  THRO 140
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX          THRO 150
     5,CDINJ(20)                                                        THRO 151
      COMMON/CALC/HO,C,GAM,GAMSAV(200),K                                THRO 160
      COMMON/LOAD/CST(20),VAC1DI(20)                                    THRO 170
      COMMON/ZONE/KZONE,PGPSIA,OXFUL(20),LETOUT                         THRO 180
      COMMON/MIXED/PMIX,PO(20)                                          THRO 190
      EQUIVALENCE  (YSAVE(201),PSI(1)),(YPSAV(601),AMACH(1))            THRO 200
      EXTERNAL FCT,OUTP                                                 THRO 210
      RTSQ=RT*RT                                                        THRO 220
      URTSQ=1./RTSQ                                                     THRO 230
      NINCH=NCHA1+NCHA2                                                 THRO 240
      IF(MTOTMX.GT.0) NCHA1=NINCH                                       THRO 250
      IF(MTOTMX.GT.0) NCHA2=0                                           THRO 260
      NCHA1P=NCHA1+1                                                    THRO 270
      NCHA2P=NCHA2+1                                                    THRO 280
      NINCHP=NINCH+1                                                    THRO 290
      NCH1P2=NCHA1+2                                                    THRO 300
      K=0                                                               THRO 310
      MBRKPG=0                                                          THRO 320
    2 LOOP=LOOP+1                                                       THRO 330
      NINCH=NCHA1+NCHA2                                                 THRO 340
      IF(MTOTMX.GT.0) NCHA1P=2                                          THRO 350
      IPLY=0                                                            THRO 360
      IF(MODECH.EQ.3.AND.LOOP.GT.7) GOTO 6                              THRO 370
      IF(MODECH.EQ.3) GOTO 21                                           THRO 380
      IF(LOOP.LE.6) GOTO 446                                            THRO 390
      PNOW=PG                                                           THRO 391
      IF(LOOP.GT.6) PG=PGCH(LOOP-1)                                     THRO 400
      IF(MODECH.EQ.2.AND.LOOP.GT.7) PG=PNOW                             THRO 401
      IF(LOOP.GT.7) WRITE(6,445)                                        THRO 410
  445 FORMAT(//,1X,80HCONVERGENCE ON PG DID NOT OCCUR IN  7 LOOPS. PARAMTHRO 420
     1ETERS DETERMINED NEVERTHELESS.,/)                                 THRO 430
      IF(LOOP.GT.6) MBRKPG=MBRKPG+1                                     THRO 440
      IF(MBRKPG.GT.1) GOTO 446                                          THRO 450
      DO 448 I=2,6                                                      THRO 460
      DO 450 II=2,6                                                     THRO 470
```

```
      IGTRT=I                                                           THRO 480
      IGTRTM=II                                                         THRO 490
      IF(II.EQ.I) GOTO 450                                              THRO 491
      IF(RMIN(I).GT.RT.AND.RMIN(II).LT.RT) GOTO 449                     THRO 500
  450 CONTINUE                                                          THRO 510
  448 CONTINUE                                                          THRO 520
      GOTO 446                                                          THRO 521
  449 PG=PGCH(IGTRT)+(RT-RMIN(IGTRT))*(PGCH(IGTRT)-PGCH(IGTRTM))/       THRO 530
     1   (RMIN(IGTRT)-RMIN(IGTRTM))                                     THRO 540
      GOTO 447                                                          THRO 550
  446 CONTINUE                                                          THRO 560
      IF(LOOP.GT.6) GOTO 6                                              THRO 570
  447 CONTINUE                                                          THRO 580
      IF(MODECH.EQ.1)GOTO 21                                            THRO 590
      DO 20 I=1,NINCH                                                   THRO 600
   20 IF(LOOP.GT.1.AND.PRF(I).LT.1.01*PG) PG=PRF(I)-.01*PRF(I)          THRO 610
   21 NCLP=3                                                            THRO 620
      CALL CHAMBR                                                       THRO 630
      IF(MTOTMX.GT.0) NINCH=1                                           THRO 640
      PGCH(LOOP)=PG                                                     THRO 650
      MOCH(LOOP)=MO                                                     THRO 660
      POCL(LOOP)=POG(1)                                                 THRO 670
      CLP(1)=POG(1)*(1.+1./(4.*RCRT/GAMG(1)+1.))*(2./(GAMG(1)+1.))      THRO 680
     1       **(GAMG(1)/(GAMG(1)-1.))                                   THRO 690
      RGAM=1.-2./(4.*RCRT+GAMG(1))+1./(16.*RCRT*RCRT+8.*GAMG(1)*RCRT    THRO 691
     1     +GAMG(1)*GAMG(1))                                            THRO 692
      CLPDEC=POG(1)*(1.-(GAMG(1)-1.)/(GAMG(1)+1.)*RGAM)                 THRO 693
     1       **(GAMG(1)/(GAMG(1)-1.))                                   THRO 694
      IF(ERWT(4).NE.0.)  CLP(1)=.5*(CLP(1)+CLPDEC)                      THRO 695
      L=0                                                               THRO 700
      IF(ERWT(3).GT.0.) GOTO 11                                         THRO 710
      ERWT(1)=1./CLP(1)                                                 THRO 720
      ERWT(2)=1.-ERWT(1)                                                THRO 730
   11 L=L+1                                                             THRO 740
      Y(1)=CLP(L)                                                       THRO 750
      Y(2)=PSIO1(1)                                                     THRO 760
      DERY(1)=ERWT(1)                                                   THRO 770
      DERY(2)=ERWT(2)                                                   THRO 780
      J=0                                                               THRO 790
      CALL HPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)                     THRO 800
      DUM1(3)=PSI(J)                                                    THRO 810
      DUM1(2)=PSI(J-1)                                                  THRO 820
      DUM1(1)=PSI(J-2)                                                  THRO 830
      DUM2(3)=XSAVE(J)                                                  THRO 840
      DUM2(2)=XSAVE(J-1)                                                THRO 850
      DUM2(1)=XSAVE(J-2)                                                THRO 860
      CALL AITKEN(DUM1,DUM2,3,2,1.,RWALL)                               THRO 870
      RW(L)=RWALL                                                       THRO 880
      IF(L.EQ.2)GOTO 552                                                THRO 890
      CLP(2)=1.05*CLP(1)                                                THRO 900
```

```
      GOTO 11
  552 MCUE=0
      CALL UPORD(RW,CLP,2,MORDER)
  553 NCLPM1=NCLP-1
      NCLPM2=NCLP-2
      NCLPM3=NCLP-3
      PP=1.
      DRDP=(RW(2)-RW(1))/(CLP(2)-CLP(1))
      DRDPSV(NCLPM2)=DRDP
      IF(DRDP.GT.0.)PP=-1.
      CLP(NCLP)=CLP(1)+PP*.05*CLP(1)
      IF(NCLP.EQ.3) CLP(3)=(CLP(2)+CLP(1))*.5
      IF(NCLP.EQ.3) GOTO 554
      IF(ABS(DRDPSV(NCLPM2)/ABS(DRDPSV(NCLPM2))-DRDPSV(NCLPM3)/
     1   ABS(DRDPSV(NCLPM3))).GT.1.) MCUE=1
      IF(MCUE.EQ.1) CLP(NCLP)=CLP(1)-PP*.02*CLP(1)
  554 Y(1)=CLP(NCLP)
      Y(2)=PSIO1(1)
      DERY(1)=ERWT(1)
      DERY(2)=ERWT(2)
      J=0
      CALL HPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
      DUM1(3)=PSI(J)
      DUM1(2)=PSI(J-1)
      DUM1(1)=PSI(J-2)
      DUM2(3)=XSAVE(J)
      DUM2(2)=XSAVE(J-1)
      DUM2(1)=XSAVE(J-2)
      CALL AITKEN(DUM1,DUM2,3,2,1.,RWALL)
      RW(NCLP)=RWALL
      IF(MCUE.EQ.0) GOTO 555
      CALL UPORD(RW,CLP,NCLP,MORDER)
      GOTO 556
  555 DRDPN=(RW(NCLP)-RW(1))/(CLP(NCLP)-CLP(1))
      DRDP2=(RW(2)-RW(1))/(CLP(2)-CLP(1))
      IF(ABS(DRDPN/ABS(DRDPN)-DRDP2/ABS(DRDP2)).GT.1.) GOTO 557
      CALL UPORD(RW,CLP,NCLP,MORDER)
      NCLP=NCLP+1
      GOTO 553
  557 RW(3)=RW(NCLP)
      CLP(3)=CLP(NCLP)
  556 NCLPM1=3
      NCLP=4
      GOTO 10
    3 NCLP=NCLP+1
      NCLPM1=NCLP-1
   10 IPLY=IPLY+1
      CALL UPORD (CLP,RW,NCLPM1,MORDER)
      CALL PLYCF(CLP,RW,NCLPM1,ACO,2,NSTOPD)
      CLP(NCLP)=-ACO(2)/ACO(3)*.5
```

```
      Y(1)=CLP(NCLP)                                                    THR01410
      Y(2)=PSI01(1)                                                     THR01420
      DERY(1)=ERWT(1)                                                   THR01430
      DERY(2)=ERWT(2)                                                   THR01440
      J=0                                                               THR01450
      CALL HPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)                     THR01460
      DUM1(3)=PSI(J)                                                    THR01470
      DUM1(2)=PSI(J-1)                                                  THR01480
      DUM1(1)=PSI(J-2)                                                  THR01490
      DUM2(3)=XSAVE(J)                                                  THR01500
      DUM2(2)=XSAVE(J-1)                                                THR01510
      DUM2(1)=XSAVE(J-2)                                                THR01520
      CALL AITKEN(DUM1,DUM2,3,2,1.,RWALL)                               THR01530
      RW(NCLP)=RWALL                                                    THR01540
      CALL UPORD(RW,CLP,NCLP,MORDER)                                    THR01550
      IF(IPLY.GT.6) GOTO 7                                              THR01560
      GOTO 558                                                          THR01570
    7 WRITE(6,701) LOOP                                                 THR01580
  701 FORMAT(/,44H IPLY=7. UNSATISFACTORY CONVERGENCE IN LOOP=,I2,/)    THR01590
      GOTO 4                                                            THR01600
  558 CONTINUE                                                          THR01610
      NCLP=3                                                            THR01620
      IF(IPLY.LT.2) GOTO 3                                              THR01630
      RWCHK=RW(1)+.0005*RW(1)                                           THR01640
      IF(RWCHK.LT.RW(2))GOTO 3                                          THR01650
      IF(ABS(CLP(1)-CLP(2))/CLP(1).GT..010) GOTO 3                      THR01660
    4 IF(ABS(RW(1)-RT)/RT.LE.0.0005)GOTO 6                              THR01670
      RMIN(LOOP)=RW(1)                                                  THR01680
      RMINSQ(LOOP)=RMIN(LOOP)*RMIN(LOOP)                                THR01690
      URMNSQ(LOOP)=1./RMINSQ(LOOP)                                      THR01700
      IF(MODECH.EQ.3) GOTO 5                                            THR01710
      IF(MODECH.EQ.2.AND.LOOP.GT.4) GOTO 14                             THR01711
      IF(MODECH.EQ.2.AND.LOOP.GE.3) PG=PG*(RW(1)/RT)**2                 THR01712
      IF(MODECH.EQ.2.AND.LOOP.GE.3) GOTO 2                              THR01713
      IF(MODECH.EQ.2.AND.LOOP.EQ.2)PG=PG+(RT-RW(1))*(PG-PGCH(1))/       THR01714
     1(RW(1)-RMIN(1))                                                   THR01715
      IF(MODECH.EQ.2.AND.LOOP.EQ.2) GOTO 2                              THR01716
      IF(MODECH.EQ.2)PG=PG*RW(1)/RT                                     THR01717
      IF(MODECH.EQ.2) GOTO 2                                            THR01718
      IF(LOOP.GT.5) GOTO 14                                             THR01720
      PG=PG*(RW(1)/RT)**2                                               THR01730
      GOTO 2                                                            THR01740
   14 CALL UPORD(RMINSQ,PGCH,LOOP,MORDER)                               THR01750
      CALL AITKEN(RMINSQ,PGCH,LOOP,2,RTSQ,APGCH)                        THR01760
      PG=APGCH                                                          THR01770
      GOTO 2                                                            THR01780
    5 IF(LOOP.GT.5) GOTO 8                                              THR01790
      MO=MO*(RT/RW(1))**2                                               THR01800
      GOTO 2                                                            THR01810
    8 CALL UPORD(URMNSQ,MOCH,LOOP,MORDER)                               THR01820
```

```
      CALL AITKEN(URMNSQ,MOCH,LOOP,2,URTSQ,AMO)                          THR01830
      MO=AMO                                                             THR01840
      GOTO 2                                                             THR01850
    6 RWCALC=RW(1)                                                       THR01860
      PGCALC=PG                                                          THR01870
      CLPCAL=CLP(1)                                                      THR01880
  444 CONTINUE                                                           THR01890
      NNINCH=NCHA1+NCHA2                                                 THR01900
      WRITE(6,222)LOOP,RCRT,MODECH,MGORLI,NAORW,NCEC71,    NSPPG         THR01910
     1,MTOTMX,LETOUT,NNINCH                                              THR01920
  222 FORMAT(2X,5HLOOP=,I2,4X,6HRC/RT=,E15.6,4X,7HMODECH=,I2,4X,7HMGORLITHR01930
     1=,I2,4X,6HNAORW=,I2,4X,7HNCEC71=,I2,/,38X,                         THR01940
     2     6HNSPPG=,I2,5X,7HMTOTMX=,I2,4X,7HLETOUT=,I2,3X,6HNINCH=,I2,   THR01950
     3/)                                                                 THR01960
      WRITE(6,111)RT,RWCALC,PGCALC,EPSC,CLPCAL                           THR01970
  111 FORMAT(///,5H  RT=,E15.6,5X,7HRTCALC=,E15.6,5X,3HPG=,    E15.6,5X,THR01980
     15HEPSC=,E15.6,5X,4HCLP=,    E15.6,/)                               THR01990
      WRITE(6,300)                                                       THR02000
  300 FORMAT(//////,54X,24HINJECTOR       PARAMETERS,//)                 THR02010
      WRITE(6,223) (PSI01(I),I=1,NCHA1P)                                 THR02020
  223 FORMAT(2X,18HZONE PSI LIMITS  =,20E10.4)                           THR02030
      IF(NCHA2.EQ.0) GOTO 225                                            THR02040
      WRITE(6,223) (PSI02(I),I=1,NCHA2P)                                 THR02050
  225 CONTINUE                                                           THR02060
      WRITE(6,226)                                                       THR02070
  226 FORMAT(/,5H ZONE,6X,2HPI,12X,4HTIFU,10X,4HAIFU,10X,4HWIFU,10X,     THR02080
     1 6HRHOIFU,8X,4HRIFU,10X,6HMDTIFU,8X,3HPRF,/,25X,4HTIOX,10X,        THR02090
     2 4HAIOX,10X,4HWIOX,10X,6HRHOIOX,8X,4HRIOX,10X,6HMDTIOX,8X,3HPRO,/)THR02100
      DO 301 I=1,NNINCH                                                  THR02110
  301 WRITE(6,302)I,PI(I),TIFU(I),AIFU(I),WIFU(I),RHOIFU(I),RIFU(I),     THR02120
     1 MDTIFU(I),PRF(I),I,TIOX(I),AIOX(I),WIOX(I),RHOIOX(I),RIOX(I),     THR02130
     2 MDTIOX(I),PRO(I)                                                  THR02140
  302 FORMAT(I5,8E14.6,/,I5,14X,7E14.6,/)                                THR02150
      IF(MODECH.EQ.1) WRITE(6,305)                                       THR02151
  305 FORMAT(1H1)                                                        THR02152
      IF(MTOTMX.EQ.0) WRITE(6,303)                                       THR02160
      IF(MTOTMX.GT.0) WRITE(6,304)                                       THR02170
  303 FORMAT(//////,53X,25HBURNED   GAS   PARAMETERS,//)                 THR02180
  304 FORMAT(//////,54X,24HMIXED   GAS   PARAMETERS,//)                  THR02190
      WRITE(6,221)                                                       THR02200
  221 FORMAT(/6X,3HPOG,11X,3HTOG,11X,3HCPG,11X,4HGAMG,10X,2HTG,12X,2HAG,THR02210
     112X,2HWG,12X,4HRHOG,10X,4HMDTI,/)                                  THR02220
      IF(MTOTMX.GT.0) NINCH=1                                            THR02230
      IF(LOOP.LE.7) GOTO 18                                              THR02240
      TOTMO=0.                                                           THR02250
      DO 230 I=1,NNINCH                                                  THR02260
  230 TOTMO=TOTMO+MDTI(I)                                                THR02270
      MO=TOTMO                                                           THR02280
   18 CONTINUE                                                           THR02290
      DO 227 I=1,NINCH                                                   THR02300
```

```
      DTI=MDTI(I)                                                       THRO2310
      IF(MTOTMX.GT.0) DTI=MO                                            THRO2320
  227 WRITE(6,228)POG(I),TOG(I),CPG(I),GAMG(I),TG(I),AG(I),WG(I),RHOG(I)THRO2330
     1   ,DTI                                                           THRO2340
  228 FORMAT(9E14.6)                                                    THRO2350
      IF(MTOTMX.GT.0) WRITE(6,443) PMIX                                 THRO2360
  443 FORMAT(///,2X,29HSTATIC PRESS. OF MIXED GASES=,E15.6)             THRO2370
      IF(MTOTMX.GT.0) WRITE(6,442)                                      THRO2380
  442 FORMAT(2X,51HTOTAL PRESS. OF PRODUCTS IN EACH ZONE BEFORE MIXING) THRO2390
      IF(MTOTMX.GT.0) WRITE(6,441) (PO(I), I=1,NNINCH)                  THRO2400
  441 FORMAT(10E15.6)                                                   THRO2410
C    **********************                                             THRO2420
C    *PERFORMANCE SECTION.*                                             THRO2430
C    **********************                                             THRO2440
      Y(1)=CLP(1)                                                       THRO2450
      Y(2)=PSIO1(1)                                                     THRO2460
      DERY(1)=ERWT(1)                                                   THRO2470
      DERY(2)=ERWT(2)                                                   THRO2480
      J=0                                                               THRO2490
      CALL HPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)                     THRO2500
      DUM1(3)=PSI(J)                                                    THRO2510
      DUM1(2)=PSI(J-1)                                                  THRO2520
      DUM1(1)=PSI(J-2)                                                  THRO2530
      DUM2(3)=XSAVE(J)                                                  THRO2540
      DUM2(2)=XSAVE(J-1)                                                THRO2550
      DUM2(1)=XSAVE(J-2)                                                THRO2560
      CALL AITKEN(DUM1,DUM2,3,2,1.,RWALL)                               THRO2570
      JM1=J-1                                                           THRO2580
      JM2=J-2                                                           THRO2590
      I=1                                                               THRO2600
      YSAVE(I,4)=TWOPI*XSAVE(2)*XSAVE(2)/8.                             THRO2610
      DO 50 I=2,JM2                                                     THRO2620
   50 YSAVE(I,4)=TWOPI/8.*((XSAVE(I+1)+XSAVE(I))**2-(XSAVE(I)+XSAVE(I-1)THRO2630
     1   )**2)                                                          THRO2640
      YSAVE(JM1,4)=TWOPI*RWALL*RWALL/2.-TWOPI/8.*(XSAVE(JM1)+XSAVE(JM2) THRO2650
     1    )**2                                                          THRO2660
      ISPVAC=YSAVE(2,2)/2.*YSAVE(1,3)+YSAVE(1,1)*YSAVE(1,4)/MO          THRO2670
      DO 51 I=2,JM2                                                     THRO2680
      IP1=I+1                                                           THRO2690
      IM1=I-1                                                           THRO2700
   51 ISPVAC=ISPVAC+(YSAVE(IP1,2)-YSAVE(IM1,2))/2.*YSAVE(I,3)           THRO2710
     1    +YSAVE(I,1)*YSAVE(I,4)/MO                                     THRO2720
      ISPVAC=ISPVAC+(1.-.5*YSAVE(JM1,2) -.5*YSAVE(JM2,2))*YSAVE(JM1,3)  THRO2730
     1    +YSAVE(JM1,1)*YSAVE(JM1,4)/MO                                 THRO2740
      RADT(1)=0.                                                        THRO2750
      RADT(2)=RWALL                                                     THRO2760
      IF(MTOTMX.GT.0) GOTO 59                                           THRO2770
      IF(NCHA1.EQ.1.AND.NCHA2.EQ.0)GOTO 59                              THRO2780
      DO 56 II=1,NCHA1P                                                 THRO2790
      DO 55 I=1,J                                                       THRO2800
```

```
      IF(YSAVE(I,2).GE.PSIO1(II)) RADT(II)=XSAVE(I)                 THRO2810
      IF(YSAVE(I,2).GE.PSIO1(II)) GOTO 56                           THRO2820
   55 CONTINUE                                                      THRO2830
   56 CONTINUE                                                      THRO2840
      IF(NCHA2.EQ.0) RADT(NCHA1P)=RWALL                             THRO2850
      IF(NCHA2.EQ.0) GOTO 59                                        THRO2860
      RADT(NCH1P2)=RWALL                                            THRO2870
      IF(NCHA2.EQ.1)GOTO 59                                         THRO2880
      DO 58 II=2,NCHA2P                                             THRO2890
      III=NCHA1P+II-1                                               THRO2900
      DO 57 I=1,J                                                   THRO2910
      IF(YSAVE(I,2).GE.PSIO2(II))RADT(III)=XSAVE(I)                 THRO2920
      IF(YSAVE(I,2).GE.PSIO2(II))             GOTO 58               THRO2930
   57 CONTINUE                                                      THRO2940
   58 CONTINUE                                                      THRO2950
      RADT(III)=RWALL                                               THRO2960
   59 CONTINUE                                                      THRO2970
      MO1D=0.                                                       THRO2980
      DO 64 I=1,NINCH                                               THRO2990
      IP=I+1                                                        THRO3000
      ASTAR=TWOPI/2.*(RADT(IP)*RADT(IP)-RADT(I)*RADT(I))            THRO3010
      PIR=GAMG(I)/RGAS(I)*(2./(GAMG(I)+1.))**((GAMG(I)+1.)/(GAMG(I)-1.))THRO3020
      IF(NCEC71.GE.2) MO1D=MO1D+ASTAR*PG/CST(I)                     THRO3030
      IF(NCEC71.LT.2) MO1D=MO1D+ASTAR*POG(I)*SQRT(PIR/TOG(I))       THRO3040
   64 CONTINUE                                                      THRO3050
      CD=MO/MO1D                                                    THRO3060
      WTPO=0.                                                       THRO3070
      DO 52 I=2,NCHA1P                                              THRO3080
      IM1=I-1                                                       THRO3090
   52 WTPO=POG(IM1)*(PSIO1(I)-PSIO1(IM1))+WTPO                      THRO3100
      IF(NCHA2.EQ.0)GOTO 54                                         THRO3110
      DO 53 I=2,NCHA2P                                              THRO3120
      IM1=NCHA1+I-1                                                 THRO3130
   53 WTPO=POG(IM1)*(PSIO2(I)-PSIO2(I-1))+WTPO                      THRO3140
   54 CONTINUE                                                      THRO3150
      CSTAR=WTPO*TWOPI*RWALL*RWALL/2./MO                            THRO3160
      CST1D=WTPO*TWOPI*RWALL*RWALL/2./MO1D                          THRO3170
      ISPV1D=0.                                                     THRO3180
      IF(NCEC71.GT.1) GOTO 70                                       THRO3190
      NCHA11=NCHA1                                                  THRO3200
      IF(MTOTMX.GT.0)NCHA11=NINCH                                   THRO3210
      DO 60 I=1,NCHA11                                              THRO3220
      TOTO=1./(1.+(GAMG(I)-1.)/2.)                                  THRO3230
      TEMP=TOG(I)*TOTO                                              THRO3240
      U=SQRT(GAMG(I)*RGAS(I)*TEMP)                                  THRO3250
      P=POG(I)*TOTO**(GAMG(I)/(GAMG(I)-1.))                         THRO3260
      IK=I+1                                                        THRO3270
      DPSI=PSIO1(IK)-PSIO1(I)                                       THRO3280
      DAREA=TWOPI/2.*(RADT(IK)*RADT(IK)-RADT(I)*RADT(I))            THRO3290
   60 ISPV1D=ISPV1D+DPSI*U+P*DAREA/MO1D                             THRO3300
```

```
      IF(NCHA2.EQ.0)GOTO 62                                             THRO3310
      IK=1                                                              THRO3320
      DO 61 I=NCHA1P,NINCH                                              THRO3330
      IKE=I+1                                                           THRO3340
      TOTO=1./(1.+(GAMG(I)-1.)/2.)                                      THRO3350
      TEMP=TOG(I)*TOTO                                                  THRO3360
      U=SQRT(GAMG(I)*RGAS(I)*TEMP)                                      THRO3370
      P=POG(I)*TOTO**(GAMG(I)/(GAMG(I)-1.))                             THRO3380
      IK=IK+1                                                           THRO3390
      IKM=IK-1                                                          THRO3400
      DPSI=PSI02(IK)-PSI02(IKM)                                         THRO3410
      DAREA=TWOPI/2.*(RADT(IKE)*RADT(IKE)-RADT(I)*RADT(I))              THRO3420
   61 ISPV1D=ISPV1D+DPSI*U+P*DAREA/MO1D                                 THRO3430
   62 CONTINUE                                                          THRO3440
      GOTO 71                                                           THRO3450
   70 DO 72 I=1,NINCH                                                   THRO3460
   72 ISPV1D=ISPV1D+VAC1DI(I)*MDTI(I)/MO*32.174                         THRO3470
   71 CONTINUE                                                          THRO3480
      VAOV1D=ISPVAC/ISPV1D                                              THRO3490
   68 FORMAT(1H1,59X,12HTHROAT PLANE,///)                               THRO3500
      WRITE(6,68)                                                       THRO3510
      CSOC1D=CSTAR/CST1D                                                THRO3520
      WRITE(6,65)                                                       THRO3530
   65 FORMAT(     6X,1HR,14X,1HP,13X,3HPSI,13X,1HT,14X,1HW,14X,1HM,     THRO3540
     1  13X,3HGAM,/)                                                    THRO3550
      WRITE(6,66) (XSAVE(I),YSAVE(I,1),YSAVE(I,2),T(I),YSAVE(I,3),      THRO3560
     1   YPSAV(I,4),GAMSAV(I),I=1,J)                                    THRO3570
   66 FORMAT(7E15.6)                                                    THRO3580
      DUM2(3)=YSAVE(J,1)                                                THRO3590
      DUM2(2)=YSAVE(JM1,1)                                              THRO3600
      DUM2(1)=YSAVE(JM2,1)                                              THRO3610
      CALL AITKEN(DUM1,DUM2,3,2,1.,PWAL)                                THRO3620
      DUM2(3)=T(J)                                                      THRO3630
      DUM2(2)=T(JM1)                                                    THRO3640
      DUM2(1)=T(JM2)                                                    THRO3650
      CALL AITKEN(DUM1,DUM2,3,2,1.,TWAL)                                THRO3660
      DUM2(3)=YSAVE(J,3)                                                THRO3670
      DUM2(2)=YSAVE(JM1,3)                                              THRO3680
      DUM2(1)=YSAVE(JM2,3)                                              THRO3690
      CALL AITKEN(DUM1,DUM2,3,2,1.,WWAL)                                THRO3700
      DUM2(3)=YPSAV(J,4)                                                THRO3710
      DUM2(2)=YPSAV(JM1,4)                                              THRO3720
      DUM2(1)=YPSAV(JM2,4)                                              THRO3730
      CALL AITKEN(DUM1,DUM2,3,2,1.,EMWAL)                               THRO3740
      DUM2(3)=GAMSAV(J)                                                 THRO3750
      DUM2(2)=GAMSAV(JM1)                                               THRO3760
      DUM2(1)=GAMSAV(JM2)                                               THRO3770
      CALL AITKEN(DUM1,DUM2,3,2,1.,GAMWAL)                              THRO3780
      PSIWAL=1.                                                         THRO3790
      WRITE(6,69)                                                       THRO3800
```

```
 69 FORMAT(1X,11HTHROAT WALL)                                             THR03810
    WRITE(6,66)RWALL,PWAL,PSIWAL,TWAL,WWAL,EMWAL,GAMWAL                   THR03820
    WRITE(6,67)                                                           THR03830
 67 FORMAT(1H1,60X,11HPERFORMANCE,///)                                    THR03840
    ISPVAC=ISPVAC/32.174                                                  THR03841
    ISPV1D=ISPV1D/32.174                                                  THR03842
    WRITE(6,63)ISPVAC,CSTAR,ISPV1D, CST1D,MO,MO1D,CD,VAOV1D,CSOC1D        THR03850
 63 FORMAT(//,48X,39HTHROAT ISPVAC,LBF-SEC/LBM  CSTAR,FT/SEC,/,53X,       THR03860
   12E17.6,//,48X,39HTHROAT ISPV1D,LBF-SEC/LBM  CST1D,FT/SEC,/,53X,       THR03870
   22E17.6,//,48X,13HMO,LBF-SEC/FT,9X,15HMO1D,LBF-SEC/FT,/,46X,E16.6,     THR03880
   36X,E16.6,//,48X,25HCD,DISCHARGE COEFFICIENT=,E15.6,//,48X,21HTHROATHR03890
   4T ISPVAC/ISPV1D=,E15.6,//,48X,12HCSTAR/CST1D=,E15.6,/,1H1)            THR03900
    RETURN                                                                THR03910
    END                                                                   THR03920
```

```
      SUBROUTINE CHAMBR
      REAL MO1,MO2,MO,MDTI,MWG,MOCH,MDTIFU,MDTIOX
      DIMENSION TGA(20,10),SUMAGG(20),PII(20)
      COMMON     PRMT(6),DERY(4),AUX(16,4),Y(4),RW(10)
     1           ,POCL(7),ERWT(4),RMIN(7),MODECH,RMINSQ(7)
     2           ,PGCH(7),CLP(10),RLIM,MORDER,LOOP,MOCH(7),URMNSQ(7)
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSI01(10),PSI02(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      COMMON/CALC/HO,C,GAM,GAMSAV(200),K
      COMMON/MIXED/PMIX,PO(20)
      NINCH=NCHA1+NCHA2
      FNINCH=NINCH
      NCHA1P=NCHA1+1
      NCHA2P=NCHA2+1
      NINCHP=NINCH+1
      LOOPC=1
      NOAI=1
      IF(MODECH.EQ.1)PIAVG=1.01*PG
      IF(MODECH.EQ.2) CALL PRSPFD(NOAI,NINCH,PIAVG)
      IF(MODECH.EQ.3) CALL PGFIX
      IF(MODECH.EQ.3) PIAVG=1.01*PG
      MO=0.
      DO 32 I=1,NINCH
   32 MO=MDTI(I)+MO
      DO 1 I=1,NINCHP
      TG(I)=0.
      AG(I)=0.
      WG(I)=0.
      IF(MODECH.EQ.1) PRF(I)=0.
      IF(MODECH.EQ.1) PRO(I)=0.
    1 RHOG(I)=0.
      IF(MODECH.EQ.3) GOTO 3
      CALL EQULIB(NINCH)
    3 DO 4 I=1,NINCH
      IF(NINCH.GT.1) NSPPG=0
      IF(NSPPG.EQ.1.AND.MODECH.EQ.1) CALL SPPG(PIAVG)
      IF(NSPPG.EQ.1.AND.MODECH.EQ.3) CALL SPPG(PIAVG)
      IF(NSPPG.EQ.1) GOTO 11
      PI(I)=1.01*PG
      IF(MODECH.EQ.2)PI(I)=PIAVG
      IF(MGORLI.EQ.2) GOTO 40
      RHOIFU(I)=PI(I)/RIFU(I)/TIFU(I)
      IF(OXFULE(I).EQ.0.) RHOIOX(I)=0.
```

```
      IF(OXFULE(I).EQ.0.) GOTO 40                                       CHAM 500
      RHOIOX(I)=PI(I)/RIOX(I)/TIOX(I)                                   CHAM 510
   40 CONTINUE                                                          CHAM 520
      IF(NAORW.EQ.1)WIFU(I)=MDTIFU(I)/AIFU(I)/RHOIFU(I)                 CHAM 530
      IF(NAORW.EQ.2)AIFU(I)=MDTIFU(I)/WIFU(I)/RHOIFU(I)                 CHAM 540
      IF(OXFULE(I).EQ.0.) WIOX(I)=0.                                    CHAM 550
      IF(OXFULE(I).EQ.0.) AIOX(I)=0.                                    CHAM 560
      IF(OXFULE(I).EQ.0.) GOTO 4                                        CHAM 570
      IF(NAORW.EQ.1)WIOX(I)=MDTIOX(I)/AIOX(I)/RHOIOX(I)                 CHAM 580
      IF(NAORW.EQ.2)AIOX(I)=MDTIOX(I)/WIOX(I)/RHOIOX(I)                 CHAM 590
    4 CONTINUE                                                          CHAM 600
      GOTO 5                                                            CHAM 610
   10 DO 2 I=1,NINCH                                                    CHAM 620
    2 PI(I)=PG+(PI(I)-PG)*(SUMAG/AC)**2                                 CHAM 630
      IF(MODECH.NE.2) GOTO 22                                           CHAM 631
      DO 23 I=1,NINCH                                                   CHAM 632
      IF(PRF(I).LE.PI(I)) PI(I)=.91*PRF(I)                              CHAM 633
   23 CONTINUE                                                          CHAM 634
   22 CONTINUE                                                          CHAM 635
      PIAVG=PI(1)                                                       CHAM 640
      IF(NINCH.EQ.1)GOTO 27                                             CHAM 650
      DO 25 I=2,NINCH                                                   CHAM 660
   25 PIAVG=PIAVG+PI(I)                                                 CHAM 670
      PIAVG=PIAVG/FNINCH                                                CHAM 680
  251 IF(PIAVG.LT.PG)PIAVG=PG+.002*PG                                   CHAM 690
      DO 26 I=1,NINCH                                                   CHAM 700
   26 PI(I)=PIAVG                                                       CHAM 710
   27 CONTINUE                                                          CHAM 720
      IF(NINCH.EQ.1.AND.PIAVG.LE.PG)PI(1)=PG+.002*PG                    CHAM 730
      PIAVG=PI(1)                                                       CHAM 740
      DO 24 I=1,NINCH                                                   CHAM 750
      IF(MGORLI.EQ.2) GOTO 41                                           CHAM 760
      RHOIFU(I)=PI(I)/RIFU(I)/TIFU(I)                                   CHAM 770
      IF(OXFULE(I).EQ.0.) RHOIOX(I)=0.                                  CHAM 780
      IF(OXFULE(I).EQ.0.) GOTO 41                                       CHAM 790
      RHOIOX(I)=PI(I)/RIOX(I)/TIOX(I)                                   CHAM 800
   41 CONTINUE                                                          CHAM 810
      IF(NAORW.EQ.2)AIFU(I)=MDTIFU(I)/WIFU(I)/RHOIFU(I)                 CHAM 820
      IF(OXFULE(I).EQ.0.) AIOX(I)=0.                                    CHAM 830
      IF(OXFULE(I).EQ.0.) GOTO 24                                       CHAM 840
      IF(NAORW.EQ.2)AIOX(I)=MDTIOX(I)/WIOX(I)/RHOIOX(I)                 CHAM 850
   24 CONTINUE                                                          CHAM 860
      IF(MODECH.EQ.2) CALL PRSPFD(NOAI,NINCH,PIAVG)                     CHAM 870
      DO 20 I=1,NINCH                                                   CHAM 880
      IF(NAORW.EQ.1)WIFU(I)=MDTIFU(I)/AIFU(I)/RHOIFU(I)                 CHAM 890
      IF(OXFULE(I).EQ.0.) WIOX(I)=0.                                    CHAM 900
      IF(OXFULE(I).EQ.0.) GOTO 20                                       CHAM 910
      IF(NAORW.EQ.1)WIOX(I)=MDTIOX(I)/AIOX(I)/RHOIOX(I)                 CHAM 920
   20 CONTINUE                                                          CHAM 930
    5 DO 6 I=1,NINCH                                                    CHAM 940
```

```
      TG(I)=TOG(I)-WG(I)*WG(I)/2./CPG(I)                                 CHAM 950
      TGA(I,LOOPC)=TG(I)                                                 CHAM 960
      RHOG(I)=PG/RGAS(I)/TG(I)                                           CHAM 970
      BOB=-(WIFU(I)+OXFULE(I)*WIOX(I))/(1.+OXFULE(I))                    CHAM 980
      COC=(PG-PI(I))/RHOG(I)                                             CHAM 990
      WG(I)=(-BOB+SQRT(BOB*BOB-4.*COC))*.5                               CHAM1000
    6 AG(I)=MDTI(I)/RHOG(I)/WG(I)                                        CHAM1010
      IF(LOOPC.GT.1)GOTO 7                                               CHAM1020
      LOOPC=LOOPC+1                                                      CHAM1030
      GOTO 5                                                             CHAM1040
   16 LOOPC=LOOPC+1                                                      CHAM1050
      GOTO 5                                                             CHAM1060
    7 LOOPCM=LOOPC-1                                                     CHAM1070
      DO 8 I=1,NINCH                                                     CHAM1080
      IF(ABS(TGA(I,LOOPC)-TGA(I,LOOPCM))/TGA(I,LOOPC).GT..0010)GOTO 16   CHAM1090
    8 CONTINUE                                                           CHAM1100
      SUMAG=0.                                                           CHAM1110
      DO 9 I=1,NINCH                                                     CHAM1120
    9 SUMAG=SUMAG+AG(I)                                                  CHAM1130
      IF(ABS(SUMAG-AC)/AC.LT..0010)GOTO 11                               CHAM1140
      SUMAGG(NOAI)=SUMAG                                                 CHAM1150
      PII(NOAI)=PIAVG                                                    CHAM1160
   31 NOAI=NOAI+1                                                        CHAM1170
      LOOPC=1                                                            CHAM1180
      IF(NOAI.GT.20)GOTO 17                                              CHAM1190
      GOTO 10                                                            CHAM1200
   11 CONTINUE                                                           CHAM1210
      MO1=0.                                                             CHAM1220
      MO2=0.                                                             CHAM1230
C                                                                        CHAM1240
C     CONVERGENCE ACHIEVED ON TG AND SUMAG                               CHAM1250
C                                                                        CHAM1260
      DO 12 I=1,NCHA1                                                    CHAM1270
   12 MO1=RHOG(I)*AG(I)*WG(I)+MO1                                        CHAM1280
      IF(NCHA2.EQ.0) GOTO 28                                             CHAM1290
      DO 13 I=NCHA1P,NINCH                                               CHAM1300
   13 MO2=RHOG(I)*AG(I)*WG(I)+MO2                                        CHAM1310
   28 MO=0.                                                              CHAM1320
      DO 34 I=1,NINCH                                                    CHAM1330
   34 MO=MO+MDTI(I)                                                      CHAM1340
      RAD(1)=SQRT(2.*AG(1)/TWOPI)                                        CHAM1350
      IF(NINCH.EQ.1) GOTO 29                                             CHAM1360
      DO 14 I=2,NINCH                                                    CHAM1370
      IM1=I-1                                                            CHAM1380
   14 RAD(I)=SQRT(2.*AG(I)/TWOPI+RAD(IM1)*RAD(IM1))                      CHAM1390
   29 CONTINUE                                                           CHAM1400
      DO 18 I=1,NINCH                                                    CHAM1410
      GOGM1=GAMG(I)/(GAMG(I)-1.)                                         CHAM1420
      IF(MODECH.GT.1)GOTO 21                                             CHAM1430
      PRF(I)=PIAVG+.5*(MDTIFU(I)/AIFU(I)/.7)**2/RHOIFU(I)                CHAM1440
```

```
      IF(OXFULE(I).EQ.0.) PRO(I)=0.                                     CHAM1450
      IF(OXFULE(I).EQ.0.) GOTO 21                                       CHAM1460
      PRO(I)=PIAVG+.5*(MDTIOX(I)/AIOX(I)/.7)**2/RHOIOX(I)               CHAM1470
   21 POG(I)=PG*(TOG(I)/TG(I))**GOGM1                                   CHAM1480
   18 PO(I)=POG(I)                                                      CHAM1490
      PSI01(1)=0.                                                       CHAM1500
      DO 15 I=2,NCHA1P                                                  CHAM1510
   15 PSI01(I)=PSI01(I-1)+RHOG(I-1)*AG(I-1)*WG(I-1)/MO                  CHAM1520
      PSI12=PSI01(NCHA1P)                                               CHAM1530
      PSI02(1)=PSI01(NCHA1P)                                            CHAM1540
      IF(NCHA2.LT.1)GOTO 17                                             CHAM1550
      DO 19 I=2,NCHA2P                                                  CHAM1560
      NCHAIM=NCHA1+I-1                                                  CHAM1570
   19 PSI02(I)=PSI02(I-1)+RHOG(NCHAIM)*AG(NCHAIM)*WG(NCHAIM)/MO         CHAM1580
   17 CONTINUE                                                          CHAM1590
      IF(NOAI.GT.20) WRITE(6,33) PG,PIAVG,SUMAG,AC                      CHAM1600
   33 FORMAT(1X,41HCONVERGENCE FAILED. PG,PIAVG,SUMAG,AC ARE,/,4E15.6,/)CHAM1610
      IF(MTOTMX.GT.0) GOTO 50                                           CHAM1620
      RETURN                                                            CHAM1630
   50 CPMIX=0.                                                          CHAM1640
      WMMIX=0.                                                          CHAM1650
      TOMIX=0.                                                          CHAM1660
      VMIX=0.                                                           CHAM1670
      PSI01(1)=0.                                                       CHAM1680
      PSI01(2)=1.                                                       CHAM1690
      PSI12=1.                                                          CHAM1700
      PSI02(1)=1.                                                       CHAM1710
      AG(1)=SUMAG                                                       CHAM1720
      DO 51 I=1,NINCH                                                   CHAM1730
      CPMIX=CPMIX+CPG(I)*MDTI(I)/MO                                     CHAM1740
      VMIX=VMIX+MDTI(I)*WG(I)/MO                                        CHAM1750
   51 WMMIX=WMMIX+MDTI(I)/MO/MWG(I)                                     CHAM1760
      WMMIX=1./WMMIX                                                    CHAM1770
      RGAS(1)=UNIVR*32.174/WMMIX                                        CHAM1780
      GAMG(1)=CPMIX/(CPMIX-RGAS(1))                                     CHAM1790
      REW=GAMG(1)                                                       CHAM1800
      MWG(1)=WMMIX                                                      CHAM1810
      DO 52 I=1,NINCH                                                   CHAM1820
   52 TOMIX=TOMIX+CPG(I)*TOG(I)*MDTI(I)                                 CHAM1830
      CPG(1)=CPMIX                                                      CHAM1840
      TOG(1)=TOMIX/CPMIX/MO                                             CHAM1850
      GOGMIX=REW/(REW-1.)                                               CHAM1860
      B=-VMIX-PG*SUMAG/MO                                               CHAM1870
      A=.5*(REW+1.)/REW                                                 CHAM1880
      C=TOG(1)*RGAS(1)                                                  CHAM1890
      VMIX=(-B-SQRT(B*B-4.*A*C))/A*.5                                   CHAM1900
      WG(1)=VMIX                                                        CHAM1910
      RHOG(1)=MO/SUMAG/VMIX                                             CHAM1920
      TG(1)=TOG(1)-VMIX*VMIX/CPMIX*.5                                   CHAM1930
      PMIX=RHOG(1)*TG(1)*RGAS(1)                                        CHAM1940
```

```
      POG(1)=PMIX*(TOG(1)/TG(1))**GOGMIX                                CHAM1950
      RETURN                                                            CHAM1960
      END                                                               CHAM1970
```

```
      SUBROUTINE PGFIX
      REAL MOCH,MO,MDTI,MDTIFU,MDTIOX,MWG
      COMMON      PRMT(6),DERY(4),AUX(16,4),Y(4),RW(10)
     1           ,POCL(7),ERWT(4),RMIN(7),MODECH,RMINSQ(7)
     2           ,PGCH(7),CLP(10),RLIM,MORDER,LOOP,MOCH(7),URMNSQ(7)
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSI01(10),PSI02(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      NINCH=NCHA1+NCHA2
      IF(LOOP.EQ.1) CALL EQULIB(NINCH)
      IF(LOOP.EQ.1) GOTO 1
      IF(MTOTMX.GT.0.AND.LOOP.GT.1) CALL EQULIB(NINCH)
      GOTO 3
    1 ASTAR=TWOPI*RT*RT/2.
      UNIV=SQRT(UNIVR*32.174)
      SUM=0.
      DO 2 I=1,NINCH
      FOFGAM=SQRT(GAMG(I))*(2./(GAMG(I)+1.))**((GAMG(I)+1.)/(2.*GAMG(I)
     1 -2.))
    2 SUM=SUM+PCTMO(I)/FOFGAM*SQRT(TOG(I)/MWG(I))
      MO=PG*ASTAR/UNIV/SUM
    3 DO 4 I=1,NINCH
      MDTI(I)=MO*PCTMO(I)
      MDTIFU(I)=MDTI(I)/(1.+OXFULE(I))
    4 MDTIOX(I)=MDTI(I)-MDTIFU(I)
      RETURN
      END
```

```
      SUBROUTINE PRSPFD(NOAI,NINCH,PIAVG)
C
C     INJECTOR DISCHARGE COEFFS. = .7 ARE USED UNLESS READ BY NUM.
C     THIS OPTION MAY NOT BE USED IF INJECTOR VELOCITIES ARE
C     SPECIFIED AND IS NOT STRICTLY VALID FOR GAS INJECTION.
C
      REAL MDTI,MDTIFU,MDTIOX,MWG
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      IF(NOAI.GT.1) GOTO 1
      PIAVG=1.010*PG
    1 DO 2 I=1,NINCH
      IF(MGORLI.EQ.2) MDTIFU(I)=CDINJ(I)*AIFU(I)*SQRT(2.*RHOIFU(I)
     1 *(PRF(I)-PIAVG))
      IF(MGORLI.EQ.2) GOTO 2
      MDTIFU(I)=CDINJ(I)*AIFU(I)*SQRT((2.*PIAVG*PRF(I)-2.*PIAVG**2)
     1 /RIFU(I)/TIFU(I) )
    2 CONTINUE
      DO 3 I=1,NINCH
      MDTI(I)=MDTIFU(I)*(1.+OXFULE(I))
      MDTIOX(I)=MDTI(I)-MDTIFU(I)
      IF(OXFULE(I).EQ.0.) AIOX(I)=0.
      IF(OXFULE(I).EQ.0.) GOTO 3
      IF(MGORLI.EQ.2) AIOX(I)=MDTIOX(I)/CDINJ(I)/SQRT(RHOIOX(I)*2.*
     1    (PRO(I)-PIAVG))
      IF(MGORLI.EQ.1) AIOX(I)=MDTIOX(I)/CDINJ(I)/SQRT(PIAVG/TIOX(I)/
     1 RIOX(I)*2.*(PRO(I)-PIAVG))
    3 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SPPG(PIAVG)
C
C     SPPG IS A SPECIAL ROUTINE FOR MODECH=1 OR 3 AND NINCH=1.
C     CALCULATIONS USE AG=AIFU=AC AND TG=TIFU.  USE OCCURS WHEN NSPPG=1.
C     THE SOLUTION CORRESPONDS TO OXFULE(1)=0. AND MAY BE USED ONLY
C     WHEN NCEC71=0.  NAORW AND MGORLI ARE NOT REQUIRED.
C     THIS SPECIAL OPTION MAY PROVE USEFUL FOR STUDY OF SINGLE ZONE
C     CURVATURE EFFECTS WITH AIR OR LOW TEMPERATURE GAS FLOW WHERE
C     DETAILED CHAMBER PROPERTIES ARE NOT DESIRED.
C
      REAL MO,MOCH,MDTI,MDTIFU,MDTIOX,MWG
      DIMENSION TGA(15)
      COMMON     PRMT(6),DERY(4),AUX(16,4),Y(4),RW(10)
     1           ,POCL(7),ERWT(4),RMIN(7),MODECH,RMINSQ(7)
     2           ,PGCH(7),CLP(10),RLIM,MORDER,LOOP,MOCH(7),URMNSQ(7)
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSIO1(10),PSIO2(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      COMMON/CALC/HO,C,GAM,GAMSAV(200),K
      LOOPSP=0
      AG(1)=AC
      PI(1)=PG
      PIAVG=PG
      WIFU(1)=0.
      WIOX(1)=0.
      WG(1)=0.
    1 TG(1)=TOG(1)-WG(1)*WG(1)/2./CPG(1)
      RHOG(1)=PG/RGAS(1)/TG(1)
      WG(1)=MDTI(1)/RHOG(1)/AG(1)
      LOOPSP=LOOPSP+1
      TGA(LOOPSP)=TG(1)
      IF(LOOPSP.EQ.1) GOTO 1
      LOPSPM=LOOPSP-1
      IF(ABS(TGA(LOOPSP)-TGA(LOPSPM))/TGA(LOOPSP).LT..0005) GOTO 4
    2 CONTINUE
      IF(LOOPSP.EQ.15) WRITE(6,3)
    3 FORMAT(1X,30HLOOPSP=15,TG DID NOT CONVERGE.,/)
      IF(LOOPSP.LT.15) GOTO 1
      RETURN
    4 CONTINUE
      TIOX(1)=0.
      RIOX(1)=0.
      RHOIOX(1)=0.
      AIOX(1)=0.
```

```
      TIFU(1)=TG(1)
      WIFU(1)=WG(1)
      RIFU(1)=RGAS(1)
      RHOIFU(1)=RHOG(1)
      AIFU(1)=MDTIFU(1)/RHOIFU(1)/WIFU(1)
      IF(OXFULE(1).EQ.0.) RETURN
      TIOX(1)=TG(1)
      WIOX(1)=WG(1)
      RIOX(1)=RGAS(1)
      RHOIOX(1)=RHOG(1)
      AIOX(1)=MDTIOX(1)/RHOIOX(1)/WIOX(1)
      RETURN
      END
```

```
      SUBROUTINE EQULIB(NINCH)                                          EQUL  10
      REAL MO,MOCH,MDTI,MDTIFU,MDTIOX,MWG                               EQUL  20
      COMMON      PRMT(6),DERY(4),AUX(16,4),Y(4),RW(10)                 EQUL  30
     1            ,POCL(7),ERWT(4),RMIN(7),MODECH,RMINSQ(7)             EQUL  40
     2            ,PGCH(7),CLP(10),RLIM,MORDER,LOOP,MOCH(7),URMNSQ(7)   EQUL  50
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)   EQUL  60
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),EQUL  70
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSIO1(10),PSIO2(10),TWOPI,PSI12,MO EQUL  80
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),         EQUL  90
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),         EQUL 100
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,        EQUL 110
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),  EQUL 120
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NADRW,MTOTMX          EQUL 130
     5,CDINJ(20)                                                        EQUL 131
      COMMON/ZONE/KZONE,PGPSIA,OXFUL(20),LETOUT                         EQUL 140
      DIMENSION SAVTOG(20),SAVGAM(20),SAVMWG(20)                        EQUL 150
      NAMELIST/EQU/TOG,GAMG,MWG,NCHANG                                  EQUL 160
      DO 3 I=1,20                                                       EQUL 170
    3 OXFUL(I)=OXFULE(I)                                                EQUL 180
      NCHANG=0                                                          EQUL 190
      IF(MODECH.NE.3.AND.NCEC71.GT.1.AND.LOOP.GT.1) RETURN              EQUL 200
      IF(MODECH.NE.3.AND.NCEC71.GT.1.AND.LETOUT.EQ.71) RETURN           EQUL 210
      IF(MODECH.NE.3.AND.NCEC71.EQ.1.AND.MTOTMX.EQ.1.AND.LETOUT.EQ.71)  EQUL 220
     1   GOTO 4                                                         EQUL 230
      IF(MODECH.NE.3.AND.NCEC71.EQ.1.AND.LOOP.GT.1.AND.MTOTMX.EQ.1)     EQUL 240
     1   GOTO 4                                                         EQUL 250
      IF(MODECH.NE.3.AND.NCEC71.NE.0.AND.LOOP.GT.1)  RETURN             EQUL 260
      IF(MODECH.NE.3.AND.NCEC71.NE.0) CALL DRIVCE(NINCH,PG)             EQUL 270
      IF(MODECH.NE.3.AND.NCEC71.EQ.1.AND.LOOP.EQ.1.AND.MTOTMX.EQ.1)     EQUL 280
     1   GOTO 7                                                         EQUL 290
      IF(MODECH.NE.3.AND.NCEC71.NE.0) GOTO 2                            EQUL 300
C                                                                       EQUL 310
      IF(LOOP.GT.1.AND.MTOTMX.GT.0) GOTO 4                              EQUL 320
    6 CONTINUE                                                          EQUL 330
      IF(LOOP.GT.1) RETURN                                              EQUL 340
      IF(MODECH.EQ.3.AND.NCEC71.NE.0) CALL DRIVCE(NINCH,PG)             EQUL 350
      IF(MODECH.EQ.3.AND.NCEC71.EQ.1.AND.LOOP.EQ.1.AND.MTOTMX.EQ.1.AND. EQUL 360
     1LETOUT.EQ.71) GOTO 4                                              EQUL 370
      IF(MODECH.EQ.3.AND.NCEC71.EQ.1.AND.LOOP.EQ.1.AND.MTOTMX.EQ.1)     EQUL 380
     1 GOTO 7                                                           EQUL 390
      IF(MODECH.EQ.3.AND.NCEC71.NE.0) GOTO 2                            EQUL 400
C                                                                       EQUL 410
      READ(5,EQU)                                                       EQUL 420
      IF(LOOP.EQ.1.AND.NCHANG.GT.0) GOTO 7                              EQUL 430
      IF(LOOP.EQ.1.AND.NCHANG.EQ.0) GOTO 4                              EQUL 440
    9 CONTINUE                                                          EQUL 450
      IF(NCHANG.EQ.0) RETURN                                            EQUL 460
    2 CONTINUE                                                          EQUL 470
      IF(NCEC71.NE.0) RETURN                                            EQUL 480
      DO 1 I=1,NINCH                                                    EQUL 490
```

```
      RGAS(I)=UNIVR*32.174/MWG(I)
    1 CPG(I)=GAMG(I)*RGAS(I)/(GAMG(I)-1.)
      RETURN
    4 DO 5 I=1,NINCH
      TOG(I)=SAVTOG(I)
      GAMG(I)=SAVGAM(I)
      MWG(I)=SAVMWG(I)
      RGAS(I)=UNIVR*32.174/MWG(I)
    5 CPG(I)=GAMG(I)*RGAS(I)/(GAMG(I)-1.)
      IF(LOOP.EQ.1.AND.NCHANG.EQ.0) GOTO 9
      GOTO 6
    7 DO 8 I=1,NINCH
      SAVTOG(I)=TOG(I)
      SAVGAM(I)=GAMG(I)
    8 SAVMWG(I)=MWG(I)
      GOTO 9
      END
```

```
      SUBROUTINE LOAD1(NPT)
C     SEVERAL CONVERSION FACTORS ARE IN THIS SUBROUTINE TO CONVERT
C     DATA GENERATED BY MODIFIED PROGRAM CEC71 TO THE ENGLISH TECHNICAL
C     UNITS REQUIRED BY CHAMBER AND THROAT CALCULATIONS.
C     MODIFICATION OF THIS SUBROUTINE WOULD BE ESSENTIAL IN
C     A CONVERSION TO A DIFFERENT UNIT SYSTEM.
      REAL MWG,MO,MDTI,MDTIFU,MDTIOX
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS
      LOGICAL EQL,FROZ,AREA
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)
     2 ,VLM(13),TOTN(13)
      COMMON/PERF/PCP(22),VMOC(13),SPIM(13),VACI(13),SUBAR(13),SUPAR(13)
     1  ,APP(13),AEAT(13),CSTR,EQL,FROZ,SSO,AREA,AWT
      COMMON/ACSTAR/CSTAR(13)
      COMMON/ZONE/KZONE,PGPSIA,OXFUL(20),LETOUT
      COMMON/LOAD/CST(20),VAC1DI(20)
      COMMON/LOTAB/APRES(13),ASON(20,13),AMAC(20,13),AVEL(20,13),
     1ARHO(20,13),GAMR(20,13),TEMPE(20,13),WMR(20,13),VACSPI(20,13)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSI01(10),PSI02(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      DO 1 I=1,NPT
      APRES(I)=PG/APP(I)
      ASON(KZONE,I)=SONVEL(I)*3.281
      AMAC(KZONE,I)=VMOC(I)
      AVEL(KZONE,I)=ASON(KZONE,I)*VMOC(I)
      ARHO(KZONE,I)=1./VLM(I)*1.94179
      GAMR(KZONE,I)=GAMMAS(I)
      TEMPE(KZONE,I)=(TTT(I)+.5)*1.8
    1 WMR(KZONE,I)=WM(I)
      TOG(KZONE)= (TTT(1)+.5)*1.8
      RGAS(KZONE)=UNIVR*32.174/WM(1)
      GAMG(KZONE)=GAMMAS(1)
      CPG(KZONE)=GAMG(KZONE)*RGAS(KZONE)/(GAMG(KZONE)-1.)
      MWG(KZONE)=WM(1)
      RHOG(KZONE)=ARHO(KZONE,1)
      RETURN
      ENTRY LOAD2(NPT)
      DO 2 I=2,NPT
    2 VACSPI(KZONE,I)=VACI(I)
      VAC1DI(KZONE)=VACI(2)
      CST(KZONE)=CSTAR(2)
      RETURN
      END
```

```
      SUBROUTINE HPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)             HPCG1080
      DIMENSION PRMT(6),Y(NDIM),DERY(NDIM),AUX(16,NDIM)               HPCG1110
      N=1                                                             HPCG1120
      IHLF=0                                                          HPCG1130
      X=PRMT(1)                                                       HPCG1140
      H=PRMT(3)                                                       HPCG1150
      PRMT(5)=0.                                                      HPCG1160
      DO 1 I=1,NDIM                                                   HPCG1170
      AUX(16,I)=0.                                                    HPCG1180
      AUX(15,I)=DERY(I)                                               HPCG1190
    1 AUX(1,I)=Y(I)                                                   HPCG1200
      IF(H*(PRMT(2)-X))3,2,4                                          HPCG1210
    2 IHLF=12                                                         HPCG1240
      GOTO 4                                                          HPCG1250
    3 IHLF=13                                                         HPCG1260
    4 CALL FCT(X,Y,DERY)                                              HPCG1290
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)                              HPCG1320
      IF(PRMT(5))6,5,6                                                HPCG1330
    5 IF(IHLF)7,7,6                                                   HPCG1340
    6 RETURN                                                          HPCG1350
    7 DO 8 I=1,NDIM                                                   HPCG1360
    8 AUX(8,I)=DERY(I)                                                HPCG1370
      ISW=1                                                           HPCG1400
      GOTO 100                                                        HPCG1410
    9 X=X+H                                                           HPCG1430
      DO 10 I=1,NDIM                                                  HPCG1440
   10 AUX(2,I)=Y(I)                                                   HPCG1450
   11 IHLF=IHLF+1                                                     HPCG1480
      X=X-H                                                           HPCG1490
      DO 12 I=1,NDIM                                                  HPCG1500
   12 AUX(4,I)=AUX(2,I)                                               HPCG1510
      H=.5*H                                                          HPCG1520
      N=1                                                             HPCG1530
      ISW=2                                                           HPCG1540
      GOTO 100                                                        HPCG1550
   13 X=X+H                                                           HPCG1570
      CALL FCT(X,Y,DERY)                                              HPCG1580
      IF(Y(1).LE.0.) RETURN                                           HPCG158A
      N=2                                                             HPCG1590
      DO 14 I=1,NDIM                                                  HPCG1600
      AUX(2,I)=Y(I)                                                   HPCG1610
   14 AUX(9,I)=DERY(I)                                                HPCG1620
      ISW=3                                                           HPCG1630
      GOTO 100                                                        HPCG1640
   15 DELT=0.                                                         HPCG1670
      DO 16 I=1,NDIM                                                  HPCG1680
   16 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))                          HPCG1690
      DELT=.06666667*DELT                                             HPCG1700
      IF(DELT-PRMT(4))19,19,17                                        HPCG1710
   17 IF(IHLF-10)11,18,18                                             HPCG1720
```

```
   18 IHLF=11                                                          HPCG1750
      X=X+H                                                            HPCG1760
      GOTO 4                                                           HPCG1770
   19 X=X+H                                                            HPCG1800
      CALL FCT(X,Y,DERY)                                               HPCG1810
      IF(Y(1).LE.0.) RETURN                                            HPCG181A
      DO 20 I=1,NDIM                                                   HPCG1820
      AUX(3,I)=Y(I)                                                    HPCG1830
   20 AUX(10,I)=DERY(I)                                                HPCG1840
      N=3                                                              HPCG1850
      ISW=4                                                            HPCG1860
      GOTO 100                                                         HPCG1870
   21 N=1                                                              HPCG1890
      X=X+H                                                            HPCG1900
      CALL FCT(X,Y,DERY)                                               HPCG1910
      IF(Y(1).LE.0.) RETURN                                            HPCG191A
      X=PRMT(1)                                                        HPCG1920
      DO 22 I=1,NDIM                                                   HPCG1930
      AUX(11,I)=DERY(I)                                                HPCG1940
   22 Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.7916667*AUX(9,I)                 HPCG1950
     1-.2083333*AUX(10,I)+.04166667*DERY(I))                           HPCG1960
   23 X=X+H                                                            HPCG1970
      N=N+1                                                            HPCG1980
      CALL FCT(X,Y,DERY)                                               HPCG1990
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)                               HPCG2000
      IF(PRMT(5))6,24,6                                                HPCG2010
   24 IF(N-4)25,200,200                                                HPCG2020
   25 DO 26 I=1,NDIM                                                   HPCG2030
      AUX(N,I)=Y(I)                                                    HPCG2040
   26 AUX(N+7,I)=DERY(I)                                               HPCG2050
      IF(N-3)27,29,200                                                 HPCG2060
   27 DO 28 I=1,NDIM                                                   HPCG2080
      DELT=AUX(9,I)+AUX(9,I)                                           HPCG2090
      DELT=DELT+DELT                                                   HPCG2100
   28 Y(I)=AUX(1,I)+.3333333*H*(AUX(8,I)+DELT+AUX(10,I))               HPCG2110
      GOTO 23                                                          HPCG2120
   29 DO 30 I=1,NDIM                                                   HPCG2140
      DELT=AUX(9,I)+AUX(10,I)                                          HPCG2150
      DELT=DELT+DELT+DELT                                              HPCG2160
   30 Y(I)=AUX(1,I)+.375*H*(AUX(8,I)+DELT+AUX(11,I))                   HPCG2170
      GOTO 23                                                          HPCG2180
  100 DO 101 I=1,NDIM                                                  HPCG2230
      Z=H*AUX(N+7,I)                                                   HPCG2240
      AUX(5,I)=Z                                                       HPCG2250
  101 Y(I)=AUX(N,I)+.4*Z                                               HPCG2260
      Z=X+.4*H                                                         HPCG2290
      CALL FCT(Z,Y,DERY)                                               HPCG2300
      IF(Y(1).LE.0.) RETURN                                            HPCG230A
      DO 102 I=1,NDIM                                                  HPCG2310
      Z=H*DERY(I)                                                      HPCG2320
```

```
      AUX(6,I)=Z                                                        HPCG2330
  102 Y(I)=AUX(N,I)+.2969776*AUX(5,I)+.1587596*Z                        HPCG2340
      Z=X+.4557372*H                                                    HPCG2360
      CALL FCT(Z,Y,DERY)                                                HPCG2370
      IF(Y(1).LE.0.) RETURN                                             HPCG237A
      DO 103 I=1,NDIM                                                   HPCG2380
      Z=H*DERY(I)                                                       HPCG2390
      AUX(7,I)=Z                                                        HPCG2400
  103 Y(I)=AUX(N,I)+.2181004*AUX(5,I)-3.050965*AUX(6,I)+3.832865*Z      HPCG2410
      Z=X+H                                                             HPCG2430
      CALL FCT(Z,Y,DERY)                                                HPCG2440
      IF(Y(1).LE.0.) RETURN                                             HPCG244A
      DO 104 I=1,NDIM                                                   HPCG2450
  1040Y(I)=AUX(N,I)+.1747603*AUX(5,I)-.5514807*AUX(6,I)                 HPCG2460
     1+1.205536*AUX(7,I)+.1711848*H*DERY(I)                             HPCG2470
      GOTO(9,13,15,21),ISW                                              HPCG2480
  200 ISTEP=3                                                           HPCG2540
  201 IF(N-8)204,202,204                                                HPCG2550
  202 DO 203 N=2,7                                                      HPCG2580
      DO 203 I=1,NDIM                                                   HPCG2590
      AUX(N-1,I)=AUX(N,I)                                               HPCG2600
  203 AUX(N+6,I)=AUX(N+7,I)                                             HPCG2610
      N=7                                                               HPCG2620
  204 N=N+1                                                             HPCG2650
      DO 205 I=1,NDIM                                                   HPCG2680
      AUX(N-1,I)=Y(I)                                                   HPCG2690
  205 AUX(N+6,I)=DERY(I)                                                HPCG2700
      X=X+H                                                             HPCG2710
  206 ISTEP=ISTEP+1                                                     HPCG2720
      DO 207 I=1,NDIM                                                   HPCG2730
     0DELT=AUX(N-4,I)+1.333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+     HPCG2740
     1AUX(N+4,I)+AUX(N+4,I))                                            HPCG2750
      Y(I)=DELT-.9256198*AUX(16,I)                                      HPCG2760
  207 AUX(16,I)=DELT                                                    HPCG2770
      CALL FCT(X,Y,DERY)                                                HPCG2810
      IF(Y(1).LE.0.) RETURN                                             HPCG281A
      DO 208 I=1,NDIM                                                   HPCG2840
     0DELT=.125*(9.*AUX(N-1,I)-AUX(N-3,I)+3.*H*(DERY(I)+AUX(N+6,I)+     HPCG2850
     1AUX(N+6,I)-AUX(N+5,I)))                                           HPCG2860
      AUX(16,I)=AUX(16,I)-DELT                                          HPCG2870
  208 Y(I)=DELT+.07438017*AUX(16,I)                                     HPCG2880
      DELT=0.                                                           HPCG2910
      DO 209 I=1,NDIM                                                   HPCG2920
  209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))                                HPCG2930
      IF(DELT-PRMT(4))210,222,222                                       HPCG2940
  210 CALL FCT(X,Y,DERY)                                                HPCG2970
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)                                HPCG2980
      IF(PRMT(5))212,211,212                                            HPCG2990
  211 IF(IHLF-11)213,212,212                                            HPCG3000
  212 RETURN                                                            HPCG3010
```

```
  213 IF(H*(X-PRMT(2)))214,212,212                                      HPCG3020
  214 IF(ABS(X-PRMT(2))-.1*ABS(H))212,215,215                           HPCG3030
  215 IF(DELT-.02*PRMT(4))216,216,201                                   HPCG3040
  216 IF(IHLF)201,201,217                                               HPCG3090
  217 IF(N-7)201,218,218                                                HPCG3100
  218 IF(ISTEP-4)201,219,219                                            HPCG3110
  219 IMOD=ISTEP/2                                                      HPCG3120
      IF(ISTEP-IMOD-IMOD)201,220,201                                    HPCG3130
  220 H=H+H                                                             HPCG3140
      IHLF=IHLF-1                                                       HPCG3150
      ISTEP=0                                                           HPCG3160
      DO 221 I=1,NDIM                                                   HPCG3170
      AUX(N-1,I)=AUX(N-2,I)                                             HPCG3180
      AUX(N-2,I)=AUX(N-4,I)                                             HPCG3190
      AUX(N-3,I)=AUX(N-6,I)                                             HPCG3200
      AUX(N+6,I)=AUX(N+5,I)                                             HPCG3210
      AUX(N+5,I)=AUX(N+3,I)                                             HPCG3220
      AUX(N+4,I)=AUX(N+1,I)                                             HPCG3230
      DELT=AUX(N+6,I)+AUX(N+5,I)                                        HPCG3240
      DELT=DELT+DELT+DELT                                               HPCG3250
  2210AUX(16,I)=8.962963*(Y(I)-AUX(N-3,I))-3.361111*H*(DERY(I)+DELT     HPCG3260
     1+AUX(N+4,I))                                                      HPCG3270
      GOTO 201                                                          HPCG3280
  222 IHLF=IHLF+1                                                       HPCG3320
      IF(IHLF-10)223,223,210                                            HPCG3330
  223 H=.5*H                                                            HPCG3340
      ISTEP=0                                                           HPCG3350
      DO 224 I=1,NDIM                                                   HPCG3360
     0Y(I)=.00390625*(80.*AUX(N-1,I)+135.*AUX(N-2,I)+40.*AUX(N-3,I)+    HPCG3370
     1AUX(N-4,I))-.1171875*(AUX(N+6,I)-6.*AUX(N+5,I)-AUX(N+4,I))*H      HPCG3380
     0AUX(N-4,I)=.00390625*(12.*AUX(N-1,I)+135.*AUX(N-2,I)+             HPCG3390
     1108.*AUX(N-3,I)+AUX(N-4,I))-.0234375*(AUX(N+6,I)+18.*AUX(N+5,I)-  HPCG3400
     29.*AUX(N+4,I))*H                                                  HPCG3410
      AUX(N-3,I)=AUX(N-2,I)                                             HPCG3420
  224 AUX(N+4,I)=AUX(N+5,I)                                             HPCG3430
      X=X-H                                                             HPCG3440
      DELT=X-(H+H)                                                      HPCG3450
      CALL FCT(DELT,Y,DERY)                                             HPCG3460
      IF(Y(1).LE.0.) RETURN                                             HPCG346A
      DO 225 I=1,NDIM                                                   HPCG3470
      AUX(N-2,I)=Y(I)                                                   HPCG3480
      AUX(N+5,I)=DERY(I)                                                HPCG3490
  225 Y(I)=AUX(N-4,I)                                                   HPCG3500
      DELT=DELT-(H+H)                                                   HPCG3510
      CALL FCT(DELT,Y,DERY)                                             HPCG3520
      IF(Y(1).LE.0.) RETURN                                             HPCG352A
      DO 226 I=1,NDIM                                                   HPCG3530
      DELT=AUX(N+5,I)+AUX(N+4,I)                                        HPCG3540
      DELT=DELT+DELT+DELT                                               HPCG3550
     0AUX(16,I)=8.962963*(AUX(N-1,I)-Y(I))-3.361111*H*(AUX(N+6,I)+DELT  HPCG3560
```

```
    1+DERY(I))                                                    HPCG3570
226 AUX(N+3,I)=DERY(I)                                            HPCG3580
    GOTO 206                                                      HPCG3590
    END                                                           HPCG3600
```

```
      SUBROUTINE FCT(R,Y,DERY)
      REAL MO,MOCH,MDTI,MDTIFU,MDTIOX,NEXPSI,MWG
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSIO1(10),PSIO2(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      COMMON/CALC/HO,C,GAM,GAMSAV(200),K
      DIMENSION Y(NDIM),DERY(NDIM)
      NCHA1P=NCHA1+1
      NCHA2P=NCHA2+1
      IF(MTOTMX.GT.0)NCHA1P=2
      IF(J.GE.2)GOTO 3
      IF(J.GE.1)GOTO 2
      DO 1 I=1,NDIM
    1 DERY(I)=0.
      K=1
      GAM=GAMG(K)
      IF(NCEC71.LT.2) GOTO 12
      Y1=Y(1)
      IF(NCEC71.GT.1) CALL TAB(Y1,DENS,VELO,TEMP,EMACH)
      T(1)=TEMP
      YPSAV(1,4)=EMACH
      YSAVE(1,3)=VELO
   12 CONTINUE
      RETURN
    2 K=1
      GOTO 7
    3 JM1=J-1
      NEXPSI=YSAVE(J,2)+(R-XSAVE(J))*(YSAVE(J,2)-YSAVE(JM1,2))/
     1          (XSAVE(J)-XSAVE(JM1))
      K=1
      DO 4 I=2,NCHA1P
      IF(NEXPSI.LE.PSIO1(I))GOTO 7
    4 K=K+1
    5 CONTINUE
      K=K-1
      IF(NCHA2.EQ.0)GOTO 7
      IF(PSI12.GE.1.)GOTO 7
      K=K+1
      DO 6 I=2,NCHA2P
      IF(NEXPSI.LE.PSIO2(I))GOTO 7
    6 K=K+1
      K=K-1
    7 CONTINUE
      IF(NCEC71.LT.2) GOTO 14
```

```
      Y1=Y(1)
      IF(NCEC71.GT.1) CALL  TAB(Y1,DENS,VELO,TEMP,EMACH)
      JP1=J+1
      T(JP1)=TEMP
      YPSAV(JP1,4)=EMACH
      YSAVE(JP1,3)=VELO
      IF(NCEC71.GT.1) GOTO 8
   14 HO=CPG(K)*TOG(K)
      C=POG(K)/(POG(K)/RGAS(K)/TOG(K))**GAMG(K)
      GAM=GAMG(K)
      G2=1./GAM
      G3=(GAM-1.)/GAM
    8 G1=RCWALL*((RT*(1.+RCWALL/R)-R)/RCWALL)**1.5
      IF(Y(1).LE.O.) WRITE(6,9) R,YSAVE(1,1)
    9 FORMAT(26H PRESSURE IS LE ZERO AT R=,E15.6,10X,4HPCL=,E15.6)
      IF(Y(1).LE.O.) DERY(1)=0.
      IF(Y(1).LE.O.) DERY(2)=0.
      IF(Y(1).LE.O.) RETURN
      IF(NCEC71.LT.2) GOTO 10
      DERY(1)=-DENS*VELO*VELO/G1
      DERY(2)=TWOPI*R*DENS*VELO/MO
      GOTO 11
   10 DERY(1)=-(Y(1)/C)**G2*(2.*(HO-Y(1)**G3*C**G2/G3))/G1
      DERY(2)=(Y(1)/C)**G2*TWOPI*R*SQRT(2.*(HO-Y(1)**G3*C**G2/G3))/MO
   11 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE OUTP(X,Y,DERY,IHL,NDI,PRMT)
C     YSAVE(J,1) IS PRESSURE.
C     YSAVE(J,2) IS PSI.
C     YSAVE(J,3) IS VELOCITY.
C     YPSAV(J,4) IS THE LOCAL MACH NUMBER.
      REAL MDTI,MDTIFU,MDTIOX,MO,MWG
      DIMENSION PRMT(6),Y(NDI),DERY(NDI)
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71,T(200)
      COMMON/TWO/YSAVE(200,4),XSAVE(200),J,CPG(20),POG(20),YPSAV(200,4),
     1RT,RCWALL,GAMG(20),NCHA1,NCHA2,PSI01(10),PSI02(10),TWOPI,PSI12,MO
      COMMON/THR/OXFULE(20),PG,TOG(20),RGAS(20),UNIVR,WIFU(20),
     1 AIFU(20),RHOIFU(20),RIFU(20),TIFU(20),WIOX(20),AIOX(20),
     2 RHOIOX(20),RIOX(20),TIOX(20),AIPCTF(20),ZCH,PI(20),DIACH,
     3 TG(20),AG(20),WG(20),RHOG(20),RAD(20),MDTI(20),PRO(20),PRF(20),
     4 PCTMO(20),MDTIFU(20),MDTIOX(20),MWG(20),AC,NAORW,MTOTMX
     5,CDINJ(20)
      COMMON/CALC/HO,C,GAM,GAMSAV(200),K
      IHLF=IHL
      NDIM=NDI
      IF(Y(1).LE.0.) PRMT(5)=1.
      IF(Y(1).LE.0.) RETURN
      J=J+1
    1 XSAVE(J)=X
      DO 2 I=1,NDIM
      YSAVE(J,I)=Y(I)
    2 YPSAV(J,I)=DERY(I)
      IF(NCEC71.GT.1) GOTO 4
      YPSAV(J,3)=1./(YSAVE(J,1)/POG(K))**(1.-1./GAM)
      T(J)=TOG(K)/YPSAV(J,3)
      A=SQRT(TOG(K)*GAM*RGAS(K)/YPSAV(J,3))
      YPSAV(J,4)=SQRT(2.*(YPSAV(J,3)-1.)/(GAM-1.))
      YSAVE(J,3)=A*YPSAV(J,4)
    4 IF(YSAVE(J,2).GE.PRMT(6)) PRMT(5)=1.0
      GAMSAV(J)=GAM
      RETURN
      END
```

```
      SUBROUTINE TAB(Y1,DENS,VELO,TEMP,EMACH)
      COMMON/LOTAB/APRES(13),ASON(20,13),AMAC(20,13),AVEL(20,13),
     1ARHO(20,13),GAMR(20,13),TEMPE(20,13),WMR(20,13),VACSPI(20,13)
      COMMON/CALC/HO,C,GAM,GAMSAV(200),K
      IF(Y1.GE.APRES(3)) GOTO 1
      IF(Y1.LE.APRES(12)) GOTO 2
      DO 4 I=4,12
      M1=I
      M2=I-1
      IF(Y1.GE.APRES(I)) GOTO 3
    4 CONTINUE
    3 SLO=(Y1-APRES(M1))/(APRES(M2)-APRES(M1))
      DENS=ARHO(K,M1)+SLO*(ARHO(K,M2)-ARHO(K,M1))
      VELO=AVEL(K,M1)+SLO*(AVEL(K,M2)-AVEL(K,M1))
      TEMP=TEMPE(K,M1)+SLO*(TEMPE(K,M2)-TEMPE(K,M1))
      EMACH=AMAC(K,M1)+SLO*(AMAC(K,M2)-AMAC(K,M1))
      GAM=GAMR(K,M1)+SLO*(GAMR(K,M2)-GAMR(K,M1))
      RETURN
    1 M1=3
      M2=4
      GOTO 3
    2 M1=12
      M2=11
      GOTO 3
      END
```

```
      SUBROUTINE AITKEN(X,Y,N,K,XB,YB)                                  AITK  10
      DIMENSION X(N),Y(N),XX(11),YY(11)                                 AITK  20
      K1=K+1                                                            AITK  30
      IF(X(N)-X(1))10,1,1                                               AITK  40
    1 IF(XB-X(1))2,2,3                                                  AITK  50
    2 LL=0                                                              AITK  60
      GOTO 19                                                           AITK  70
    3 IF(X(N)-XB)4,4,5                                                  AITK  80
    4 LL=N-K1                                                           AITK  90
      GOTO 19                                                           AITK 100
    5 LL=1                                                              AITK 110
      LU=N                                                              AITK 120
    6 LLP1=LL+1                                                         AITK 130
      IF(LU-LLP1)17,17,7                                                AITK 140
    7 LI=(LL+LU)/2                                                      AITK 150
      IF(X(LI)-XB)8,8,9                                                 AITK 160
    8 LL=LI                                                             AITK 170
      GOTO 6                                                            AITK 180
    9 LU=LI                                                             AITK 190
      GOTO 6                                                            AITK 200
   10 IF(XB-X(1))11,2,2                                                 AITK 210
   11 IF(X(N)-XB)12,4,4                                                 AITK 220
   12 LL=1                                                              AITK 230
      LU=N                                                              AITK 240
   13 LLP1=LL+1                                                         AITK 250
      IF(LU-LLP1)17,17,14                                               AITK 260
   14 LI=(LL+LU)/2                                                      AITK 270
      IF(X(LI)-XB)15,16,16                                              AITK 280
   15 LU=LI                                                             AITK 290
      GOTO 13                                                           AITK 300
   16 LL=LI                                                             AITK 310
      GOTO 13                                                           AITK 320
   17 LL=LL-(K1+1)/2                                                    AITK 330
      IF(LL)2,19,18                                                     AITK 340
   18 IF(LL+K1-N)19,19,4                                                AITK 350
   19 DO 20 I=1,K1                                                      AITK 360
      I1=LL+I                                                           AITK 370
      XX(I)=X(I1)-XB                                                    AITK 380
   20 YY(I)=Y(I1)                                                       AITK 390
      DO 21 I=1,K                                                       AITK 400
      DO 21 J=I,K                                                       AITK 410
   21 YY(J+1)=(1./(XX(J+1)-XX(I)))*(YY(I)*XX(J+1)-YY(J+1)*XX(I))        AITK 420
      YB=YY(K1)                                                         AITK 430
      RETURN                                                            AITK 440
      END                                                               AITK 450
```

```
      SUBROUTINE UPORD(RW,CLP,NCLP,MORDER)                              UPOR  10
      DIMENSION RW(NCLP),CLP(NCLP)                                      UPOR  20
      DO 12 KP=1,NCLP                                                   UPOR  30
      RCHK=RW(KP)                                                       UPOR  40
      PCHK=CLP(KP)                                                      UPOR  50
      DO 11 LP=KP,NCLP                                                  UPOR  60
      IF(MORDER)8,9,8                                                   UPOR  70
    8 IF(RW(LP)-RCHK)10,11,11                                           UPOR  80
    9 IF(CLP(LP)-PCHK)10,11,11                                          UPOR  90
   10 STOR=RCHK                                                         UPOR 100
      STOP=PCHK                                                         UPOR 110
      RCHK=RW(LP)                                                       UPOR 120
      PCHK=CLP(LP)                                                      UPOR 130
      RW(LP)=STOR                                                       UPOR 140
      CLP(LP)=STOP                                                      UPOR 150
   11 CONTINUE                                                          UPOR 160
      CLP(KP)=PCHK                                                      UPOR 170
   12 RW(KP)=RCHK                                                       UPOR 180
      RETURN                                                            UPOR 190
      END                                                               UPOR 200
```

```
      SUBROUTINE PLYCF (X,Y,NEP,A,NORD,NSTOPD)
      DOUBLE PRECISION XC,P,V,DP,B(10,11),BS(10,10),A(11)
      DIMENSION X(NEP),Y(NEP)
      JA=6
      INORM=1
      NSTOPD=1
       NGORD=0
      IF(NORD-NGORD)2,2,1
    1 NGORD=NORD
    2 NGRD1=NGORD+1
    3 N=NORD
      N1=N+1
      N2=N+2
      IF(NEP.GE.N1)GOTO 6
    4 WRITE(JA,5)
    5 FORMAT(/43H NOT SUFFICIENT DATA POINTS - CASE OMITTED.  )
      NSTOPD=2
      GOTO 34
    6 DO 7    I=1,N2
    7 B(I,N2)=0.
    8 DO 9    J=1,NGRD1
      DO 9    I=1,NGRD1
    9 BS(J,I)=0.
      DO 13   K=1,NEP
      XC=1.
      DO 13   J=1,NGRD1
      IF(J.EQ.1)GOTO 11
   10 BS(J,NGRD1)=BS(J,NGRD1)+XC
      XC=XC*X(K)
      GOTO 13
   11 DO 12   I=1,NGRD1
      BS(1,I)=BS(1,I)+XC
   12 XC=XC*X(K)
   13 CONTINUE
   14 DO 15   J=2,NGRD1
      DO 15   I=1,NGORD
   15 BS(J,I)=BS(J-1,I+1)
      DO 16   I=1,NEP
      XC=1.
      DO 16   J=1,N1
      B(J,N2)=B(J,N2)+XC*Y(I)
   16 XC=XC*X(I)
   17 DO 18   I=1,N1
      DO 18   J=1,N1
   18 B(I,J)=BS(I,J)
      DO 27   I=1,N1
      IF(B(I,I).NE.0.)GOTO 22
   19 NSTOPD=2
   20 WRITE(JA,21)
   21 FORMAT(/,25H MATRIX SINGULAR NO SOLN.)
```

```
      NSTOPD=2                                                          PLYC 510
      GOTO 34                                                           PLYC 520
   22 P=1./B(I,I)                                                       PLYC 530
      DO 23  J=1,N2                                                     PLYC 540
   23 B(I,J)=P*B(I,J)                                                   PLYC 550
      DO 26  K=1,N1                                                     PLYC 560
      NUT=I-K                                                           PLYC 570
      IF(NUT.EQ.0)GOTO 26                                               PLYC 580
   24 Q=B(K,I)                                                          PLYC 590
      DO 25  J=1,N2                                                     PLYC 600
   25 B(K,J)=B(K,J)-Q*B(I,J)                                            PLYC 610
   26 CONTINUE                                                          PLYC 620
   27 CONTINUE                                                          PLYC 630
   28 DO 29  I=1,N1                                                     PLYC 640
   29 A(I)=B(I,N2)                                                      PLYC 650
   30 DEVMAX=0.                                                         PLYC 660
      RMS=0.                                                            PLYC 670
      DO 33  I=1,NEP                                                    PLYC 680
      V=0.                                                              PLYC 690
      DO 31  J=1,N1                                                     PLYC 700
      K1=N1-J+1                                                         PLYC 710
   31 V=V*X(I)+A(K1)                                                    PLYC 720
      DEV=V-Y(I)                                                        PLYC 730
      IF(ABS(DEV).LT.ABS(DEVMAX))GOTO 33                                PLYC 740
   32 IMAX=I                                                            PLYC 750
      DEVMAX=DEV                                                        PLYC 760
   33 RMS=RMS+DEV*DEV                                                   PLYC 770
      H=NEP                                                             PLYC 780
      RMS=SQRT(RMS/H)                                                   PLYC 790
   34 CONTINUE                                                          PLYC 800
      RETURN                                                            PLYC 810
      END                                                               PLYC 820
```

```
      SUBROUTINE DRIVCE(NINCH,PG)
C   MAIN PROGRAM                                                         MAIN    1
C     COMMON BLK. MISC IN MAIN,SAVE,&EQLBRM CONTAIN SIZE--OTHERS DO NOT  MAINA   1
C                                                                        MAIN    2
      DOUBLE PRECISION G,X                                               MAIN    3
C                                                                        MAIN    4
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR      MAIN    5
C     IBM 360 MACHINES ONLY                                              MAIN    6
C                                                                        MAIN    7
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS                  MAIN    8
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN                            MAIN    9
C                                                                        MAIN   10
      REAL MIX(15)                                                       MAIN   11
      INTEGER DATA, OMIT, ENSERT, REAC, BLANK, THRM, END,SUB             MAIN   12
      LOGICAL SHOCK,MMHG,UV,IC,DETN,SIUNIT,EUNITS,NSQM,CALCH             MAIN   13
      LOGICAL HP,SP,TP,NEWR,IONS,MOLES,FROZ,EQL,PSIA,RKT,VOL,TV,SV       MAIN   14
      LOGICAL FA,OF,ERATIO,FPCT,OTTO                                     MAIN   15
C                                                                        MAIN   16
      DIMENSION OMIT(3,3),NCD(4),ENSERT(3,3),RHO(26),LVP(2),VM(2),VL(26)MAIN   17
     1,DAT(22)                                                           MAIN   18
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)        MAIN   19
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)   MAIN   20
     2 ,VLM(13),TOTN(13)                                                 MAIN   21
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)    MAIN   22
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)     MAIN   23
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)   MAIN   24
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)MAIN   25
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)          MAIN   26
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)MAIN   27
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE,SIZE       MAIN   28
      COMMON /DOUBLE/ G(20,21), X(20)                                    MAIN   29
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM   MAIN   30
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ MAIN   31
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHMAIN   32
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                     MAIN   33
      COMMON/PERF/PCP(22),VMOC(13),SPIM(13),VACI(13),SUBAR(13),SUPAR(13)MAIN   34
     1   ,APP(13),AEAT(13),CSTR,EQL,FROZ,SSO,AREA,AWT                    MAIN   35
      COMMON/ZONE/KZONE,PGPSIA,OXFUL( 20),LETOUT                         MAINA  35
C                                                                        MAIN   36
      EQUIVALENCE (OMIT,ENLN),(ENSERT,DELN),(OXF,MIX)                    MAIN   37
     1   ,(OF,OXFL),(RHO,P,VL),(SO,SO),(OTTO,CPCVFR),(DATA,DAT)          MAIN   38
C                                                                        MAIN   39
      DATA MIT/4HOMIT/,BLANK/1H /, PSIA/4HPSIA/,REAC/4HREAC/,IZ/2HOO/    MAIN   40
     1   ,NMLT/4HNAME/,IE/1HE/,INSERT/4HINSE/,THRM/4HTHER/,END/3HEND/    MAIN   41
     2 ,GAS/1HG/,LAST/4HLAST/                                            MAIN   42
C                                                                        MAIN   43
C     NAMELIST/INPT2/KASE,T,P,PSIA,MMHG,NSQM,V,RHO,ERATIO,OF,FPCT,FA,    MAIN/  44
C    1MIX,TP,HP,SP,TV,UV,SV,RKT,SHOCK,DETN,OTTO,CR,SO,SO,IONS,IDEBUG,    MAIN/  45
C    2TRACE,SIUNIT,EUNITS                                                MAIN/  46
      PGPSIA=PG/144.                                                     MAINA  46
```

```
      IF(LETOUT.EQ.71) RETURN                                         MAINB  46
C                                                                     MAIN   47
      NEWR = .FALSE.                                                  MAIN   48
C                                                                     MAIN   49
      NPT=0                                                           MAINA  49
      KZONE=0                                                         MAINB  49
    1 KZONE=KZONE+1                                                   MAINC  49
      IF(KZONE.GT.NINCH)RETURN                                        MAIND  49
C   1 WRITE(6,400)                                                    MAIN/  50
  400 FORMAT(1H1)                                                     MAIN   51
      RR = 8314.3                                                     MAIN   52
      R = RR/4184.                                                    MAIN   53
  203 READ (5,204) (DATA(I),I=1,15)                                   MAIN   54
  204 FORMAT(5(3A4,3X))                                               MAIN   55
C     WRITE (6,2045)(DATA(I),I=1,15)                                  MAIN/  56
 2045 FORMAT(1X,5(3A4,3X))                                            MAIN   57
      IF(DATA(1).EQ.THRM)  GO TO 90                                   MAIN   58
      IF(DATA(1).EQ.REAC)  GO TO 11                                   MAIN   59
      IF (DATA(1).EQ.MIT) GO TO 205                                   MAIN   60
      IF (DATA(1).EQ.INSERT) GO TO 180                                MAIN   61
      IF(DATA(1).EQ.NMLT)  GO TO 210                                  MAIN   62
      IF(DATA(1).EQ.BLANK) GO TO 203                                  MAIN   63
 1023 WRITE(6,1024)                                                   MAIN   64
 1024 FORMAT(40HOERROR IN ABOVE CARD.  CONTENTS IGNORED. )            MAIN   65
      GO TO 203                                                       MAIN   66
   11 NSERT = 0                                                       MAIN   67
      MOLES = .FALSE.                                                 MAIN   68
      CALL REACT                                                      MAIN   69
      IF(NLM.EQ.0)  WRITE(6,52)                                       MAIN   70
   52 FORMAT(24HOERROR IN REACTANT CARDS)                             MAIN   71
      CALCH = .FALSE.                                                 MAIN   72
      DO 755 N=1,NREAC                                                MAIN   73
      IF(NAME(N,5).EQ.IZ)  CALCH=.TRUE.                               MAIN   74
  755 CONTINUE                                                        MAIN   75
      GO TO 203                                                       MAIN   76
C                                                                     MAIN   77
C     READ THERMO DATA FROM CARDS AND STORE ON  TAPE 4                MAIN   78
C                                                                     MAIN   79
   90 NEWR = .TRUE.                                                   MAIN   80
      REWIND 4                                                        MAIN   81
      READ(5,5) TLOW,TMID,THIGH                                       MAIN   82
    5 FORMAT (3F10.3)                                                 MAIN   83
      WRITE (4,5) TLOW,TMID,THIGH                                     MAIN   84
   97 READ (5,10)(DAT(I),I=1,16),NCD(1)                               MAIN   85
   10 FORMAT(3A4,6X,2A3,4(A2,F3.0),A1,2F10.3,I15)                     MAIN   86
      IF(DATA(1).EQ.BLANK) DATA(1)=END                                MAIN   87
      WRITE (4,10)(DAT(I),I=1,16)                                     MAIN   88
      IF(DATA(1).NE.END)  GO TO 18                                    MAIN   89
      WRITE(4,10)  LAST                                               MAIN   90
      GO TO 203                                                       MAIN   91
```

```
   18 READ(5,20)(DAT(I),I=1,5),NCD(2),(DAT(J),J=6,10),NCD(3),(DAT(K),
     1K=11,14),NCD(4)
   20 FORMAT(5E15.8,I5/5E15.8,I5/4E15.8,I20)
      WRITE (4,21)(DAT(I),I=1,14)
   21 FORMAT(5E15.8/5E15.8/4E15.8)
      DO 25  I=1,4
      IF(NCD(I).EQ.I)  GO TO 25
      WRITE(6,22) (DATA(J),J=1,3)
   22 FORMAT(28HOERROR IN ORDER OF CARDS FOR ,3A4)
   25 CONTINUE
      GO TO 97
C
C    CHECK INSERT CARDS
C
  180 DO 185 I=4,15,3
      IF (DATA(I).EQ.BLANK) GO TO 185
      NSERT = NSERT+1
      ENSERT(1,NSERT) = DATA(I)
      ENSERT(2,NSERT) = DATA(I+1)
      ENSERT(3,NSERT) = DATA(I+2)
  185 CONTINUE
      GO TO 203
C
C     CHECK OMIT CARDS
C
  205 DO 208  I=4,15,3
      IF(DATA(I).EQ.BLANK) GO TO 208
      NOMIT = NOMIT+1
      OMIT(1,NOMIT) = DATA(I)
      OMIT(2,NOMIT) = DATA(I+1)
      OMIT(3,NOMIT) = DATA(I+2)
  208 CONTINUE
      NEWR= .TRUE.
      REWIND 4
      GO TO 203
C
C     BEGIN NAMELIST INPT2
C
  210 DO 300 I=1,26
      P(I)= 0.
      T(I)= 0.
      V(I) = 0.
  300 CONTINUE
      TRACE = 0.
      SO = 0.
      V1 = 0.
      V2 = 0.
      CR = 0.
      RHOP = 0.
      KASE= 0
```

```
      TP = .FALSE.                                                      MAIN 142
      HP=.FALSE.                                                        MAIN 143
      SP=.FALSE.                                                        MAIN 144
      TV = .FALSE.                                                      MAIN 145
      UV = .FALSE.                                                      MAIN 146
      SV = .FALSE.                                                      MAIN 147
      OTTO = .FALSE.                                                    MAIN 148
      RKT = .TRUE.                                                      MAIN/149
      SHOCK = .FALSE.                                                   MAIN 150
      DETN = .FALSE.                                                    MAIN 151
      VOL = .FALSE.                                                     MAIN 152
      MMHG   = .FALSE.                                                  MAIN 153
      PSIA = .TRUE.                                                     MAIN/154
      NSQM = .FALSE.                                                    MAIN 155
      SIUNIT = .FALSE.                                                  MAIN 156
      EUNITS = .FALSE.                                                  MAIN 157
      IONS = .FALSE.                                                    MAIN 158
      IDEBUG = 0                                                        MAIN 159
      FA= .FALSE.                                                       MAIN 160
      OF = .TRUE.                                                       MAIN/161
      ERATIO = .FALSE.                                                  MAIN 162
      FPCT= .FALSE.                                                     MAIN 163
      DO 303 I=1,15                                                     MAIN 164
      MIX(I) = 0.                                                       MAIN 165
  303 CONTINUE                                                          MAIN 166
      NT = 1                                                            MAIN 167
      EQL = .TRUE.                                                      MAIN 168
C     READ(5,INPT2)                                                     MAIN/169
      P(1)=PGPSIA                                                       MAINA169
      MIX(1)=OXFUL(KZONE)                                               MAINB169
C     WRITE(6,INPT2)                                                    MAIN/170
      IF(.NOT.DETN.AND..NOT.SHOCK)  GO TO 1303                          MAIN 171
      DO 1300  N=1,NREAC                                                MAIN 172
      IF(FAZ(N).NE.GAS)  GO TO 1301                                     MAIN 173
 1300 CONTINUE                                                          MAIN 174
      GO TO 1303                                                        MAIN 175
 1301 WRITE(6,1302)                                                     MAIN 176
 1302 FORMAT(60HOCONDENSED REACTANTS NOT PERMITTED IN DETN OR SHOCK PROBMAIN 177
     1LEMS)                                                             MAIN 178
      GO TO 1                                                           MAIN 179
 1303 IF(.NOT.TV.AND..NOT.UV.AND..NOT.SV)  GO TO 304                    MAIN 180
      VOL = .TRUE.                                                      MAIN 181
      DO 1304  I=1,26                                                   MAIN 182
      IF(RHO(I).NE.0.)  VL(I) = 1./RHO(I)                               MAIN 183
      IF(V(I).NE.0.)  VL(I)=V(I)                                        MAIN 184
      IF(VL(I).EQ.0.)  GO TO 1305                                       MAIN 185
      NP = I                                                            MAIN 186
 1304 CONTINUE                                                          MAIN 187
 1305 TP = TV                                                           MAIN 188
      HP = UV                                                           MAIN 189
```

```
      SP = SV                                                           MAIN 190
      GO TO 322                                                         MAIN 191
  304 DO 305  I=1,26                                                    MAIN 192
      IF(P(I ).EQ.0.) GO TO 322                                         MAIN 193
      NP = I                                                            MAIN 194
      IF (MMHG) P(NP) = P(NP)/760.                                      MAIN 195
      IF(PSIA         )  P(NP)=P(NP)/14.696006                          MAIN 196
      IF(NSQM) P(NP)=P(NP)/101325.                                      MAIN 197
  305 CONTINUE                                                          MAIN 198
  322 DO 307 IT = 1,26                                                  MAIN 199
      IF (T(IT).EQ.0.) GO TO 722                                        MAIN 200
      NT = IT                                                           MAIN 201
  307 CONTINUE                                                          MAIN 202
  722 DO 625 IST=1,15                                                   MAIN 203
      IF(WP(1).EQ.0.)  OXF(IST)=0.
      IF( MIX   (IST).NE.0.) GO TO 323                                  MAIN 204
      IF(IST.NE.1) GO TO 745                                            MAIN 205
      WRITE(6,724)                                                      MAIN 206
  724 FORMAT(48HONO INPT2 VALUE GIVEN FOR OF, EQRAT, FA, OR FPCT  )     MAIN 207
      IF (WP(2).NE.0.) OXFL = WP(1)/WP(2)                               MAIN 208
      GO TO 333                                                         MAIN 209
  323 OXFL  = MIX(IST)                                                  MAIN 210
      IF(FA) OXFL =1./   MIX(IST)                                       MAIN 211
      IF(FPCT) OXFL =(100.-   MIX(IST))/   MIX(IST)                     MAIN 212
      IF(.NOT.ERATIO)  GO TO 333                                        MAIN 213
      EQRAT = MIX(IST)                                                  MAIN 214
      IF(EQRAT.EQ.1.)  EQRAT = 1.000005                                 MAIN 215
      OXFL  = (-EQRAT*VMIN(2)-VPLS(2))/(VPLS(1)+EQRAT*VMIN(1))          MAIN 216
  333 OXF(IST) = OXFL                                                   MAIN 217
      NOF = IST                                                         MAIN 218
  625 CONTINUE                                                          MAIN 219
  745 IF (.NOT.IONS) GO TO 746                                          MAIN 220
      IF(LLMT(NLM).EQ.IE)  GO TO 748                                    MAIN 221
      NLM = NLM+1                                                       MAIN 222
C     IF(LLMT(NLM).EQ.IE)  GO TO 748                                    MAIN 223
      LLMT(NLM) = IE                                                    MAIN 225
      BOP(NLM,1) = 0.                                                   MAIN 226
      BOP(NLM,2) = 0.                                                   MAIN 227
      GO TO 747                                                         MAIN 228
  746 IF(LLMT(NLM).EQ.IE)  NLM=NLM-1                                    MAIN 229
      NLM1 = NLM+1
      IF(LLMT(NLM1).NE.IE)  GO TO 748
      LLMT(NLM1) = 0
  747 NEWR = .TRUE.
      REWIND 4
  748 IF(NEWR)  CALL SEARCH                                             MAIN 234
      IF(NS.EQ.0)  GO TO 1                                              MAIN 235
C                                                                       MAIN 236
C    INITIAL ESTIMATES                                                  MAIN 237
C                                                                       MAIN 238
```

```
      SO = SO/R                                                  MAIN 239
      ENN = .1                                                   MAIN 240
      ENNL = -2.3025851                                          MAIN 241
      SUMN = ENN                                                 MAIN 242
      XI = NS - NC                                               MAIN 243
      XI = ENN/XI                                                MAIN 244
      XLN = ALOG(XI)                                             MAIN 245
      DO 432 J=1,NS                                              MAIN 246
      IF(IUSE(J).GT.0)  IUSE(J)=-IUSE(J)                         MAIN 247
      EN(J,1) = 0.                                               MAIN 249
      ENLN(J) = 0.                                               MAIN 250
      IF (IUSE(J).NE.0)  GO TO 432                               MAIN 251
      EN(J,1) = XI                                               MAIN 252
      ENLN(J) = XLN                                              MAIN 253
  432 CONTINUE                                                   MAIN 254
      IQ1 = NLM+1                                                MAIN 255
      IF (NC.EQ.0.OR.NSERT.EQ.0) GO TO 790                       MAIN 256
      DO 302 I=1,NSERT                                           MAIN 257
      INC = 0                                                    MAIN 258
      DO 301 J=1,NS                                              MAIN 259
      IF(IUSE(J).EQ.0) GO TO 301                                 MAIN 260
      INC = INC+1                                                MAIN 261
      IF(SUB(J,1).NE.ENSERT(1,I)) GO TO 301                      MAIN 262
      IF(SUB(J,2).NE.ENSERT(2,I)) GO TO 301                      MAIN 263
      IF(SUB(J,3).NE.ENSERT(3,I)) GO TO 301                      MAIN 264
      IF(T(1).EQ.0.)  GO TO 295                                  MAIN 265
      IF(T(1).LT.TEMP(INC,1).OR.T(1).GT.TEMP(INC,2))  GO TO 301  MAIN 266
  295 IQ1 = IQ1+1                                                MAIN 267
      IUSE(J)= -IUSE(J)                                          MAIN 268
      GO TO 302                                                  MAIN 269
  301 CONTINUE                                                   MAIN 270
  302 CONTINUE                                                   MAIN 271
  790 CONTINUE                                                   MAIN 272
      IF(.NOT.TP.AND..NOT.HP.AND..NOT.SP)  GO TO 791             MAIN 273
      CALL THERMP                                                MAIN 274
      GO TO 800                                                  MAIN 275
  791 CONTINUE                                                   MAIN 276
C     IF(DETN) CALL DETON                                        MAIN/277
      IF(RKT) CALL ROCKET                                        MAIN 278
C     IF(SHOCK)  CALL SHCK                                       MAIN/279
  800 NSERT = 0                                                  MAIN 280
      GO TO 1                                                    MAIN 281
      END                                                        MAIN 282
```

```
      SUBROUTINE REACT                                                  REAC
C                                                                       REAC
      LOGICAL HP,SP,TP,        CONVG,NEWR,IONS,MOLES,EQL,FROZ,VOL       REAC
C                                                                       REAC
      DIMENSION ANAME(15,5),V(15),LLMTS(15),SBOP(15,2)                  REAC
C                                                                       REAC
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  REAC
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)REAC
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         REAC
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)REAC
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE           REAC
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  REAC
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ REAC
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHREAC
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    REAC
C                                                                       REAC
      EQUIVALENCE (NAME,ANAME),(NLM,L),(BLANK,LANK)                     REAC
C                                                                       REAC
      DATA MOL/1HM/,OX/1HO/,LANK/1H /,IZERO/2H00/,NLS/0/,ZERO/1H0/      REAC
C                                                                       REAC
      DO 10 K=1,2                                                       REAC
      WP(K)=0.                                                          REAC
      HPP(K)=0.                                                         REAC
      RH(K)=0.                                                          REAC
      VPLS(K)=0.                                                        REAC
      VMIN(K)=0.                                                        REAC
      AM(K)=0.                                                          REAC
      DO 8 J=1,15                                                       REAC
      LLMT(J)=0                                                         REAC
      BOP(J,K)=0.                                                       REAC
    8 CONTINUE                                                          REAC
   10 CONTINUE                                                          REAC
      NFUEL = 0                                                         REAC
      N=1                                                               REAC
      L=1                                                               REAC
C                                                                       REAC
C     READ AND WRITE REACTANT CARDS                                     REAC
C                                                                       REAC
   20 READ(5,21)(NAME(N,I),ANUM(N,I),I=1,5),PECWT(N),MOLE,ENTH(N),FAZ(N)REAC
     1 ,RTEMP(N),FOX(N),DENS(N)                                         REAC
   21 FORMAT(5(A2,F7.5),F7.5,A1,F9.5,A1,F8.5,A1,F8.5)                   REAC
      IF(NAME(N,1).EQ.LANK) GO TO 200                                   REAC
      IF(L.EQ.0)GO TO 20                                                REAC
C     WRITE (6,31)(NAME(N,I),ANUM(N,I),I=1,5),PECWT(N),MOLE,ENTH(N),FAZ REAC/
C    1   (N),RTEMP(N),FOX(N),DENS(N)                                    REAC/
   31 FORMAT(1X,5(A2,1X,F7.4,2X),F10.6,2X,A1,F11.2,2X,A1,2X,F8.3,2X,    REAC
     1A1,3X,F8.5)                                                       REAC
   35 IF(MOLE.EQ.MOL) MOLES=.TRUE.                                      REAC
C                                                                       REAC
C     IF OXIDANT, K=1                                                   REAC
```

```
C     IF FUEL, K=2
C
      IF(FOX(N).EQ.ZERO)  FOX(N)=OX
      K = 1
      IF(FOX(N).EQ.OX)  GO TO 37
      K = 2
      NFUEL = NFUEL+1
   37 DO 38  J=1,15
      DATA(J) = 0.
   38 CONTINUE
      RM=0.
C
C     STORE ATOMIC SYMBOLS IN LLMT ARRAY.
C     CALCULATE MOLECULAR WEIGHT.
C     TEMPORARILY STORE ATOMIC VALENCE IN V.
C
      DO 100 JJ=1,5
      IF(ANUM(N,JJ).EQ.0.)GO TO 101
      IF(ANAME(N,JJ).EQ.ZERO)  ANAME(N,JJ)=OX
      DO 41 J=1,15
      NJ = J
      IF(LLMT(J).EQ.0) GO TO 45
      IF(NAME(N,JJ).EQ.LLMT(J))GO TO 46
   41 CONTINUE
   45 L = NJ
      LLMT(J)=NAME(N,JJ)
   46 DO 48 KK=1,101
      IF(ATOM(1,KK).EQ.ANAME(N,JJ))GO TO 50
   48 CONTINUE
      L=0
      GO TO 20
   50 RM=RM+ANUM(N,JJ)*ATOM(2,KK)
      V(J)=ATOM(3,KK)
      DATA(J)=ANUM(N,JJ)
  100 CONTINUE
C
C     ADD CONTRIBUTIONS TO WP(K), HPP(K), AM(K), BOP(I,K) AND RH(K)
C
  101 PCWT=PECWT(N)
      IF(MOLES) PCWT=PCWT*RM
      WP(K)=WP(K) + PCWT
      EM = ENTH(N)
      IF(NAME(N,5).NE.IZERO)HPP(K)=HPP(K)+EM      *PCWT/(RM*R)
      AM(K)=AM(K)+PCWT/RM
      DO 110  J=1,L
      BOP(J,K)=DATA(J)*PCWT/RM +BOP(J,K)
  110 CONTINUE
      IF(DENS(N).NE.0.)GO TO 115
      GO TO 117
  115 RH(K)=RH(K)+PCWT/DENS(N)
```

```
  117 RMW(N) = RM
      N = N+1
      IF(N.NE.16) GO TO 20
  200 NREAC =N-1
      IF(NFUEL.GT.0)  GO TO 210
C
C     100 PERCENT OXIDANT, CALL REACTANTS FUEL
C
      DO 205  N=1,NREAC
      FOX(N) = BLANK
  205 CONTINUE
      RH(2) = RH(1)
      RH(1) = 0.
      WP(2) = WP(1)
      WP(1) = 0.
      HPP(2) = HPP(1)
      AM(2) = AM(1)
      AM(1) = 0.
      DO 208  J=1,L
      BOP(J,2) = BOP(J,1)
  208 CONTINUE
  210 IF(L.EQ.0)  GO TO 1000
C
C     NORMALIZE HPP(K),AM(K),BOP(I,K), AND PECWT(N).
C     CALCULATE RH(K), V+(K), AND V-(K)
C
      DO 220 K=1,2
      IF(WP(K).EQ.0.)GO TO 220
      HPP(K)=HPP(K)/WP(K)
      AM(K) = WP(K)/AM(K)
      IF(RH(K).NE.0.)RH(K)=WP(K)/RH(K)
      DO 215 J=1,L
      BOP(J,K)=BOP(J,K)/WP(K)
      IF(V(J).LT.0.)VMIN(K)= VMIN(K)+BOP(J,K)*V(J)
      IF(V(J).GT.0.)VPLS(K)=VPLS(K)+BOP(J,K)*V(J)
  215 CONTINUE
      IF(MOLES) GO TO 220
      DO 218 N=1,NREAC
      IF(FOX(N).EQ.OX.AND.K.EQ.2)  GO TO 218
      IF(FOX(N).NE.OX.AND.K.EQ.1)  GO TO 218
      PECWT(N) = PECWT(N)/WP(K)
  218 CONTINUE
  220 CONTINUE
      NEWR=.TRUE.
C
C  ARE ELEMENTS SAME AS FOR  LAST SET OF REACTANTS, IF SO, NEWR=.FALSE.
C
      IF(NLM.NE.NLS)  GO TO 226
      IF(NOMIT.NE.0)  GO TO 226
      DO 224 I=1,NLS
```

```
      DO 222  J=1,NLM
      IF(LLMT(J).NE.LLMTS(I))  GO TO 222
      SBOP(I,1) = BOP(J,1)
      SBOP(I,2) = BOP(J,2)
      GO TO 224
  222 CONTINUE
      GO TO 226
  224 CONTINUE
      NEWR = .FALSE.
      DO 225  I=1,NLM
      LLMT(I) = LLMTS(I)
      BOP(I,1)  = SBOP(I,1)
      BOP(I,2)  = SBOP(I,2)
  225 CONTINUE
      GO TO 229
C
C
  226 NLS = NLM
      NCMIT = 0
      REWIND 4
      DO 228 I=1,NLM
      LLMTS(I) = LLMT(I)
  228 CONTINUE
  229 DO 230  N=1,NREAC
      IF (DENS(N).NE.0.) GO TO 230
      RH(2) = 0.
      RH (1) = 0.
      GO TO 1000
  230 CONTINUE
 1000 RETURN
      END
```

```
      SUBROUTINE SEARCH
C
C   SEARCH TAPE FOR THERMO DATA FOR SPECIES TO BE CONSIDERED
C
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR
C      IBM 360 MACHINES ONLY
C
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN
C
      INTEGER SUB,CMIT,END,TOOBIG
C
      LOGICAL NEWR,OTTO
C
      DIMENSION DATE(2,3),MT(4),B(4),OMIT(3,3),NAM(3),TOOBIG(3,50)
C
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCH
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM
C
      EQUIVALENCE (DATE,EN),(OMIT,ENLN),(ENDD,END),(TOOBIG,ENLN)
C
      DATA  GAS/1HG/,END/3HEND/
C
      I28 = 0
      NC= 0
      IX= 0
C
C   CHECK DIMENSION FOR NUMBER OF SPECIES, CLEAR A(I,J)
C
      SUB(1,1) = END
      DO 3  I=1,1000
      IF(A(1,I).EQ.ENDD)   GO TO 4
      DO 3  J=1,NLM
      A(J,I) = 0.
    3 CONTINUE
    4 MAXNS = I-1
C
C   READ TEMPERATURE RANGES FOR COEFFICIENTS OF GASEOUS SPECIES.
C
      READ(4,5)  TLOW,TMID,THIGH
    5 FORMAT (3F10.3)
      NS = 1
```

```
C
C  BEGIN LOOP FOR READING SPECIES DATA FROM TAPE.
C
   7 READ     (4,10)(NAM(I),I=1,3),    DATE(1,NS),CATE(2,NS),(MT(J),B(J),
    1   J=1,4),PHAZ,T1,T2
  10 FORMAT(3A4,6X,2A3,4(A2,F3.0),A1,2F10.3)
     IF(NAM(1).EQ.END)   GO TO 171
     READ       (4,20)  ((COEF(I,J,NS),J=1,7),I=1,2)
  20 FORMAT (5E15.8)
     IF(NOMIT.EQ.0) GO TO 810
     DO 805 I=1,NCMIT
     DO 804 J=1,3
     IF(OMIT(J,I).NE.NAM(J)) GO TO 805
 804 CONTINUE
     GO TO 7
 805 CONTINUE
 810 DO 820 K=1,4
     IF(B(K).EQ.0.) GO TO 825
     DO 168 I=1,NLM
     IF(LLMT(I).EQ.MT(K)) GO TO 820
 168 CONTINUE
     IF(NS.GT.MAXNS)   GO TO 7
     DO 819 J=1,NLM
 819 A(J,NS) = 0.
     GO TO 7
 820 IF(NS.LE.MAXNS)   A(I,NS) = B(K)
 825 IF(NS.LE.MAXNS)   GO TO 828
     I2B = I2B+1
     DO 826  I=1,3
 826 TOOBIG(I,I2B) = NAM(I)
     GO TO 7
 828 DO 829  I=1,3
 829 SUB(NS,I) = NAM(I)
     IUSE(NS) = 0
     IF(PHAZ.EQ.GAS) GO TO 170
C
C  CONDENSED SPECIES
C
     NC= NC+1
     TEMP(NC,1)= T1
     TEMP(NC,2)= T2
     IX= IX+1
     IF(NS.EQ.1.OR.IUSE(NS-1).EQ.0)   GO TO 145
     DO 830 I=1,NLM
     IF(A(I,NS).NE.A(I,NS-1)) GO TO 145
 830 CONTINUE
     IX= IX-1
 145 IUSE(NS)= -IX
 170 NS= NS+1
     GO TO 7
```

```
C                                                                       SRCH 101
C   END CARD HAS BEEN READ.                                             SRCH 102
C                                                                       SRCH 103
  171 NS= NS-1                                                          SRCH 104
      NEWR= .FALSE.                                                     SRCH 105
C     WRITE(6,172)                                                      SRCH/106
  172 FORMAT(42HOSPECIES BEING CONSIDERED IN THIS SYSTEM    )           SRCH 107
      DO 174 I=1,NS,5                                                   SRCH 108
      I5= I+4                                                           SRCH 109
      IF(NS.LT.I5) I5=NS                                                SRCH 110
C 174 WRITE (6,176)(DATE(1,J),DATE(2,J),SUB(J,1),SUB(J,2),SUB(J,3),J=I, SRCH/111
C    1  I5)                                                             SRCH/112
  174 CONTINUE                                                          SRCHA112
  176 FORMAT(5(5X,2A3,2X,3A4))                                          SRCH 113
      IF(I2B.GT.0)  GO TO 870                                           SRCH 114
      RETURN                                                            SRCH 115
  870 WRITE(6,871)  I2B                                                 SRCH 116
  871 FORMAT(35HOINSUFFICIENT STORAGE FOR FOLLOWING,I3,8H SPECIES)      SRCH 117
      WRITE(6,880)(TOOBIG(1,J),TOOBIG(2,J),TOOBIG(3,J),J=1,I2B)         SRCH 118
  880 FORMAT(8(3X,3A4))                                                 SRCH 119
      NS = 0                                                            SRCH 120
      RETURN                                                            SRCH 121
      END                                                               SRCH 122
```

```
      SUBROUTINE HCALC
C
C     CALCULATE PROPERTIES FOR TOTAL REACTANT USING THERMO DATA FOR
C     ONE OR MORE REACTANTS.
C
C
C  THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR
C     IBM 360 MACHINES ONLY
C
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN
C
      LOGICAL MOLES,VOL,SHOCK,CALCH
C  CALCULATE ENTHALPY FOR PROPELLANT USING COEFFICIENTS
      DIMENSION NUM(15,5)
C
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)
     2 ,VLM(13),TOTN(13)
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCH
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM
C
      EQUIVALENCE (ANUM,NUM),(L,NLM),(J,JS1)
      EQUIVALENCE (AM1,DATA(20)),(CPR1,DATA(21))
      DATA AG/1HG/,IZERO/2H00/,OX/1HO/,BLK/1H /
C
      TSAVE = TT
C
C     CALCUALTE MOLECULAR WEIGHT OF TOTAL REACTANT, AM1.
C
      IF (AM(1).NE.0.0 .AND. AM(2).NE.0.0) GO TO4
      AM1= AM(2)
      IF (AM(2).EQ.0.0) AM1= AM(1)
      GO TO 9
    4 AM1=(OF+1.)*AM(1)*AM(2)/(AM(1)+OF*AM(2))
    9 TM = 0.
      IF(PP.GT.0.) TM = ALOG(PP*AM1)
      SSUM(NPT) = 0.
      HPP(1) = 0.
      HPP(2) = 0.
      HSUBO = 0.
```

```
      CPR1 = 0.                                                         HCAL  51
      ANN = (1.+OF)                                                     HCAL  52
C                                                                       HCAL  53
C     LCOOP ON REACTANTS.                                               HCAL  54
C     IF OXIDANT, K=1                                                   HCAL  55
C     IF FUEL, K=2                                                      HCAL  56
C                                                                       HCAL  57
      DO 900 N=1,NREAC                                                  HCAL  58
      K=2                                                               HCAL  59
      IF(FOX(N).EQ.OX)K=1                                               HCAL  60
      IF(NAME(N,5).NE.IZERO)  GO TO 90                                  HCAL  61
      IF(.NOT.CALCH.AND.TT.NE.0.)  GO TO 15                             HCAL  62
      TT = RTEMP(N)                                                     HCAL  63
C                                                                       HCAL  64
C   IS TT IN RANGE                                                      HCAL  65
C                                                                       HCAL  66
   15 IF(SHOCK)  GO TO 16                                               HCAL  67
      IF(TT.LT.(TLOW/1.2).OR.TT.GT.(THIGH*1.2))  GO TO 75               HCAL  68
   16 J = NUM(N,5)                                                      HCAL  69
      IF (J.NE.0)  GO TO 90                                             HCAL  70
      DO 10 J=1,L                                                       HCAL  71
      DATA(J)=0.                                                        HCAL  72
   10 CONTINUE                                                          HCAL  73
C                                                                       HCAL  74
C     TEMPORARILY STORE STOICHIOMETRIC COEFFICIENTS IN DATA ARRAY.      HCAL  75
C                                                                       HCAL  76
      DO 40 I=1,4                                                       HCAL  77
      IF(ANUM(N,I).EQ.0.)GO TO 50                                       HCAL  78
      DO 20 J=1,L                                                       HCAL  79
      IF(LLMT(J).EQ.NAME(N,I)) GO TO 30                                 HCAL  80
   20 CONTINUE                                                          HCAL  81
   30 DATA(J)=ANUM(N,I)                                                 HCAL  82
   40 CONTINUE                                                          HCAL  83
   50 IS=0                                                              HCAL  84
C                                                                       HCAL  85
C     SEARCH FOR REACTANT IN THERMO SPECIES.  STORE INDEX IN NUM(N,5).  HCAL  86
C                                                                       HCAL  87
      DO 70 J=1,NS                                                      HCAL  88
      IF(IUSE(J).EQ.0)GO TO 55                                          HCAL  89
      IS = IS+1                                                         HCAL  90
      IF(FAZ(N).EQ.AG)GO TO 70                                          HCAL  91
      IF(TT.GT.TEMP(IS,2).AND.TEMP(IS,2).NE.THIGH)  GO TO 70            HCAL  92
      IF(TT.LT.TEMP(IS,1).AND.TEMP(IS,1).NE.TLOW)  GO TO 70             HCAL  93
      GO TO 56                                                          HCAL  94
   55 IF(FAZ(N).NE.AG.AND.FAZ(N).NE.BLK)  GO TJ 70                      HCAL  95
   56 DO 60 I=1,L                                                       HCAL  96
      IF(A(I,J).NE.DATA(I))  GO TO 70                                   HCAL  97
   60 CONTINUE                                                          HCAL  98
      NUM(N,5) = J                                                      HCAL  99
      GC TO 90                                                          HCAL 100
```

```
   70 CONTINUE                                                                  HCAL 101
      GO TO 80                                                                  HCAL 102
C                                                                               HCAL 103
C     CALCULATE EN FOR REACTANT AND CALL CPHS TO CALCULATE PROPERTIES.          HCAL 104
C                                                                               HCAL 105
   90 IF (MOLES)        ENJ = PECWT(N)/WP(K)                                    HCAL 106
      IF (.NOT.MOLES)         ENJ = PECWT(N)/RMW(N)                             HCAL 107
      ENJ = ENJ/ANN                                                             HCAL 108
      IF(K.EQ.1)  ENJ = ENJ*OF                                                  HCAL 109
      IF(NAME(N,5).NE.IZERO)GO TO 500                                           HCAL 110
      NSS = NS                                                                  HCAL 111
      NS = J                                                                    HCAL 112
      TLN = ALOG(TT)                                                            HCAL 113
      IF(.NOT.CALCH)  EN(J,NPT) = ENJ                                           HCAL 114
      CALL CPHS                                                                 HCAL 115
      NS = NSS                                                                  HCAL 116
      IF (HO(J).GT.-.01 .AND. HO(J).LT..01) HO(J) = 0.                          HCAL 117
      RTEMP(N) = TT                                                             HCAL 118
      IF(VOL) HO(J)=HO(J)-1.                                                    HCAL 119
      ENTH(N) = HO(J)*R*TT                                                      HCAL 120
C                                                                               HCAL 121
C     ADD CONTRIBUTION TO CP, H, AND S OF TOTAL REACTANT.                       HCAL 122
C                                                                               HCAL 123
      CPR1 = CPR1 + CPSUM                                                       HCAL 124
      SSUM(NPT) = SSUM(NPT) + ENJ * (S(J)-ALOG(ENJ)-TM)                         HCAL 125
  500 ER = ENTH(N)*ENJ/R                                                        HCAL 126
      HSUBO = HSUBO+ER                                                          HCAL 127
      HPP(K) = HPP(K)+ER                                                        HCAL 128
  900 CONTINUE                                                                  HCAL 129
      IF(TSAVE.NE.0.)  TT=TSAVE                                                 HCAL 130
      GO TO 1000                                                                HCAL 131
   75 WRITE(6,76)                                                               HCAL 132
   76 FORMAT(50HOREACTANT TEMPERATURE OUT OF RANGE OF THERMO DATA )             HCAL 133
      TT = 0.                                                                   HCAL 134
      GO TO 1000                                                                HCAL 135
   80 WRITE(6,85) N                                                             HCAL 136
   85 FORMAT(9HOREACTANT,I2,22H IS NOT IN THERMO DATA )                         HCAL 137
      TT = 0.                                                                   HCAL 138
 1000 RETURN                                                                    HCAL 139
      END                                                                       HCAL 140
```

```
      SUBROUTINE SAVE                                                   SAVE   1
C                                                                       SAVE   2
C   SAVES OR USES COMPOSITIONS FROM PREVIOUS POINT AS INITIAL ESTIMATES SAVE   3
C                                                                       SAVE   4
C                                                                       SAVE   5
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR     SAVE   6
C      IBM 360 MACHINES ONLY                                            SAVE   7
C                                                                       SAVE   8
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN                           SAVE   9
C                                                                       SAVE  10
      LOGICAL VOL,CALCH,IONS,SHOCK                                      SAVE  11
C                                                                       SAVE  12
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)   SAVE  13
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)    SAVE  14
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  SAVE  15
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)SAVE  16
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         SAVE  17
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)SAVE  18
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE,SIZE      SAVE  19
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  SAVE  20
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ SAVE  21
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHSAVE  22
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    SAVE  23
C                                                                       SAVE  24
      EQUIVALENCE (AC(1),JSOLS),(AC(2),JLIQS)                           SAVE  25
C                                                                       SAVE  26
C                                                                       SAVE  27
      IF(ISV)100,10,200                                                 SAVE  29
C                                                                       SAVE  30
C   NEXT POINT FIRST T IN SCHEDULE, USE PREVIOUS COMPOSITIONS FOR THIS T SAVE  31
C                                                                       SAVE  32
   10 IQ1 = IQSAVE                                                      SAVE  33
      JSOL = JSOLS                                                      SAVE  34
      JLIQ = JLIQS                                                      SAVE  35
      ENN = ENSAVE                                                      SAVE  36
      ENNL = ENLSAV                                                     SAVE  37
      NLM = LSAVE                                                       SAVE  38
      DO 50 J = 1,NS                                                    SAVE  39
      IF (IUSE(J).EQ.0) GO TO 20                                        SAVE  51
      EN (J,NPT) = SLN(J)                                               SAVE  52
      IF(IUSE(J).GT.0) IUSE(J) = - IUSE(J)                              SAVE  53
      IF (EN(J,NPT).NE.0.)IUSE(J) = -IUSE(J)                            SAVE  54
      GO TO 50                                                          SAVE  55
   20 EN(J,NPT) = 0.                                                    SAVE  56
      ENLN(J) = SLN(J)                                                  SAVE  57
      IF(SLN(J).EQ.0.)  GO TO 50
      IF ((ENLN(J)-ENNL + 18.5).LE.0.) GO TO 50                         SAVE  58
      EN(J,NPT) = 2.718281828459**ENLN(J)                               SAVE  59
   50 CONTINUE                                                          SAVE  60
      GO TO 1000                                                        SAVE  62
```

```
C
C  FIRST T--SAVE COMPOSITIONS FOR FUTURE POINTS WITH THIS T
C
  100 ISV = -ISV
      JSOLS = JSOL
      JLIQS = JLIQ
      IQSAVE = IQ1
      ENSAVE = ENN
      ENLSAV = ENNL
      LSAVE = NLM
      DO 150 J = 1,NS
      SLN(J) = ENLN(J)
      IF(IUSE(J).NE.0)  SLN(J)=EN(J,ISV)
  150 CONTINUE
C
C  USE COMPOSITIONS FROM PREVIOUS POINT
C
  200 DO 300 J = 1,NS
      EN(J,NPT) = EN(J,ISV)
  300 CONTINUE
 1000 RETURN
C
C  CALCULATE NEW VALUES OF BO AND HSUBO FOR NEW OF RATIO
C
      ENTRY NEWOF
C
C
C
      WRITE(6,730) OF
  730 FORMAT(6HOOF = ,F10.6)
      EQRAT = 0.
      SUM = OF + 1.
      V2 =    (OF*VMIN(1)+VMIN(2))/SUM
      V1 =    (OF*VPLS(1)+VPLS(2))/SUM
      IF(V2.NE.0.)  EQRAT=ABS(V1/V2)
      IF  (RH(1) .NE. 0. .AND. RH(2) .NE. 0.) GO TO 744
      RHOP = RH(2)
      IF (RHOP .EQ. 0.) RHOP = RH(1)
      GO TO 745
  744 RHOP =  (OF+1.)*RH(1)*RH(2)/(RH(1)+ OF *RH(2))
  745 DO 747 I=1,NLM
      BO(I) = (OF   *BOP(I,1)+      BOP(I,2))/SUM
      IF(I.NE.1)   GO TO 746
      BIGB = BO(1)
      SMALB = BO(1)
      GO TO 747
  746 IF(BO(I).EQ.0.)  GO TO 747
      IF(BO(I).LT.SMALB)  SMALB=BO(I)
      IF(BO(I).GT.BIGB)  BIGB=BO(I)
  747 CONTINUE
      NPT = 1
```

```
      IF(.NOT.CALCH)  GO TO 750                                         SAVE0106
      CALL HCALC                                                        SAVE0107
      IF(TT.EQ.0.)  RETURN                                              SAVE0108
      CALCH = .FALSE.                                                   SAVE0109
      IF(OF.NE.0.)  HPP(1)=SUM*HPP(1)/OF                                SAVE0110
      HPP(2) = SUM*HPP(2)                                               SAVE0111
      GO TO 760                                                         SAVE0112
  750 HSUBO=  (OF*HPP(1) + HPP(2))/SUM                                  SAVE0113
  760 IC = 0                                                            SAVE0114
      TEM = SMALB/BIGB
      SIZE = 18.420681
      IF(TEM.LT.1.E-5)  SIZE=ALOG(1000./TEM)
      JSOL = 0                                                          SAVE0115
      JLIQ = 0                                                          SAVE0116
C     WRITE (6,770)                                                     SAVE/117
  770 FORMAT(1H ,25X,14HEFFECTIVE FUEL,10X,17HEFFECTIVE OXIDANT,12X,7HMISAVE0118
     1XTURE )                                                           SAVE0119
      IF(VOL)  WRITE(6,772)                                             SAVE0120
C     IF(.NOT.VOL)  WRITE(6,774)                                        SAVE/121
  772 FORMAT(16H INTERNAL ENERGY,14X,6HHPP(2),19X,6HHPP(1),19X,5HHSUBO )SAVE0122
  774 FORMAT(9H ENTHALPY,21X,6HHPP(2),19X,6HHPP(1),19X,5HHSUBO )        SAVE0123
C     WRITE(6,776)HPP(2),HPP(1),HSUBO                                   SAVE/124
  776 FORMAT(19H (KG-MOL)(DEG K)/KG,E21.8,2E25.8 )                      SAVE0125
C     WRITE(6,778)                                                      SAVE/126
  778 FORMAT(12HOKG-ATOMS/KG,17X,8HBOP(I,2),17X,8HBOP(I,1),18X,5HBO(I)) SAVE0127
  780 FORMAT(8X,A2,5X,3E25.8)                                           SAVE0128
C     WRITE(6,780)(LLMT(I),      BOP(I,2),BOP(I,1),BO(I),I=1,NLM)       SAVE/129
      RETURN                                                            SAVE0130
      END                                                               SAVE0131
```

```
      SUBROUTINE EQLBRM                                                 EQLM   1
C   ROUTINE TO CALCULATE EQUILIBRIUM COMPOSITION AND PROPERTIES         EQLM   2
C                                                                       EQLM   3
      DOUBLE PRECISION X,G,SUM,SUM2,E,BOD                               EQLM   4
C                                                                       EQLM   5
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR     EQLM   6
C     IBM 360 MACHINES ONLY                                             EQLM   7
C                                                                       EQLM   8
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS                 EQLM   9
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN                           EQLM  10
      DOUBLE PRECISION ENL,PROW,DLNT,AA                                 EQLM  11
C                                                                       EQLM  12
      LOGICAL HP,SP,TP,CONVG,IONS,SINGC,LOGV,ISING,I35,VOL,SHOCK,RITE   EQLM  13
C                                                                       EQLM  14
C                                                                       EQLM  15
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)       EQLM  16
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)  EQLM  17
     2 ,VLM(13),TOTN(13)                                                EQLM  18
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)   EQLM  19
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)    EQLM  20
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  EQLM  21
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)EQLM  22
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         EQLM  23
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)EQLM  24
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE,SIZE      EQLM  25
      COMMON /DOUBLE/ G(20,21), X(20)                                   EQLM  26
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  EQLM  27
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ EQLM  28
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHEQLM  29
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    EQLM  30
C                                                                       EQLM  31
      EQUIVALENCE (NLM,L),(LOGV,CPCVEQ)                                 EQLM  32
C                                                                       EQLM  33
      DATA IE/1HE/,SMALNO/1.E-6/,SMNOL/-13.815511/,ITN/35/              EQLM  34
C                                                                       EQLM  35
      SINGC = .FALSE.                                                   EQLM  36
      PIE = 0.
      I35 = .FALSE.
      E = 2.718281828459
      ENL = ENNL                                                        EQLM  37
      RITE = .FALSE.                                                    EQLM  38
      IF(IDEBUG.GT.0.AND.NPT.GE.IDEBUG)  RITE=.TRUE.                    EQLM  39
      ISING = .FALSE.                                                   EQLM  41
      LOGV = .FALSE.                                                    EQLM  42
      IF(.NOT.VOL) GO TO 6                                              EQLM  43
      RV = RR/101.325                                                   EQLM  44
      PP = RV*ENN*TT/VLM(NPT)                                           EQLM  45
    6 TLN = ALOG(TT)                                                    EQLM  46
      CONVG = .FALSE.                                                   EQLM  47
      ITNUMB = ITN                                                      EQLM  48
```

```
      JS1 = 1                                                           EQLM  49
      CALL CPHS                                                         EQLM  50
      TM = ALOG(PP/ENN)                                                 EQLM  51
C                                                                       EQLM  52
      IF (.NOT.ICNS.OR.IE.EQ.LLMT(L)) GO TO 33                          EQLM  53
      L = L+1                                                           EQLM  54
      IQ1 = IQ1+1                                                       EQLM  55
      DO 499 J = 1,NS                                                   EQLM  56
      IF (A(L,J) .EQ.0.) GO TO 499                                      EQLM  57
      EN(J,NPT) = 1.E-8                                                 EQLM  58
      ENLN(J) = -SIZE                                                   EQLM  59
  499 CONTINUE                                                          EQLM  61
C  33 IF(NPT.EQ.1.AND..NOT.SHOCK)         WRITE(6,244)(LLMT(I),I=1,L)   EQLM/ 62
   33 CONTINUE                                                          EQLMA 62
  244 FORMAT (4HOPT ,14(5X,A4))                                         EQLM  63
C                                                                       EQLM  64
C     BEGIN ITERATION                                                   EQLM  65
C                                                                       EQLM  66
   43 IF (.NOT.CONVG)  GO TO 62                                         EQLM  67
      SUMN = ENN                                                        EQLM  68
      IF(JSOL.EQ.0)  GO TO 62                                           EQLM  69
      ENSOL = EN(JSOL,NPT)                                              EQLM  70
      EN(JSOL,NPT) = EN(JSOL,NPT)+EN(JLIQ,NPT)                          EQLM  71
      IUSE(JLIQ) = -IUSE(JLIQ)                                          EQLM  72
      IQ1 = IQ1-1                                                       EQLM  73
      DLVTP(NPT) = 0.                                                   EQLM  74
      CPR(NPT) = 0.                                                     EQLM  75
      GAMMAS(NPT) = 0.                                                  EQLM  76
      LOGV = .TRUE.                                                     EQLM  77
   62 CALL MATRIX                                                       EQLM  78
      NUMB = ITN-ITNUMB+1                                               EQLM  79
      IQ2 = IQ1 + 1                                                     EQLM  80
      IF(CONVG)  IMAT=IMAT-1                                            EQLM  81
      IF(.NOT.RITE)  GO TO 72                                           EQLM  82
      IF(.NOT.CONVG)  GO TO 88                                          EQLM  83
      IF(.NOT.LOGV)  WRITE(6,81)                                        EQLM  84
   81 FORMAT(15HOT DERIV MATRIX)                                        EQLM  85
      IF(LOGV)  WRITE(6,82)                                             EQLM  86
   82 FORMAT(15HOP DERIV MATRIX)                                        EQLM  87
      GO TO 89                                                          EQLM  88
   88 WRITE(6,772) NUMB                                                 EQLM  89
  772 FORMAT (11HOITERATION  ,I3,6X,7HMATRIX //)                        EQLM  90
   89 DO 911 I=1,IMAT                                                   EQLM  91
  911 WRITE (6,73) (G(I,K),K=1,KMAT)                                    EQLM  92
   72 ITST = IMAT                                                       EQLM  93
      CALL GAUSS                                                        EQLM  94
      IF(ITST.NE.IMAT)  GO TO 774                                       EQLM  95
      IF(.NOT.RITE)  GO TO 773                                          EQLM  96
      WRITE (6,373)(LLMT(I),I=1,L)                                      EQLM  97
  373 FORMAT (7HOPI    ,9(A4,10X))                                      EQLM  98
```

```
      WRITE (6,73)(X(I),I=1,IMAT)                                      EQLM  99
   73 FORMAT (9E14.6)                                                  EQLM 100
  773 IF(.NOT.CONVG)  GO TO 85                                         EQLM 101
      IF(.NOT.LOGV)  GO TO 174                                         EQLM 102
      GO TO 171                                                        EQLM 103
C                                                                      EQLM 104
C     TEMPERATURE DERIVATIVES--CONVG=T, LOGV=F                         EQLM 105
C                                                                      EQLM 106
  174 DLVTP(NPT) = 1.-X(IQ1)                                           EQLM 107
  175 CPR(NPT) = G(IQ2,IQ2)                                            EQLM 108
      DO 176 J=1,IQ1                                                   EQLM 109
      CPR(NPT) = CPR(NPT)-G(IQ2,J)*X(J)                                EQLM 110
  176 CONTINUE                                                         EQLM 111
C                                                                      EQLM 112
C     PRESSURE DERIVATIVE--CONVG=T, LOGV=T                             EQLM 113
C                                                                      EQLM 114
      LOGV = .TRUE.                                                    EQLM 115
      GO TO 62                                                         EQLM 116
C                                                                      EQLM 117
C  SINGULAR MATRIX                                                     EQLM 118
C                                                                      EQLM 119
  774 IF(.NOT.CONVG)  GO TO 775                                        EQLM 120
      WRITE(6,172)                                                     EQLM 121
  172 FORMAT(28H0DERIVATIVE MATRIX SINGULAR )                          EQLM 122
      GO TO 1171                                                       EQLM 123
  775 IF(.NOT.HP.OR.NPT.NE.1.OR.NC.EQ.0.OR.TT.GT.100.)  GO TO 871      EQLM 124
      WRITE(6,874)                                                     EQLM 125
  874 FORMAT(96H0LOW TEMPERATURE IMPLIES CONDENSED SPECIES SHOULD HAVE EQLM 126
     1BEEN INCLUDED ON AN INSERT CARD, RESTART )                       EQLM 127
      GO TO 873                                                        EQLM 128
  871 WRITE(6,74)                                                      EQLM 129
   74 FORMAT(16H0SINGULAR MATRIX)                                      EQLM 130
      IF(SINGC)  GO TO 873                                             EQLM 131
      DO 970 JJ = 1, NS                                                EQLM 132
      IF(IUSE(JJ).NE.0)  GO TO 970                                     EQLM 133
      IF(EN(JJ,NPT).NE.0.)  GO TO 970                                  EQLM 134
      EN(JJ,NPT) = SMALNO                                              EQLM 135
      ENLN(JJ) = SMNOL                                                 EQLM 136
  970 CONTINUE                                                         EQLM 137
      IF(ISING)  GO TO 870                                             EQLM 138
      ISING = .TRUE.                                                   EQLM 139
      WRITE (6,776)                                                    EQLM 140
  776 FORMAT (8H0RESTART)                                              EQLM 141
      GO TO 62                                                         EQLM 142
C                                                                      EQLM 143
C     TEST FOR SINGULARITY TO CONDENSED SPECIES.                       EQLM 144
C                                                                      EQLM 145
  870 NCOND = IQ1-NLM-2                                                EQLM 146
      IF(TP.OR.VOL)  NCOND=NCOND+1
      IF(NCOND.LT.2.OR.SIZEG.EQ.0.)  GO TO 873                         EQLM 148
```

```
      DO 872  J=1,NS                                                    EQLM 149
      IF(IUSE(J).LE.0)  GO TO 872                                       EQLM 150
      IF(J.EQ.JDELG)  GO TO 872                                         EQLM 151
      DO 671  I=1,NLM                                                   EQLM 152
      IF(A(I,J).EQ.A(I,JDELG))  GO TO 671                               EQLM 153
      IF(A(I,J).EQ.0..OR.A(I,JDELG).EQ.0.)  GO TO 872                   EQLM 154
  671 CONTINUE                                                          EQLM 155
      SINGC = .TRUE.                                                    EQLM 156
      IQ1 = IQ1-1                                                       EQLM 157
      EN(J,NPT) = 0.                                                    EQLM 158
      IUSE(J) = -IUSE(J)                                                EQLM 159
  872 CONTINUE                                                          EQLM 160
      IF(SINGC)  GO TO 40                                               EQLM 161
      GO TO 873                                                         EQLM 162
C                                                                       EQLM 163
C     OBTAIN CORRECTIONS TO THE ESTIMATES                               EQLM 164
C                                                                       EQLM 165
   85 ITNUMB= ITNUMB-1                                                  EQLM 166
      KK = L + 1                                                        EQLM 167
      IF(VOL)  X(IQ2)=X(IQ1)                                            EQLM 168
      IF(TP)  X(IQ2)=0.                                                 EQLM 169
      DLNT= X(IQ2)                                                      EQLM 170
      SUM = X(IQ1)                                                      EQLM 171
      IF(.NOT.VOL) GO TO 97                                             EQLM 172
      X(IQ1) = 0.                                                       EQLM 173
      SUM =    -DLNT                                                    EQLM 174
   97 DO 101 J=1,NS                                                     EQLM 175
      IF (IUSE(J)) 101,98,100                                           EQLM 176
   98 DELN(J) = HO(J)*DLNT-HO(J)+S(J)-ENLN(J)-TM+SUM                    EQLM 177
      DO 99 K=1,L                                                       EQLM 178
      DELN(J)= DELN(J)+A(K,J)*X(K)                                      EQLM 179
   99 CONTINUE                                                          EQLM 180
      IF(PIE.NE.0.)  DELN(J)=DELN(J)+A(L+1,J)*PIE
      GO TO 101                                                         EQLM 181
  100 DELN(J) = X(KK)                                                   EQLM 182
      KK = KK + 1                                                       EQLM 183
  101 CONTINUE                                                          EQLM 184
C                                                                       EQLM 185
C     CALCULATE CONTROL FACTOR,AMBDA                                    EQLM 186
C                                                                       EQLM 187
      AMBDA= 1.                                                         EQLM 188
      AMBDA1= 1.                                                        EQLM 189
      SUM = X(IQ1)                                                      EQLM 190
      IF(SUM.LT.0.)  SUM=-SUM                                           EQLM 191
      IF(DLNT.GT.SUM)  SUM=DLNT                                         EQLM 192
      IF(-DLNT.GT.SUM)  SUM=-DLNT                                       EQLM 193
      DO 917 J=1,NS                                                     EQLM 194
      IF (IUSE(J).NE.0) GO TO 917                                       EQLM 195
      IF((EN(J,NPT).GT.0.).AND.DELN(J).GT.SUM) SUM = DELN(J)            EQLM 196
      IF((EN(J,NPT).NE.0.) .OR. DELN(J).LE.0.) GO TO 917                EQLM 197
```

```
      SUM1 = (-9.212-ENLN(J)+ ENL)/(DELN(J)-X(IQ1))                         EQLM 198
      IF(SUM1.LT.0.)  SUM1=-SUM1                                            EQLM 199
      IF (SUM1.LT.AMBDA1) AMBDA1 = SUM1                                     EQLM 200
  917 CONTINUE                                                              EQLM 201
      IF(SUM.GT.2.)AMBDA=2./SUM                                             EQLM 202
      IF (AMBDA1.LT.AMBDA) AMBDA = AMBDA1                                   EQLM 203
      IF(.NOT.RITE)  GO TO 111                                              EQLM 204
C                                                                           EQLM 205
C     INTERMEDIATE OUTPUT                                                   EQLM 206
C                                                                           EQLM 207
      WRITE(6,923) TT,ENN, ENL,PP,TM,AMBDA                                  EQLM 208
  923 FORMAT (3HOT=,E15.8,6H  ENN=,E15.8,7H  ENNL=E15.8,5H  PP=,E15.8,      EQLM 209
     1 9H  LN P/N=E15.8,8H  AMBDA=E15.8 )                                   EQLM 210
      IF(VOL)  WRITE(6,1924) VLM(NPT)                                       EQLM 211
 1924 FORMAT(8H VOLUME=,E15.8,4HCC/G)                                       EQLM 212
      WRITE (6,924)                                                         EQLM 213
  924 FORMAT(1H0,18X,2HNJ,12X,5HLN NJ,8X,9HDEL LN NJ,9X,6HHOJ/RT,9X,5HSOEQLM 214
     1J/R,10X,7H-GOJ/RT,8X,6H-GJ/RT  )                                      EQLM 215
      DO 926 J=1,NS                                                         EQLM 216
      GNEG1 = S(J)-HO(J)                                                    EQLM 217
      GNEG2 = GNEG1                                                         EQLM 218
      IF(IUSE(J).EQ.0)  GNEG2=GNEG2-ENLN(J)-TM                              EQLM 219
      WRITE (6,925) SUB(J,1),SUB(J,2),                                      EQLM 220
     1SUB(J,3),EN(J,NPT),ENLN(J),DELN(J),HO(J),S(J),GNEG1,GNEG2             EQLM 221
  925 FORMAT (1X,3A4,7E15.6)                                                EQLM 222
  926 CONTINUE                                                              EQLM 223
      WRITE (6,110)                                                         EQLM 224
  110 FORMAT(1H0)                                                           EQLM 225
C                                                                           EQLM 226
C     APPLY CORRECTIONS TO ESTIMATES                                        EQLM 227
C                                                                           EQLM 228
  111 SUM = 0.                                                              EQLM 229
      DO 113 J=1,NS                                                         EQLM 230
      IF (IUSE(J)) 113,112,114                                              EQLM 231
  112 ENLN(J)=ENLN(J)+AMBDA*DELN(J)                                         EQLM 232
      EN(J,NPT) = 0.                                                        EQLM 233
      IF((ENLN(J)- ENL+SIZE).LE.0.)  GO TO 113                              EQLM 234
      EN(J,NPT) = E**ENLN(J)                                                EQLM 235
      SUM = SUM+EN(J,NPT)                                                   EQLM 236
      GO TO 113                                                             EQLM 237
  114 EN(J,NPT) = EN(J,NPT) + AMBDA * DELN(J)                               EQLM 238
  113 CONTINUE                                                              EQLM 239
      SUMN = SUM                                                            EQLM 240
      IF (TP) GO TO 115                                                     EQLM 241
      TLN= TLN+AMBDA*DLNT                                                   EQLM 242
      TT     = EXP(TLN)                                                     EQLM 243
      JS1 = 1                                                               EQLM 244
      CALL CPHS                                                             EQLM 245
  115 IF(VOL) GO TO 2115                                                    EQLM 246
      ENL = ENL+AMBDA*X(IQ1)                                                EQLM 247
```

```
      ENN = E**ENL
      GO TO 1115
 2115 ENN = SUMN
      ENL = ALOG(ENN)
      PP =  RV*TT*ENN/VLM(NPT)
 1115 TM = ALOG(PP/ENN)
      IF (LLMT(L).NE.IE) GO TO 116
C
C     CHECK ON REMOVING IONS
C
      DO 1116 J = 1,NS
      IF (A(L,J).EQ.0.) GO TO 1116
      IF (EN(J,NPT).GT.0.) GO TO 116
 1116 CONTINUE
      PIE = X(L)
      L = L-1
      IQ1 = IQ1-1
      GO TO 43
C
C     TEST FOR CONVERGENCE
C
  116 IF (ITNUMB.EQ.0) GO TO 14
      IF (AMBDA.LT.1.) GO TO 43
      SUM = X(IQ1)
      IF (SUM.LT.0.) SUM = -SUM
      IF (SUM.GT.0.5E-5) GO TO 43
      DO 130 J=1,NS
      IF (IUSE(J).LT.0) GO TO 130
      AA= DELN(J)/SUMN
      IF(AA.LT.0.)  AA=-AA
      IF (IUSE(J).EQ.0) AA = AA*EN(J,NPT)
      IF(AA.GT.0.5E-5)  GO TO 43
  130 CONTINUE
      LE = L
      IF(TRACE.EQ.0.)  GO TO 275
      IF(ITN.GT.35)  GO TO 222
      ITN = ITN+15
      ITNUMB = ITNUMB+15
  222 DO 225  I=1,NLM
      IF(BO(I).EQ.0.)  GO TO 227
      SUM = 0.
      DO 223  J=1,NS
  223 SUM = SUM+EN(J,NPT)*A(I,J)
      BOD = DBLE(BO(I))
      IF(DABS(BOD  -SUM)/BOD  .GT..0001)  GO TO 43
  225 CONTINUE
  227 IF(.NOT.IONS)  GO TO 275
C
C     CHECK ON ELECTRON BALANCE
C
```

```
      IF(PIE.NE.0.)  LE=L+1
      IF(PIE.EQ.0.)  PIE=X(L)
      ITER = 1
  229 SUM2 = 0.
      SUM = 0.
      DO 230 J=1,NS
      IF(IUSE(J).LT.0.OR.A(LE,J).EQ.0.)  GO TO 230
      IF(ENLN(J).GT.-87.)  EN(J,NPT)=E**ENLN(J)
      AN = A(LE,J)*EN(J,NPT)
      IF(EN(J,NPT)/ENN.GT.1.E-7)  GO TO 275
      SUM = SUM+AN
      SUM2 = SUM2+AN*A(LE,J)
  230 CONTINUE
      IF(SUM2.EQ.0.)  GO TO 275
      DPIE = -SUM/SUM2
      DO 250  J=1,NS
      IF(A(LE,J).EQ.0.)  GO TO 250
      ENLN(J) = ENLN(J)+A(LE,J)*DPIE
  250 CONTINUE
      IF(ABS(DPIE).LE..0001)  GO TO 275
      PIE = PIE+DPIE
      ITER = ITER+1
      IF(ITER.LE.ITN)  GO TO 229
      WRITE(6,260)
  260 FORMAT(37HODID NOT CONVERGE ON ELECTRON BALANCE)
      GO TO 873
  275 CONTINUE
C                                                                       EQLM 283
C     CALCULATE ENTROPY, CHECK ON DELTA S FOR SP PROBLEMS               EQLM 284
C                                                                       EQLM 285
      TOTN(NPT) = 0.                                                    EQLM 286
      SSUM(NPT) = 0.                                                    EQLM 287
      DO 183 J=1,NS                                                     EQLM 288
      IF(IUSE(J).LT.0)  GO TO 183                                       EQLM 289
      TOTN(NPT) = TOTN(NPT) + EN(J,NPT)                                 EQLM 290
      SS = S(J)                                                         EQLM 291
      IF(IUSE(J).EQ.0)  SS=SS-ENLN(J)-TM                                EQLM 292
      SSUM(NPT) = SSUM(NPT)+SS*EN(J,NPT)                                EQLM 293
  183 CONTINUE                                                          EQLM 294
      IF(.NOT.SP)  GO TO 13                                             EQLM 294
      SS = SSUM(NPT) -SO                                                EQLM 296
      IF(SS.LT.(-0.00005).OR.SS.GT.0.00005)  GO TO 43                   EQLM 297
      IF(RITE)  WRITE(6,1183)  SS                                       EQLM 298
 1183 FORMAT(12HODELTA S/R =,E15.8)                                     EQLM 299
C                                                                       EQLM 300
   13 CONVG= .TRUE.                                                     EQLM 301
      GC TO 160                                                         EQLM 302
   14 WRITE(6,973) ITN,NPT                                              EQLM 303
  973 FORMAT(1HL,I2,69H ITERATIONS DID NOT SATISFY CONVERGENCE REQUIREMEEQLM 304
     1NTS FOR THE PCINT        I5)                                      EQLM 305
```

```
      IF(NC.EQ.0.OR.I35)  GO TO 873
      I35 = .TRUE.
      IF (.NOT.HP.OR.NPT.NE.1.OR.TT.GT.100.)  GO TO 261                 EQLM 306
      WRITE(6,874)                                                      EQLM 307
      GO TO 873                                                         EQLM 308
  261 NCOND = IQ1-NLM-2
      IF(TP.OR.VOL)  NCOND=NCOND+1
      IF(NCOND.NE.1.OR.ENN.GT.1.E-4)  GO TO 873
C  HIGH TEMPERATURE, INCLUDED CONDENSED CONDITION
      WRITE(6,265)
  265 FORMAT(31HOTRY REMOVING CONDENSED SPECIES)
      ENN = .1
      ENL  = -2.3025851
      SUMN = ENN
      XI = NS - NC
      XI = ENN/XI
      XLN = ALOG(XI)
      DO 432 J=1,NS
      IF(IUSE(J).GT.0)  IUSE(J)=-IUSE(J)
      EN(J,NPT) = 0.
      ENLN(J) = 0.
      IF (IUSE(J).NE.0)  GO TO 432
      EN(J,NPT) = XI
      ENLN(J) = XLN
  432 CONTINUE
      IQ1 = NLM+1
      GO TO 40
C                                                                       EQLM 309
C     CONVERGENCE TESTS ARE SATISFIED, TEST CONDENSED SPECIES.          EQLM 310
C                                                                       EQLM 311
  160 IF(NC.EQ.0) GO TO 143                                             EQLM 312
      DO 146  J=1,NS                                                    EQLM 313
      IF(EN(J,NPT).GE.0.)  GO TO 146                                    EQLM 314
      IF (J.NE.JSOL .AND. J .NE.JLIQ) GO TO 147                         EQLM 315
      JSOL = 0                                                          EQLM 316
      JLIQ = 0                                                          EQLM 317
  147 IQ1 = IQ1 - 1                                                     EQLM 318
      EN(J,NPT) = 0.                                                    EQLM 319
      GO TO 166                                                         EQLM 320
  146 CONTINUE                                                          EQLM 321
      SIZEG = 0.                                                        EQLM 322
      INC = 0                                                           EQLM 323
      DO 170 J = 1,NS                                                   EQLM 324
      IF(IUSE(J).EQ.0)  GO TO 170                                       EQLM 325
      INC = INC + 1                                                     EQLM 326
      IF(RITE)  WRITE(6,144)(SUB(J,I),I=1,3),TEMP(INC,1),TEMP(INC,2),IUSEQLM 327
     1E(J),EN(J,NPT)                                                    EQLM 328
  144 FORMAT (1HO,3A4,2F10.3,3X,5HIUSE=,I4,E15.7)                       EQLM 329
      IF(EN(J,NPT).GT.0.)  GO TO 169                                    EQLM 330
      KG = 1                                                            EQLM 331
```

```
      IF(IUSE(J).EQ.-IUSE(J+1)) GO TO 154
      IF(J.EQ.1.OR.IUSE(J).NE.-IUSE(J-1)) GO TO 153
      KG = -1
  154 JKG = J + KG
      TMELT = TEMP(INC,1)
      IMP = INC + KG
      IF(TMELT.EQ.TEMP(IMP,2  )) GO TO 158
      TMELT = TEMP(INC,2)
      IF (TMELT.EQ.TEMP(IMP,1)) GO TO 157
      WRITE (6,156)
  156 FORMAT (50H03 PHASES OF A CONDENSED SPECIES ARE OUT OF ORDER   )
      GO TO 873
C
C  JTH SPECIES A SOLID (EN=0), (J+KG)TH SPECIES A LIQUID (EN IS +)
C
  157 IF(TT.GT.TMELT) GO TO 169
      IF (TP.AND.TT.EQ.TMELT) GO TO 169
      IF (TP)  GO TO 1165
      IF (TT.LE.TMELT-150.) GO TO 1165
      JSOL = J
      JLIQ = JKG
      GO TO 159
C
C  JTH SPECIES A LIQUID(EN=0), (J+KG)TH SPECIES A SOLID (EN IS +)
C
  158 IF (TT.LT.TMELT) GO TO 169
      IF (TP.AND.TT.EQ.TMELT) GO TO 169
      IF (TP)  GO TO 1165
      IF (TT.GE.TMELT+150.) GO TO 1165
      JSOL = JKG
      JLIQ = J
  159 TLN = ALOG (TMELT)
      TT = TMELT
      EN(JKG,NPT) = .5 * EN(JKG,NPT)
      EN(J,NPT) = EN(JKG,NPT)
      GO TO 165
C
C  WRONG PHASE INCLUDED FOR T INTERVAL, SWITCH EN
C
 1165 EN(J,NPT) = EN (JKG, NPT)
      IUSE(J) = -IUSE(J)
      IUSE (JKG) = -IUSE(JKG)
      EN(JKG,NPT)= 0.
      GO TO 40
  153 IF (TT.LT.TEMP(INC,1) .AND.TEMP(INC,1).NE.TLOW) GO TO 169
      IF (TT.GT.TEMP(INC,2)) GO TO 169
C
C
      SUM = 0.
      DO 167 I = 1,L
```

```
      SUM = SUM + A(I,J)*X(I)                                           EQLM 382
  167 CONTINUE                                                          EQLM 383
      DELG = HO(J)-S(J)-SUM                                             EQLM 384
      IF(RITE)  WRITE(6,168)DELG,SIZEG                                  EQLM 385
  168 FORMAT (18H GO-SUM(AIJ*PII) =E15.7,10X,17HMAX NEG DELTA G =,E15.7)EQLM 386
      IF(DELG.GE.SIZEG .OR. DELG.GE.0.) GO TO 169                       EQLM 387
      SIZEG = DELG                                                      EQLM 388
      JDELG = J                                                         EQLM 389
  169 IF(INC.EC.NC) GO TO 1160                                          EQLM 390
  170 CONTINUE                                                          EQLM 391
 1160 IF (SIZEG.EQ.0.) GO TO 143                                        EQLM 392
      J = JDELG                                                         EQLM 393
  165 IQ1 = IQ1 + 1                                                     EQLM 394
  166 IUSE(J) = - IUSE(J)                                               EQLM 395
   40 CONVG = .FALSE.                                                   EQLM 396
      JS1 = 1                                                           EQLM 397
      CALL CPHS                                                         EQLM 398
  143 TN = NUMB                                                         EQLM 399
      IF(PIE.NE.0.)  X(LE) = PIE
C 770 IF(.NOT.SHOCK)        WRITE(6,771)NPT,(X(IL),IL=1,LE),TN          EQLM/400
  770 CONTINUE                                                          EQLMA400
  771 FORMAT (I3,14F9.3)                                                EQLM 401
      JS1 = 1                                                           EQLM 402
      IF(TP.AND.CONVG)  CALL CPHS                                       EQLM 403
      ITNUMB = ITN                                                      EQLM 404
      GO TO 43                                                          EQLM 405
C                                                                       EQLM 406
C     CALCULATE EQUILIBRIUM PROPERTIES                                  EQLM 407
C                                                                       EQLM 408
C                                                                       EQLM 409
 1171 DLVPT(NPT) = -1.                                                  EQLM 410
      DLVTP(NPT) = 1.                                                   EQLM 411
      CPR(NPT) = CPSUM                                                  EQLM 412
      GO TO 199                                                         EQLM 413
  171 DLVPT(NPT) = -1. + X(IQ1)                                         EQLM 414
      IF(JLIQ.EQ.0)  GO TO 199                                          EQLM 415
      EN(JSOL,NPT) = ENSOL                                              EQLM 416
      IUSE(JLIQ) = -IUSE(JLIQ)                                          EQLM 417
      HSUM(NPT) = HSUM(NPT)+EN(JLIQ,NPT)*(HO(JLIQ)-HO(JSOL))            EQLM 418
      IQ1 = IQ1+1                                                       EQLM 419
      GAMMAS(NPT) = -1./DLVPT(NPT)                                      EQLM 420
      GO TO 186                                                         EQLM 421
  199 GAMMAS(NPT) = -1./(DLVPT(NPT)+(DLVTP(NPT)**2)*ENN/CPR(NPT))       EQLM 422
  186 TTT(NPT) = TT                                                     EQLM 423
      ENNL = ENL                                                        EQLM 424
      PPP(NPT) = PP                                                     EQLM 425
      VLM(NPT) = RR*ENN*TT/(101.325*PP)                                 EQLM 426
      HSUM(NPT) = HSUM(NPT)*TT                                          EQLM 427
      WM(NPT) = 1./ENN                                                  EQLM 428
      IF(TRACE.EQ.0.)  GO TO 200                                        EQLM 429
```

```
      DO 1200  J=1,NS                                                   EQLM 430
      IF(IUSE(J).NE.0)  GO TO 1200                                      EQLM 431
      IF(ENLN(J).GT.-87.)  EN(J,NPT)=E**ENLN(J)                         EQLM 432
 1200 CONTINUE                                                          EQLM 433
  200 IF(.NOT.RITE)  GO TO 863                                          EQLM 434
      WRITE(6,201) NPT,           PP,TT,HSUM(NPT),SSUM(NPT),WM(NPT),CPR(NPEQLM 435
     1T),DLVPT(NPT),DLVTP(NPT),GAMMAS(NPT),VLM(NPT)                     EQLM 436
  201 FORMAT (7HOPOINT=I3,3X,2HP=E13.6,3X,2HT=E13.6,3X,4HH/R=E13.6,3X,4HEQLM 437
     1S/R=E13.6//3X,3HMW=E13.6,3X,5HCP/R=E13.6,3X,6HDLVPT=E13.6,3X,6HDLVEQLM 438
     2TP=E13.6,3X,9HGAMMA(S)=E13.6,3X,2HV=,E13.6)                       EQLM 439
  863 IF(TT.GE.TLOW.AND.TT.LE.THIGH.OR.SHOCK)  GO TO 1000               EQLM 440
      WRITE(6,306)  TT,NPT                                              EQLM 441
  306 FORMAT(17HOTHE TEMPERATURE=E12.4,26H IS OUT OF RANGE FOR POINT,I5)EQLM 442
      IF(TT.GE.TLOW/1.5.AND.TT.LE.THIGH*1.25)  GO TO 1000               EQLM 443
      NPT = NPT+1                                                       EQLM 444
C                                                                       EQLM 445
C     ERROR, SET TT=0                                                   EQLM 446
C                                                                       EQLM 447
  873 TT=0.                                                             EQLM 448
      NPT = NPT-1                                                       EQLM 449
 1000 RETURN                                                            EQLM 450
      END                                                               EQLM 451
```

```
      SUBROUTINE CPHS
C     CALCULATES THERMODYNAMIC PROPERTIES FOR INDIVIDUAL SPECIES
C
C  THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR
C     IBM 360 MACHINES ONLY
C
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN
C
C
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCH
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM
C
      EQUIVALENCE (J,JS1)
C
      K = 1
      IF(TT.LE.TMID)K = 2
      KK = 0
      CPSUM=0.
   90 IF(COEF(K,1,J).NE.0.)GO TO 97
      IF (IUSE(J).LT.0) GO TO 100
C
C  IF COEFFICIENTS ARE ZERO, USE OTHER TEMPERATURE INTERVAL
C
      KK = K
      K = 1
      IF (KK.EQ.1) K = 2
   97 S(J) =  ((((COEF(K,5,J)/4.)*TT+ COEF(K,4,J)/3.)*TT+ COEF(K,3,J)/2.
     1)* TT+COEF(K,2,J))*TT+ COEF(K,1,J)*TLN + COEF(K,7,J)
      HO(J) = ((((COEF(K,5,J)/5.)*TT+ COEF(K,4,J)/4.)*TT+ COEF(K,3,J)/3.
     1  ) *TT+ COEF(K,2,J)/2.)*TT+ COEF(K,1,J) + COEF(K,6,J)/TT
      CPSUM= CPSUM+((((COEF(K,5,J)*TT+ COEF(K,4,J))*TT+ COEF(K,3,J))*TT
     1     + COEF(K,2,J))*TT+ COEF(K,1,J))*EN(J,NPT)
      IF (KK.EQ.0) GO TO 100
      K = KK
      KK = 0
  100 IF(J.EQ.NS) GO TO 200
      J=J+1
      GO TO 90
  200 RETURN
      END
```

```
      SUBROUTINE MATRIX
C
C
      DOUBLE PRECISION G,X
      LOGICAL HP,SP,TP,IDEBUG,CONVG,NEWR,VOL,UV,SV,TV,LOGV
C
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR
C     IBM 360 MACHINES ONLY
C
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN
      DOUBLE PRECISION H,F,SS,TERM1,TERM,SSS
C
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)
     2 ,VLM(13),TOTN(13)
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE
      COMMON /DOUBLE/ G(20,21), X(20)
      CCMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM
     1  ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCH
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM
C
      EQUIVALENCE (NLM,L),(TP,TV),(SV,SP),(UV,HP),(CPCVEQ,LOGV)
C
      IQ2 = IQ1 + 1
      IQ3 = IQ2 + 1
      KMAT = IQ3
      IF(.NOT.CONVG.AND.TP) KMAT = IQ2
      IMAT = KMAT - 1
C
C     CLEAR MATRIX STORAGES TO ZERO
C
      DO 211 I=1,IMAT
      DO 211 K=1,KMAT
      G(I,K)= 0.0D0
  211 CONTINUE
      G(IC2,IQ1) = 0.
      SSS = 0.
      HSUM(NPT) = 0.
C
C     BEGIN SET UP OF ITERATION MATRIX
C
      KK = L
```

```
      DO 65  J=1,NS
      IF(IUSE(J).LT.0)  GO TO 65
      H=HO(J)*EN(J,NPT)
      IF(IUSE(J).GT.0)  GO TO 70
      F = (HO(J)-S(J)+ENLN(J)+TM)*EN(J,NPT)
      SS = H-F
      TERM1 = H
      IF (KMAT .EQ. IQ2) TERM1 = F
      DO 55 I = 1, L
C
C     CALCULATE THE ELEMENTS  R(I,K)
C
      IF (A(I,J) .EQ. 0.) GO TO 55
      TERM= A(I,J)*EN(J,NPT)
      DO 15  K=I, L
      G(I,K)= G(I,K) + A(K,J)*TERM
   15 CONTINUE
C
      G(I,IQ1)=G(I,IQ1)+TERM
      G(I,IQ2)=G(I,IQ2)+A(I,J)*TERM1
      IF (CONVG .OR.TP) GO TO 55
      G(I,IQ3)= G(I,IQ3)+A(I,J)*F
      IF (SP) G(IQ2,I) = G(IQ2,I) + A(I,J)*SS
   55 CONTINUE
      IF (KMAT .EQ. IQ2) GO TO 64
      IF(CONVG.OR.HP) GO TO 59
      G(IQ2,IQ1) = G(IQ2,IQ1) + SS
      G(IQ2,IQ2)=G(IQ2,IQ2)+HO(J)*SS
      G(IQ2,IQ3) = G(IQ2,IQ3)+(S(J) - ENLN(J)-TM)*F
      GO TO 62
   59 G(IQ2,IQ2)=G(IQ2,IQ2)+HO(J)*H
      IF (CONVG) GO TO 64
      G(IQ2,IQ3)=G(IQ2,IQ3)+HO(J)*F
   62 G(IQ1,IQ3)=G(IQ1,IQ3)+F
   64 G(IQ1,IQ2)=G(IQ1,IQ2)+TERM1
      GO TO 65
C
C    CONDENSED SPECIES
C
   70 KK = KK + 1
      DO 75  I = 1,L
      G(I,KK) = A(I,J)
      G(I,KMAT) = G(I,KMAT) - A(I,J)*EN(J,NPT)
   75 CONTINUE
      G(KK,IQ2) = HO(J)
      G(KK,KMAT) = HO(J) - S(J)
      HSUM(NPT) = HSUM(NPT)+ H
      IF(.NOT.SP) GO TO 65
      SSS = SSS + S(J)*EN(J,NPT)
      G(IQ2,KK) = S(J)
```

```
   65 CONTINUE
      SSS = SSS + G(IQ2,IQ1)
      HSUM(NPT) = HSUM(NPT) + G(IQ1,IQ2)
      G(IQ1,IQ1) = SUMN - ENN
C
C     REFLECT SYMMETRIC PORTIONS OF THE MATRIX
C
      ISYM = IQ1
      IF(HP.OR.CONVG)ISYM=IQ2
      DO 102 I=1,ISYM
      DO 102 J=I,ISYM
      G(J,I)=G(I,J)
  102 CONTINUE
C
C     COMPLETE THE RIGHT HAND SIDE
C
      IF(.NOT.CONVG)  GO TO 140
      IF(.NOT.LOGV)  GO TO 175
C
C     LOGV = .TRUE.-- SET UP MATRIX TO SOLVE FOR DLVPT
C
      G(IQ1,IQ2) = ENN
      IQ = IQ1 - 1
      DO 135 I = 1,IQ
      G(I,IQ2) = G(I,IQ1)
  135 CONTINUE
      GO TO 175
  140 DO 145 I=1,L
      X(I)=BO(I)-G(I,IQ1)
      G(I,KMAT) = G(I,KMAT)+X(I)
  145 CONTINUE
      G(IQ1,KMAT) = G(IQ1,KMAT)+ENN-SUMN
C
C     COMPLETE ENERGY ROW AND TEMPERATURE COLUMN
C
      IF (KMAT .EQ. IQ2) GO TO 185
      IF (SP)ENERGY = SO+ENN-SUMN - SSS
      IF(HP)ENERGY=HSUBO/TT - HSUM(NPT)
      G(IQ2,IQ3)=G(IQ2,IQ3)+ ENERGY
  175 G(IQ2,IQ2)= G(IQ2,IQ2)+CPSUM
  185 IF(.NOT.VOL.OR.CONVG) GO TO 1000
C
C     CONSTANT VOLUME MATRIX
C
      IQ =IQ1-1
      IF(KMAT.EQ.IQ2) GO TO 230
      DO 220 I=1,IQ
      G(IQ1,I) = G(IQ2,I)-G(IQ1,I)
      G(I,IQ1) = G(I,IQ2)-G(I,IQ1)
      G(I,IQ2) = G(I,IQ3)
```

```
  220 CONTINUE                                                    MATX 150
      G(IQ1,IQ1) = G(IQ2,IQ2)-G(IQ1,IQ2)-G(IQ2,IQ1)              MATX 151
      G(IQ1,IQ2) = G(IQ2,IQ3)-G(IQ1,IQ3)                         MATX 152
      IF (UV) G(IQ1,IQ2) = G(IQ1,IQ2) + ENN                      MATX 153
      GO TO 260                                                   MATX 154
  230 DO 240 I=1,IQ                                               MATX 155
      G(I,IQ1) = G(I,IQ2)                                         MATX 156
  240 CONTINUE                                                    MATX 157
  260 KMAT = IMAT                                                 MATX 158
      IMAT = IMAT-1                                               MATX 159
 1000 RETURN                                                      MATX 160
      END                                                         MATX 161
```

```
      SUBROUTINE GAUSS                                                  GAUS
C                                                                       GAUS
C     SOLVE ANY LINEAR SET OF UP TO 20 EQUATIONS                        GAUS
C     NUMBER OF EQUATIONS = IMAT                                        GAUS
C                                                                       GAUS
      DOUBLE PRECISION G,X,COEFX(20),SUM,Z                              GAUS
C                                                                       GAUS
      COMMON/DOUBLE/G(20,21),X(20)                                      GAUS
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  GAUS
     1  ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ GAUS
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHGAUS
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    GAUS
C                                                                       GAUS
      DATA BIGNO/1.E+38/                                                GAUS
C                                                                       GAUS
C     BEGIN ELIMINATION OF NNTH VARIABLE                                GAUS
C                                                                       GAUS
      IUSE1 = IMAT+1                                                    GAUS
    6 DO 45  NN=1,IMAT                                                  GAUS
      IF(NN-IMAT)  8,83,8                                               GAUS
   83 IF(G(NN,NN))31,23,31                                              GAUS
C                                                                       GAUS
C     SEARCH FOR MAXIMUM COEFFICIENT IN EACH ROW                        GAUS
C                                                                       GAUS
    8 DO 18  I=NN,IMAT                                                  GAUS
      COEFX(I) = BIGNO                                                  GAUS
      IF(G(I,NN).EQ.0.)  GO TO 18                                       GAUS
      COEFX(I) = 0.                                                     GAUS
      DO 10 J=NN,IUSE1                                                  GAUS
      SUM = G(I,J)                                                      GAUS
      IF(SUM.LT.0.)  SUM=-SUM                                           GAUS
      IF(J.NE.NN)  GO TO 9                                              GAUS
      Z = SUM                                                           GAUS
      GO TO 10                                                          GAUS
    9 IF(SUM.GT.COEFX(I))  COEFX(I)=SUM                                 GAUS
   10 CONTINUE                                                          GAUS
      COEFX(I) = COEFX(I)/Z                                             GAUS
   18 CONTINUE                                                          GAUS
C                                                                       GAUS
C     LOCATE ROW WITH SMALLEST MAXIMUM COEFFICIENT                      GAUS
C                                                                       GAUS
      TEMP = BIGNO                                                      GAUS
      I=0                                                               GAUS
   20 DO 22  J=NN,IMAT                                                  GAUS
      IF (COEFX(J)-TEMP) 87,22,22                                       GAUS
   87 TEMP=COEFX(J)                                                     GAUS
      I=J                                                               GAUS
   22 CONTINUE                                                          GAUS
      IF(I) 28,23,28                                                    GAUS
C                                                                       GAUS
```

```
C      INDEX I LOCATES EQUATION TO BE USED FOR ELIMINATING THE NTH
C      VARIABLE FROM THE REMAINING EQUATIONS
C
C       INTERCHANGE EQUATIONS I AND NN
C
   28 IF(NN-I) 29,31,29
   29 DO 30 J=NN,IUSE1
      Z=G(I,J)
      G(I,J)=G(NN,J)
      G(NN,J)=Z
   30 CONTINUE
C
C      DIVIDE NTH ROW BY NTH DIAGONAL ELEMENT AND ELIMINATE THE NTH
C      VARIABLE FROM THE REMAINING EQUATIONS
C
   31 K = NN + 1
      DO 36 J = K, IUSE1
      IF(G(NN,NN).EQ.0.)  GO TO 23
           G(NN,J) = G(NN,J) / G(NN,NN)
   36 CONTINUE
      IF(K-IUSE1) 88,45,88
   88 DO 44  I=K,IMAT
   40 DO 44  J = K, IUSE1
           G(I,J) = G(I,J) - G(I,NN)*G(NN,J)
   44 CONTINUE
   45 CONTINUE
C
C      BACKSOLVE FOR THE VARIABLES
C
      K = IMAT
   47 J = K + 1
      X(K) = 0.0D0
      SUM = 0.0
      IF(IMAT-J)  51,48,48
   48 DO 50  I=J,IMAT
           SUM = SUM + G(K,I)* X(I)
   50 CONTINUE
   51 X(K) = G(K,IUSE1) - SUM
      K = K - 1
      IF (K) 47,151,47
   23 IMAT = IMAT-1
  151 RETURN
      END
```

```
      SUBROUTINE OUT1
C
      DOUBLE PRECISION G,X
C
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR
C     IBM 360 MACHINES ONLY
C
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN
C
      LOGICAL EQL,FRCZ,TP,HP,SP,HPSP,TPSP,MOLES,VOL
C
      DIMENSION NV(13),Z(10,3),HEAD(15),YX(5),YN(5),FSB(3),FRHO(3)
C
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)
     2 ,VLM(13),TOTN(13)
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE
      COMMON /DOUBLE/ G(20,21), X(20)
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCH
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM
      COMMON/PERF/PCP(22),VMOC(13),SPIM(13),VACI(13),SUBAR(13),SUPAR(13)
     1   ,APP(13),AEAT(13),CSTR,EQL,FROZ,SSO,AREA,AWT
      COMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)
     1   ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMTI9,FA1,FA2
     2   ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,FO
C
      EQUIVALENCE (V,NV),(Z,HO),(IB,FB)
C
C     HEAD=(1H ,2A4,5  (A2,F8.5,3X),5  X,F7.5,F13.3,4X,A1,F10.2,F9.4)
C
      DATA HEAD/4H(1H ,4H,2A4,2H,5,4H(A2,,4HF8.5 ,4H,3X),2H,5 ,2HX,
     1     ,4HF7.5 ,4H,F13 ,4H.3,4 ,4HX,A1 ,4H,F10 ,4H.2,F ,4H9.4)/
      DATA FUEL/4HFUEL/,OXID/4HOXID/,ANT/3HANT/,OX/1HO/,IZ/2HOO/
     1       ,YN/2H,1, 2H,2, 2H,3, 2H,4, 2H,5 /,F75/4HF7.5/
     2       ,YX/3H,57,3H,44,3H,31,3H,18,2H,5 /,F73/4HF7.3/
      DATA FRHO/4HRHO,,4H G/C,1HC/
C
C     IF(KASE.NE.0) WRITE (6,3) KASE
    3 FORMAT (9H CASE NO. ,I8)
C     IF(.NOT.MOLES) WRITE(6,5)
    5 FORMAT (77X,46HWT FRACTION   ENERGY   STATE   TEMP    DENSITY/
```

```
     1 10X,16HCHEMICAL FORMULA,51X,21H(SEE NOTE)    CAL/MOL,10X,5HDEG K,OUTP   51
     2 4X,4HG/CC )                                                      OUTP   52
C       IF(MOLES) WRITE(6,6)                                             OUTP/  53
    6 FORMAT (79X,5HMOLES,7X,  33H ENERGY    STATE   TEMP    DENSITY/    OUTP   54
     1 10X,16HCHEMICAL FORMULA,66X,7HCAL/MOL,10X,13HDEG K    G/CC )      OUTP   55
      DO 15 N=1,NREAC                                                    OUTP   56
      IF(FOX(N).NE.CX)GO TO 10                                           OUTP   57
      HD1 = OXID                                                         OUTP   58
      HD2 = ANT                                                          OUTP   59
      GO TO 11                                                           OUTP   60
   10 HD1 = FUEL                                                         OUTP   61
      HD2 = FB                                                           OUTP   62
   11 DO 13 J=1,5                                                        OUTP   63
      IF(NAME(N,J).EQ.IZ.OR.NAME(N,J).EQ.IB)  GO TO 14                   OUTP   64
   13 CONTINUE                                                           OUTP   65
      J=6                                                                OUTP   66
   14 J=J-1                                                              OUTP   67
      HEAD(3)=YN(J)                                                      OUTP   68
      HEAD(7)=YX(J)                                                      OUTP   69
      HEAD(9) = F75                                                      OUTP   70
      IF(PECWT(N).GE.10.)  HEAD(9)=F73                                   OUTP   71
C     WRITE(6,HEAD)HD1,HD2,(NAME(N,JJ),ANUM(N,JJ),JJ=1,J),PECWT(N),ENTH(OUTP/  72
C    1N), FAZ(N),RTEMP(N),DENS(N)                                        OUTP/  73
   15 CONTINUE                                                           OUTP   74
      FPC = 100./(1.+OF)                                                 OUTP   75
C     WRITE(6,20)  OF   ,FPC,EQRAT,RHOP                                  OUTP/  76
   20 FORMAT (1H0,15X, 4HO/F=, F8.4,4X,13HPERCENT FUEL=,F8.4,4X,         OUTP   77
     1  19HEQUIVALENCE RATIO= ,F7.4,4X,17HREACTANT DENSITY=,F8.4//)      OUTP   78
C                                                                        OUTP   79
      AGV = 9.80665                                                      OUTP   80
      RETURN                                                             OUTP   81
C                                                                        OUTP   82
      ENTRY OUT2                                                         OUTP   83
      FMT(4) = FMT(6)                                                    OUTP   84
C                                                                        OUTP   85
C    PRESSURE                                                            OUTP   86
C                                                                        OUTP   87
   50 IF(R.LT.10.)  GO TO 60                                             OUTP   88
      CALL EFMT(NPT,FP,PPP)                                              OUTP   89
      GO TO 64                                                           OUTP   90
   60 CALL VARFMT (PPP,NPT)                                              OUTP   91
C     WRITE (6,FMT)(FP(I),I=1,4),(PPP(J),J=1,NPT)                        OUTP/  92
C                                                                        OUTP   93
C    TEMPERATURE                                                         OUTP   94
C                                                                        OUTP   95
   64 DO 65 I=1,NPT                                                      OUTP   96
      NV(I)= TTT(I)+.5                                                   OUTP   97
   65 CONTINUE                                                           OUTP   98
      FMT(4)= FMT13                                                      OUTP   99
      FMT(5)= FMTI9                                                      OUTP  100
```

```
C     WRITE (6,FMT)(FT(I),I=1,4),(NV(J),J=1,NPT)                      OUTP/101
C                                                                     OUTP 102
C     DENSITY                                                         OUTP 103
C                                                                     OUTP 104
      DO 70  I=1,NPT                                                  OUTP 105
      IF(VLM(I).NE.0.)  V(I)=1./VLM(I)                                OUTP 106
   70 CONTINUE                                                        OUTP 107
      CALL EFMT(NPT,FRHO,V)                                           OUTP 108
C                                                                     OUTP 109
C    ENTHALPY                                                         OUTP 110
C                                                                     OUTP 111
      DO 75 I=1,NPT                                                   OUTP 112
      V(I) = HSUM(I) * R                                              OUTP 113
   75 CONTINUE                                                        OUTP 114
      FMT(5)= FB                                                      OUTP 115
      IF(R.LT.10.) GO TO 76                                           OUTP 116
      CALL EFMT(NPT,FH,V)                                             OUTP 117
      FMT(7) = F1                                                     OUTP 118
      GO TO 77                                                        OUTP 119
   76 FMT(7) = F1                                                     OUTP 120
C     WRITE (6,FMT)(FH(I),I=1,4),(V(J),J=1,NPT)                       OUTP/121
C                                                                     OUTP 122
C    ENTROPY                                                          OUTP 123
C                                                                     OUTP 124
      FMT(7)=F4                                                       OUTP 125
   77 DO 78 I=1,NPT                                                   OUTP 126
      V(I) = SSUM(I) * R                                              OUTP 127
   78 CONTINUE                                                        OUTP 128
C     WRITE (6,FMT)(FS(I),I=1,4),(V(J),J=1,NPT)                       OUTP/129
C     WRITE (6,80)                                                    OUTP/130
   80 FORMAT ( 1H )                                                   OUTP 131
C                                                                     OUTP 132
C    MOLECULAR WEIGHT                                                 OUTP 133
C                                                                     OUTP 134
      FMT(7)= F3                                                      OUTP 135
C     WRITE (6,FMT)(FM(I),I=1,4),(WM(J),J=1,NPT)                      OUTP/136
C                                                                     OUTP 137
C    (DLV/DLP)T                                                       OUTP 138
C                                                                     OUTP 139
      FMT(7)=F5                                                       OUTP 140
C     IF(EQL) WRITE(6,FMT)(FV(I),I=1,4),(DLVPT(J),J=1,NPT)            OUTP/141
C                                                                     OUTP 142
C    (DLV/DLT)P                                                       OUTP 143
C                                                                     OUTP 144
      FMT(7)= F4                                                      OUTP 145
C     IF(EQL) WRITE(6,FMT)(FD(I),I=1,4),(DLVTP(J),J=1,NPT)            OUTP/146
C                                                                     OUTP 147
C    HEAT CAPACITY                                                    OUTP 148
C                                                                     OUTP 149
      IF(R.GT.10.) FMT(7)=F1                                          OUTP 150
```

```
      DO 85  I=1,NPT                                                    OUTP 151
      V(I) = CPR(I) * R                                                 OUTP 152
   85 CONTINUE                                                          OUTP 153
C     WRITE(6,FMT)(FC(I),I=1,4),(V(J),J=1,NPT)                          OUTP/154
C                                                                       OUTP 155
C   GAMMA(S)                                                            OUTP 156
C                                                                       OUTP 157
      FMT(7) = F4                                                       OUTP 158
C     WRITE(6,FMT)(FG(I),I=1,4),(GAMMAS(J),J=1,NPT)                     OUTP/159
C                                                                       OUTP 160
C   SONIC VELOCITY                                                      OUTP 161
C                                                                       OUTP 162
      FMT(7)= F1                                                        OUTP 163
      DO 95 I = 1,NPT                                                   OUTP 164
      SONVEL(I) =      (RR*GAMMAS(I)*TTT(I)/WM(I))**.5                  OUTP 165
   95 CONTINUE                                                          OUTP 166
C     WRITE(6,FMT)(FL(I),I=1,4),(SONVEL(J),J=1,NPT)                     OUTP/167
      RETURN                                                            OUTP 168
C                                                                       OUTP 169
      ENTRY OUT3                                                        OUTP 170
C                                                                       OUTP 171
      TRA = 5.E-6                                                       OUTP 172
      IF(TRACE.NE.0.)  TRA= TRACE                                       OUTP 173
      IF(.NOT.EQL)  GO TO 331                                           OUTP 174
C                                                                       OUTP 175
C   MOLE FRACTIONS - EQUILIBRIUM                                        OUTP 176
C                                                                       OUTP 177
C     WRITE (6,80)                                                      OUTP/178
      FMT(7)= F5                                                        OUTP 179
C     WRITE(6,310)                                                      OUTP/180
  310 FORMAT(15HOMOLE FRACTIONS //)                                     OUTP 181
      DO 330 K=1,NS                                                     OUTP 182
      DO 315 I=1,NPT                                                    OUTP 183
      V(I) = EN(K,I)/TOTN(I)                                            OUTP 184
  315 CONTINUE                                                          OUTP 185
      DO 316 I=1,NPT                                                    OUTP 186
      IF(TRACE.EQ.0.)  GO TO 317                                        OUTP 187
      IF(V(I).GE.TRACE)  GO TO 325                                      OUTP 188
  317 IF(V(I).GE.(5.E-6))  GO TO 320                                    OUTP 189
  316 CONTINUE                                                          OUTP 190
      GO TO 330                                                         OUTP 191
C 320 WRITE (6,FMT) SUB(K,1),SUB(K,2),SUB(K,3),FB,(V(I),I=1,NPT)        OUTP/192
  320 CONTINUE                                                          OUTPA192
      GO TO 330                                                         OUTP 193
  325 FSB(1) = SUB(K,1)                                                 OUTP 194
      FSB(2) = SUB(K,2)                                                 OUTP 195
      FSB(3) = SUB(K,3)                                                 OUTP 196
      CALL EFMT(NPT,FSB,V)                                              OUTP 197
  330 CONTINUE                                                          OUTP 198
C 331 WRITE(6,335) TRA                                                  OUTP/199
```

```
  331 CONTINUE                                                          OUTPA199
  335 FORMAT(83HOADDITIONAL PRODUCTS WHICH WERE CONSIDERED BUT WHOSE MOLOUTP 200
     1E FRACTIONS WERE LESS THAN ,E12.5,28H FOR ALL ASSIGNED CONDITIONS/OUTP 201
     2/)                                                                OUTP 202
      LINE= 0                                                           OUTP 203
      NN = 1                                                            OUTP 204
      IF(EQL)   NN=NPT                                                  OUTP 205
      DO 350 K=1,NS                                                     OUTP 206
      DO 340  I=1,NN                                                    OUTP 207
      IF((EN(K,I)/TOTN(I)).GE.TRA)   GO TO 343                          OUTP 208
  340 CONTINUE                                                          OUTP 209
      LINE= LINE+1                                                      OUTP 210
      Z(LINE,1)= SUB(K,1)                                               OUTP 211
      Z(LINE,2)= SUB(K,2)                                               OUTP 212
      Z(LINE,3)= SUB(K,3)                                               OUTP 213
  343 IF ((LINE.NE.10) .AND. K.NE.NS) GO TO 350                         OUTP 214
      IF (LINE.EQ.0) GO TO 1000                                         OUTP 215
C     WRITE(6,345) (Z(LN,1),Z(LN,2),Z(LN,3),LN=1,LINE)                  OUTP/216
  345 FORMAT (10(1X,3A4))                                               OUTP 217
      LINE= 0                                                           OUTP 218
  350 CONTINUE                                                          OUTP 219
C     IF(.NOT.MOLES) WRITE(6,360)                                       OUTP/220
  360 FORMAT(78HONOTE. WEIGHT FRACTION OF FUEL IN TOTAL FUELS AND OF OXIOUTP 221
     2DANT IN TOTAL OXIDANTS    )                                       OUTP 222
 1000 RETURN                                                            OUTP 223
      END                                                               OUTP 224
```

```
      SUBROUTINE VARFMT(V,NPT)
C
      DIMENSION V(13)
C
      CCMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)
     1  ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMTI9,FA1,FA2
     2  ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,F0
C
      DO 45 I=1,NPT
      K= 2*I+3
      FMT(K) = F4
      IF (V(I).GE.10.) FMT(K) = F3
      IF (V(I).GE.100.) FMT(K) = F2
      IF (V(I).GE.10000.)FMT(K) = F1
      IF (V(I).GE.1000000.) FMT(K) = F0
   45 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE EFMT(NPT,AA,V)
C
      DIMENSION AA(3), V(13), W(13), NE(13), FRMT(7)
C
      COMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)
     1 ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMTI9,FA1,FA2
     2 ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,FO
C
      DATA FRMT/3H(1H,4H,3A4,4H,11X,4H,13(,4HF7.4,4H,I2),1H)/,F63/4HF6.3
     1/,FI3/4H,I3)/,F74/4HF7.4/,FI2/4H,I2)/,F11X/4H,11X/,F2X/3H,2X/
C
      FRMT(5) = F74
      FRMT(6) = FI2
      J1 = 1
      FRMT(3) = F2X
      IF(FMT(4).NE.FMT9X)  GO TO 130
      J1 = 2
      FRMT(3) = F11X
  130 DO 145  I=J1,NPT
      IF(V(I).NE.0.) GO TO 140
      W(I) = 0.
      NE(I) = 0.
      GO TO 145
  140 EE = ALOG10(ABS(V(I)))
      NE(I) = EE
      FE = NE(I)
      IF(EE.LE.0..AND.FE.NE.EE)  NE(I)=NE(I)-1
      IF(IABS(NE(I)).LT.10)  GO TO 144
      FRMT(5) = F63
      FRMT(6) = FI3
  144 W(I) = V(I)/10.**NE(I)
  145 CONTINUE
C     WRITE(6,FRMT)  (AA(I),I=1,3),(W(J), NE(J),J=J1,NPT)
 1000 RETURN
      END
```

```
      SUBROUTINE THERMP
C
C   THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR
C      IBM 360 MACHINES ONLY
C
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN
C
      LOGICAL HP,SP,TP,UV,SV,NEWR,IONS,MOLES,FROZ,EQL,PSIA,RKT,VOL,TV
     1 ,CALCH
C
      DIMENSION VL(26)
C
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)
     2 ,VLM(13),TOTN(13)
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCH
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM
      COMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)
     1   ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMTI9,FA1,FA2
     2   ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,FO
C
      EQUIVALENCE (K,ISV ),(VL,P),(UV,HP),(TP,TV),(SP,SV)
C
      DATA FUU/4HU, C/
C
      IF(T(1).EQ.0.)  T(1) = 3800.
C
      IOF = 0
   95 IOF = IOF+1
      OF = OXF(IOF)
      CALL NEWOF
      IF(TT.EQ.0..AND.CALCH)  RETURN
C
C     SET ASSIGNED P OR VOLUME
C
      DO 903  IP=1,NP
      PP = P(IP)
      VLM(NPT) = VL(IP)
C
C     SET ASSIGNED T
```

```
C                                                                          THRP  51
      DO 902 IT=1,NT                                                       THRP  52
      TT = T(IT)                                                           THRP  53
      CALL EQLBRM                                                          THRP  54
      IF(TT.NE.0.)  GO TO 800                                              THRP  55
      IF(NPT.EQ.0)  GO TO 1000                                             THRP  56
  800 K = 0                                                                THRP  57
      IF(IP.EQ.NP.AND.IT.EQ.NT.OR.TT.EQ.0.)  GO TO 860                     THRP  58
      K = NPT                                                              THRP  59
      IF(NPT.NE.13)  GO TO 870                                             THRP  60
  860 IF(.NOT.HP)  WRITE(6,5)                                              THRP  61
    5 FORMAT(1H1,41X,48HTHERMODYNAMIC EQUILIBRIUM PROPERTIES AT ASSIGNEDTHRP  62
     1)                                                                    THRP  63
      IF(HP)  WRITE(6,6)                                                   THRP  64
    6 FORMAT(1H1,36X,59HTHERMODYNAMIC EQUILIBRIUM COMBUSTION PROPERTIES THRP  65
     1AT ASSIGNED  )                                                       THRP  66
      IF(.NOT.VOL)  GO TO 861                                              THRP  67
      IF(UV)  WRITE(6,10)                                                  THRP  68
   10 FORMAT(1H0,62X,7H VOLUME /)                                          THRP  69
      IF(TV)  WRITE(6,11)                                                  THRP  70
   11 FORMAT(1H0,54X,22HTEMPERATURE AND VOLUME/)                           THRP  71
      IF(SV)  WRITE(6,12)                                                  THRP  72
   12 FORMAT(1H0,56X,18HENTROPY AND VOLUME/)                               THRP  73
      GO TO 862                                                            THRP  74
  861 IF(HP)  WRITE(6,20)                                                  THRP  75
   20 FORMAT(1H0,62X,10H PRESSURES  /)                                     THRP  76
      IF(TP)  WRITE(6,21)                                                  THRP  77
   21 FORMAT(1H0,53X,24HTEMPERATURE AND PRESSURE/)                         THRP  78
      IF(SP)  WRITE(6,22)                                                  THRP  79
   22 FORMAT(1H0,55X,20HENTROPY AND PRESSURE/)                             THRP  80
  862 CALL OUT1                                                            THRP  81
      WRITE (6,863)                                                        THRP  82
  863 FORMAT (25H0THERMODYNAMIC PROPERTIES//)                              THRP  83
      IF(.NOT.VOL)  GO TO 864                                              THRP  84
      FMT(4) = FMT(6)                                                      THRP  85
      IF(.NOT.UV) GO TO 864                                                THRP  86
      DO 63 I=1,NPT                                                        THRP  87
      FMT(2*I+3) = F2                                                      THRP  88
      V(I) = HSUBO*R                                                       THRP  89
   63 CONTINUE                                                             THRP  90
      WRITE(6,FMT) FUU,FH(2),FB,FB,(V(I),I=1,NPT)                          THRP  91
  864 CALL OUT2                                                            THRP  92
      CALL OUT3                                                            THRP  93
  865 IF(K.EQ.0 .AND. IOF.EQ.NOF) GO TO 1000                               THRP  94
      IF(IDEBUG.GT.13)  IDEBUG=IDEBUG-13                                   THRP  95
C     WRITE(6,868)                                                         THRP/ 96
  868 FORMAT(1H1)                                                          THRP  97
      NPT = 0                                                              THRP  98
  870 NPT = NPT + 1                                                        THRP  99
      IF(.NOT.TP.AND.TT.NE.0.)  T(1)=TT                                    THRP 100
```

```
      IF(NT.EQ.1.AND.NP.EQ.1)  GO TO 95                                 THRP 101
      IF(IP.EQ.1.AND.IT.EQ.1)  ISV=-ISV                                 THRP 102
      IF(NT.EQ.1)  GO TO 871                                            THRP 103
      IF(IT.EQ.NT.OR.TT.EQ.0.)  ISV=0                                   THRP 104
  871 CALL SAVE                                                         THRP 105
  902 CONTINUE                                                          THRP 106
  903 CONTINUE                                                          THRP 107
      GC TO 95                                                          THRP 108
 1000 RETURN                                                            THRP 109
      END                                                               THRP 110
```

```
      SUBROUTINE ROCKET                                                 ROCK   1
C     FOR VERY LOW RC/RT, THE INCREMENT ON PCPIN IN STATEMENT 1 MAY BE  ROCKA  1
C     SET TO .3, .4, OR .5 AS REQUIRED TO AVOID EXTRAPOLATION IN        ROCKB  1
C     THERMOCHEMICAL DATA TABLES.                                       ROCKC  1
C                                                                       ROCK   2
C     ROCKET PERFORMANCE                                                ROCK   3
      DOUBLE PRECISION USQ,ASQ                                          ROCK   4
C                                                                       ROCK   5
C  THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR      ROCK   6
C     IBM 360 MACHINES ONLY                                             ROCK   7
C                                                                       ROCK   8
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS                 ROCK   9
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN                           ROCK  10
C                                                                       ROCK  11
C                                                                       ROCK  12
      LOGICAL HP,SP,TP,THI,FROZ,EQL,AREA,SEQL,CALCH,TCEST               ROCK  13
C                                                                       ROCK  14
C                                                                       ROCK  15
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)       ROCK  16
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)  ROCK  17
     2 ,VLM(13),TOTN(13)                                                ROCK  18
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)   ROCK  19
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)    ROCK  20
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  ROCK  21
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)ROCK  22
     2 ,HPP(2),RHO(2),VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         ROCK  23
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)ROCK  24
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE           ROCK  25
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  ROCK  26
     1  ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ ROCK  27
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHROCK  28
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    ROCK  29
      COMMON/PERF/PCP(22),VMOC(13),SPIM(13),VACI(13),SUBAR(13),SUPAR(13)ROCK  30
     1  ,APP(13),AEAT(13),CSTR,EQL,FROZ,SSO,AREA,AWT                    ROCK  31
      DIMENSION PCPIN(20)                                               ROCKA 31
      COMMON/ONE/NDIM,IHLF,RCRT,EPSC,MGORLI,NSPPG,ICASE,NCEC71          ROCKB 31
C                                                                       ROCK  32
C     NAMELIST/RKTINP/EQL,FROZ,SUBAR,SUPAR,PCP,TCEST                    ROCK/ 33
      TCEST = .FALSE.
      ITM = 1                                                           ROCK  34
      NFZ = 1                                                           ROCK  35
      APP(1) = 1.                                                       ROCK  36
      DO 298 I=14,22                                                    ROCKA 36
  298 PCP(I)=0.                                                         ROCKB 36
C     DO 300 I=1,26                                                     ROCK/ 37
      DO 300 I=1,13                                                     ROCKA 37
      PCP(I)= 0.                                                        ROCK  38
      SUBAR(I) = 0.                                                     ROCK  39
      SUPAR(I)=0.                                                       ROCKA 39
  300 CONTINUE                                                          ROCK  40
```

```
      EQL = .TRUE.                                                      ROCK   41
      FROZ = .TRUE.                                                     ROCK   42
      IF(NCEC71.EQ.1) EQL=.FALSE.                                       ROCKA  42
      IF(NCEC71.EQ.2) FROZ=.FALSE.                                      ROCKB  42
      IF(NCEC71.EQ.3) EQL=.FALSE.                                       ROCKC  42
C     READ (5,RKTINP)                                                   ROCK/  43
      IF(NCEC71.EQ.1) GOTO 2                                            ROCKA  43
      PCPIN(1)=1.3                                                      ROCKB  43
      DO 1 I=2,10                                                       ROCKC  43
    1 PCPIN(I)=PCPIN(I-1)+.2                                            ROCKD  43
      DO 299 I=1,10                                                     ROCKE  43
  299 PCP(I)=PCPIN(I)                                                   ROCKF  43
    2 CONTINUE                                                          ROCKG  43
      NPP = 0                                                           ROCK   44
      DO 305 I=1,22                                                     ROCK   45
      IF(PCP(I).EQ.0.) GO TO 306                                        ROCK   46
      NPP = I                                                           ROCK   47
  305 CONTINUE                                                          ROCK   48
  306 NPP = NPP+2                                                       ROCK   49
  311 NSUB = 0                                                          ROCK   50
      NSUP = 0                                                          ROCK   51
      DO 320  I=1,13                                                    ROCK   52
      IF(SUBAR(I).NE.0.) NSUB=NSUB+1                                    ROCK   53
      IF(SUPAR(I).NE.0.) NSUP=NSUP+1                                    ROCK   54
  320 CONTINUE                                                          ROCK   55
C     WRITE (6,RKTINP)                                                  ROCK/  56
      SEQL = EQL                                                        ROCK   57
      IOF = 0                                                           ROCK   58
      TT = 3800.                                                        ROCK   59
C                                                                       ROCK   60
C     LOOP FOR EACH O/F                                                 ROCK   61
C                                                                       ROCK   62
  321 IT = 1                                                            ROCK   63
      IOF = IOF + 1                                                     ROCK   64
      OF = OXF(IOF)                                                     ROCK   65
      CALL NEWOF                                                        ROCK   66
      IF(CALCH.AND.TT.EQ.0.)  RETURN                                    ROCK   67
      IF(T(1).EQ.0.)  GO TO 322                                         ROCK   68
      TT = T(1)                                                         ROCK   69
C                                                                       ROCK   70
C     LOOP FOR CHAMBER PRESSURES                                        ROCK   71
  322 DO 998 IP = 1,NP                                                  ROCK   72
      ITNUM = 0                                                         ROCK   73
      AREA = .FALSE.                                                    ROCK   74
      IF(T(1).EQ.0..OR.TCEST)  HP=.TRUE.                                ROCK   75
      IF(T(1).NE.0..AND..NOT.TCEST)  TP=.TRUE.                          ROCK   76
      SP = .FALSE.                                                      ROCK   77
      EQL = .TRUE.                                                      ROCK   78
      ISUB = 1                                                          ROCK   79
      ISUP = 1                                                          ROCK   80
```

```
      PP = P(IP)
      IPP = 1
C
C     LOOP FOR PRESSURE RATIOS
C
  331 IF(EQL)  GO TO 332
      CALL FROZEN
      GO TO 1332
  332 CALL EQLBRM
C
C     TT = 0 IF NO CONVERGENCE
C
 1332 IF(TT.NE.0.)  GO TO 333
      IF(NPT.LT.1)  GO TO 1000
      GO TO 900
  333 IF(IPP.GT.1) GO TO 195
C
C     COMBUSTION CHAMBER
C
      EQL = SEQL
      TP = .FALSE.
      HP = .FALSE.
      SP = .TRUE.
      SO = SSUM(1)
      CPRF = CPSUM
  334 TMELT=0.
      ITROT = 3
      THI = .FALSE.
      APP(2)=((GAMMAS(1)+1.)/2.)**(GAMMAS(1)/(GAMMAS(1)-1.))
      PP = PPP(1)/APP(2)
      TT = 2.*TT/(GAMMAS(1)+1.)
      ISV = 1
      GO TO 870
  195 USQ = 2.*(HSUM(1)-HSUM(NPT)) * RR
      IF (IPP.GT.2) GO TO 900
C
C     THROAT
C
  190 IF(.NOT.THI)  GO TO 191
      GAMMAS(2) = 0.
      GO TO 899
  191 ASQ = GAMMAS(2)*TT*ENN*RR
C     IF(EQL)  WRITE(6,194) APP(2),TT
  194 FORMAT (7H PC/PT= , F9.6, 6H   T = , F9.2)
      IF(IDEBUG.EQ.1.OR.IDEBUG.EQ.2)  WRITE(6,923)USQ,ASQ
  923 FORMAT(5HOUSQ=,E15.8,5X,4HASQ=,E15.8)
      DH = (USQ-ASQ)/ASQ
      IF(DH.LT.0.)  DH=-DH
      IF(DH.LE.0.4E-4.OR.ITROT.EQ.0)  GO TO 899
      IF(JSOL.NE.0) GO TO 925
```

```
      IF(TMELT.EQ.O.) GO TO 192                                         ROCK 131
      DLT = ALOG(TMELT/TT)                                              ROCK 132
      DD = DLT*CPR(2)/(ENN*DLVTP(2))
      PP = PP*EXP(DD)                                                   ROCK 133
      APP(2) = P(IP)/PP                                                 ROCK 134
      THI = .TRUE.                                                      ROCK 135
      GO TO 331                                                         ROCK 136
  925 TMELT = TT                                                        ROCK 137
  192 APP(2)= APP(2)/(1.+(USQ-ASQ)/(ENN*TT*RR*(GAMMAS(2)+1.)))          ROCK 138
  193 PP = P(IP)/APP(2)                                                 ROCK 139
      ITROT = ITROT-1                                                   ROCK 140
      GO TO 331                                                         ROCK 141
  899 AWT =ENN*TT/(PP*USQ**.5)                                          ROCK 142
      PCPLT = ALOG(APP(2))                                              ROCK 143
C                                                                       ROCK 144
C                                                                       ROCK 145
  900 ISV =0                                                            ROCK 146
      AEAT(NPT) = ENN*TTT(NPT)/(PP*USQ**.5*AWT)
      IF(TT.EQ.O.)  GO TO 860                                           ROCK 147
      IF(AREA)  GO TO 800                                               ROCK 149
      IF(IPP.LT.NPP)  GO TO 859                                         ROCK 150
      IF(NSUB.EQ.O.AND.NSUP.EQ.O)   GO TO 860                           ROCK 151
      AREA = .TRUE.                                                     ROCK 152
C                                                                       ROCK 153
C     PCP ESTIMATES FOR AREA RATIOS                                     ROCK 154
C                                                                       ROCK 155
  800 IF(ITNUM.NE.O)   GO TO 810                                        ROCK 156
      DLNP = 1.                                                         ROCK 157
      ITNUM = 1                                                         ROCK 158
      ARATIO = SUBAR(ISUB)                                              ROCK 159
      IF(NSUB.LE.O)  ARATIO=SUPAR(ISUP)                                 ROCK 160
      ELN = ALOG(ARATIO)                                                ROCK 161
      IF(NSUB.LE.O)  GO TO 799                                          ROCK 162
      APPL = PCPLT/(SUBAR(ISUB)+(10.587*ELN**2+9.454)*ELN)              ROCK 163
      IF(ARATIO.LT.1.09)  APPL=.9*APPL                                  ROCK 164
      IF (ARATIO .GT.10.) APPL = APPL/ARATIO
      GO TO 859                                                         ROCK 165
  799 IF(SUPAR(ISUP).LT.2.)  GO TO 805                                  ROCK 166
      IF(ISUP.GT.1.AND.SUPAR(ISUP-1).GE.2.)  GO TO 802                  ROCK 167
      APPL = GAMMAS(2)+ELN*1.4                                          ROCK 168
      GO TO 859                                                         ROCK 169
  805 APPL = SQRT(ELN*(1.535+3.294*ELN))+PCPLT                          ROCK 170
      GO TO 859                                                         ROCK 171
C                                                                       ROCK 172
C     TEST FOR CONVERGENCE ON AREA RATIO.                               ROCK 173
C                                                                       ROCK 174
  810 CHECK = .00004                                                    ROCK 175
      IF(IDEBUG.LE.O.OR.NPT.LT.IDEBUG)  GO TO 809                       ROCK 176
      WRITE(6,1811)ITNUM,ARATIO,AEAT(NPT),APP(NPT),DLNP                 ROCK 177
 1811 FORMAT (6HOITER=,I2,5X,15HASSIGNED AE/AT=,F15.8,5X,6HAE/AT=,F15.8,ROCK 178
```

```
     15X,5HPC/P=,F15.8,5X,13HDELTA LN PCP=,F15.8)                      ROCK 179
  809 IF(ABS(AEAT(NPT)-   ARATIO) /ARATIO        .LE.CHECK)    GO TO 830 ROCK 180
      DELTAE = (AEAT(NPT)-ARATIO)/ARATIO                               ROCK 181
      IF(ABS(DLNP).LT..00004)  GO TO 830                               ROCK 182
      AEATL= ALOG(AEAT(NPT))                                           ROCK 183
      ITNUM = ITNUM+1                                                  ROCK 184
      IF(ITNUM.GT.10)  GO TO 840                                       ROCK 185
C                                                                      ROCK 186
C     IMPROVED PCP ESTIMATES.                                          ROCK 187
C                                                                      ROCK 188
      ASQ = GAMMAS(NPT)*ENN*RR*TT                                      ROCK 189
      DLNPE = GAMMAS(NPT)*USQ/(USQ-ASQ)                                ROCK 190
  802 DLNP = DLNPE*ELN-DLNPE*AEATL                                     ROCK 191
      APPL = APPL+DLNP                                                 ROCK 192
      IF(ITNUM.EQ.1)  GO TO 859                                        ROCK 193
      IF (APPL.LT.0.) APPL = .000001
      APP(NPT) = EXP(APPL)                                             ROCK 194
      PP = P(IP)/APP(NPT)                                              ROCK 195
      GO TO 331                                                        ROCK 196
C                                                                      ROCK 197
  830 ITNUM = 0                                                        ROCK 198
      AEAT(NPT) = ARATIO                                               ROCK 199
      IF(NSUB.LE.0)  GO TO 834                                         ROCK 200
      ISUB = ISUB+1                                                    ROCK 201
      IF(ISUB.LE.NSUB)  GO TO 800                                      ROCK 202
      ISUB = 1                                                         ROCK 203
      NSUB = -NSUB                                                     ROCK 204
      IF(ISUP.LE.NSUP)  GO TO 800                                      ROCK 205
      GO TO 835                                                        ROCK 206
  834 ISUP = ISUP+1                                                    ROCK 207
      IF(ISUP.LE.NSUP)  GO TO 800                                      ROCK 208
      ISUP = 1                                                         ROCK 209
  835 AREA = .FALSE.                                                   ROCK 210
      GO TO 860                                                        ROCK 211
  840 WRITE(6,841) ARATIO                                              ROCK 212
  841 FORMAT(34H0DID NOT CONVERGE FOR AREA RATIO =,F10.5)              ROCK 213
      GO TO 830                                                        ROCK 214
C                                                                      ROCK 215
C     TEST FOR OUTPUT -- END OF PCP,SUBAR,AND SUPAR SCHEDULES OR NPT=13.ROCK 216
C                                                                      ROCK 217
  859 ISV = NPT                                                        ROCK 218
      IF(NPT.NE.13)  GO TO 870                                         ROCK 219
  860 IF(EQL)  GO TO 861                                               ROCK 220
      CPR(1) = CPRF                                                    ROCK 221
      GAMMAS(1) = CPRF/(CPRF-1./WM(1))                                 ROCK 222
  861 CALL RKTOUT                                                      ROCK 223
      IF(TT.EQ.0.)   AREA=.FALSE.                                      ROCK 224
      IF(.NOT.EQL.AND.TT.EQ.0.)  WRITE(6,862)                          ROCK 225
  862 FORMAT(105H0CALCULATIONS WERE STOPPED BECAUSE NEXT POINT IS MORE TROCK 226
     1HAN 50 DEG BELOW TEMP RANGE OF A CONDENSED SPECIES)              ROCK 227
```

```
      IF (ISV.EQ.0) GO TO 990                                           ROCK 228
      IDEBUG = IDEBUG-13                                                ROCK 229
C     IF(EQL)  WRITE(6,865)                                             ROCK/230
  865 FORMAT(1H1)                                                       ROCK 231
      NPT = 2                                                           ROCK 232
C                                                                       ROCK 233
C     SET INDICES AND ESTIMATES FOR NEXT POINT.                         ROCK 234
C                                                                       ROCK 235
  870 NPT = NPT + 1                                                     ROCK 236
      IF(.NOT.EQL.AND.(ISV.NE.1.OR.SEQL))   GO TO 880                   ROCK 237
      IF(ISV.EQ.1)  ISV = -1                                            ROCK 238
      CALL SAVE                                                         ROCK 239
  880 IPP = IPP+1                                                       ROCK 240
      IF(NPT.EQ.2)  GO TO 331                                           ROCK 241
      IF(.NOT.AREA)   APP(NPT)=PCP(IPP-2)                               ROCK 242
      IF(AREA)   APP(NPT)=EXP(APPL)                                     ROCK 243
      PP = P(IP)/APP(NPT)                                               ROCK 244
      GO TO 331                                                         ROCK 245
C                                                                       ROCK 246
C     END OF PCP, SUBAR, AND SUPAR SCHEDULES.                           ROCK 247
C                                                                       ROCK 248
  990 IF(IDEBUG.LT.0)  IDEBUG=IDEBUG+13                                 ROCK 249
      IF(NSUB.LT.0)  NSUB=-NSUB                                         ROCK 250
      IF (.NOT.FROZ.OR..NOT.EQL)  GO TO 997                             ROCK 251
C                                                                       ROCK 252
C     SET UP FOR FROZEN.                                                ROCK 253
C                                                                       ROCK 254
      TT = TTT(1)                                                       ROCK 255
      IPP = 1                                                           ROCK 256
      NPT = 1                                                           ROCK 257
      ITROT = 3                                                         ROCK 258
      CALL SAVE                                                         ROCK 259
      EQL = .FALSE.                                                     ROCK 260
      ENN = 1./WM(1)                                                    ROCK 261
      ITNUM = 0
      ISUB = 1
      ISUP = 1
      GO TO 334                                                         ROCK 262
  997 NPT = 1                                                           ROCK 263
C                                                                       ROCK 264
C     ARE THERE MORE ASSIGNED,                                          ROCK 265
C      1) CHAMBER PRESSURES(IP = NP)                                    ROCK 266
C      2) CHAMBER TEMPERATURES(IT = NT)                                 ROCK 267
C      3) O/F VALUES(IOF = NOF)                                         ROCK 268
C                                                                       ROCK 269
      IF(IP.EQ.NP.AND.IT.EQ.NT.AND.IOF.EQ.NOF)  GO TO 1000              ROCK 270
C     WRITE(6,865)                                                      ROCK/271
      CALL SAVE                                                         ROCK 272
      TT = TTT(1)                                                       ROCK 273
  998 CONTINUE                                                          ROCK 274
```

```
      IF(IT.GE.NT)  GO TO 999                                          ROCK 275
      IT = IT+1                                                        ROCK 276
      TT = T(IT)                                                       ROCK 277
      GO TO 322                                                        ROCK 278
  999 IF (IOF.GE.NOF)  GO TO 1000                                      ROCK 279
      GO TO 321                                                        ROCK 280
 1000 RETURN                                                           ROCK 281
      END                                                              ROCK 282
```

```
      SUBROUTINE RKTOUT                                                 ROUT   1
C                                                                       ROUT   2
C  ROCKET PERFORMANCE PARAMETERS                                        ROUT   3
C                                                                       ROUT   4
C  THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR      ROUT   5
C     IBM 360 MACHINES ONLY                                             ROUT   6
C                                                                       ROUT   7
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS                 ROUT   8
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN                           ROUT   9
C                                                                       ROUT  10
      LOGICAL EQL,FROZ ,TP,HP,SP,SHOCK,AREA                             ROUT  11
C                                                                       ROUT  12
      DIMENSION NV(13),Z(10,4)                                          ROUT  13
C                                                                       ROUT  14
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)       ROUT  15
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)  ROUT  16
     2 ,VLM(13),TOTN(13)                                                ROUT  17
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)   ROUT  18
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)    ROUT  19
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  ROUT  20
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)ROUT  21
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         ROUT  22
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)ROUT  23
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE           ROUT  24
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  ROUT  25
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ ROUT  26
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHROUT  27
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    ROUT  28
      COMMON/PERF/PCP(22),VMOC(13),SPIM(13),VACI(13),SUBAR(13),SUPAR(13)ROUT  29
     1   ,APP(13),AEAT(13),CSTR,EQL,FROZ,SSO,AREA,AWT                   ROUT  30
      COMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)     ROUT  31
     1   ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMT19,FA1,FA2       ROUT  32
     2   ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,FO                      ROUT  33
      COMMON/ZONE/KZONE,PGPSIA,OXFUL(20),LETOUT                         ROUTA 33
      COMMON/ACSTAR/CSTAR(13)                                           ROUTB 33
C                                                                       ROUT  34
      EQUIVALENCE (V,NV),(Z,HO)                                         ROUT  35
C                                                                       ROUT  36
      DATA EXIT/4HEXIT/                                                 ROUT  37
C                                                                       ROUT  38
      IF(.NOT.EQL)  GO TO 636                                           ROUT  39
C     WRITE(6,37)                                                       ROUT/ 40
   37 FORMAT(1H1/24X,84HTHEORETICAL ROCKET PERFORMANCE ASSUMING EQUILIBRROUT  41
     1IUM COMPOSITION DURING EXPANSION           /)                     ROUT  42
      GO TO 39                                                          ROUT  43
C 636 WRITE(6,38)                                                       ROUT/ 44
  636 CONTINUE                                                          ROUTA 44
   38 FORMAT(1H1,26X,78HTHEORETICAL ROCKET PERFORMANCE ASSUMING FROZEN CROUT  45
     1OMPOSITION DURING EXPANSION          /)                           ROUT  46
      IF(NFZ.GT.1)  WRITE(6,637)NFZ                                     ROUT  47
```

```
  637 FORMAT(58X,11HAFTER POINT,I2)                                    ROUT   48
   39 IF(TTT(1).EQ.T(IT)) WRITE(6,737)                                 ROUT   49
  737 FORMAT (52X,28HAT AN ASSIGNED TEMPERATURE       )                ROUT   50
      TEM  = PPP(1)*14.696006                                          ROUT   51
C     WRITE (6,40) TEM                                                 ROUT/  52
   40 FORMAT(5HOPC = ,F8.1,5H PSIA)                                    ROUT   53
      CALL OUT1                                                        ROUT   54
      NEX = NPT - 2                                                    ROUT   55
      DO 862 I = 1,NEX                                                 ROUT   56
  862 V(I) = EXIT                                                      ROUT   57
C     WRITE(6,48) (V(I),I=1,NEX)                                       ROUT/  58
   48 FORMAT(1H0,16X,16HCHAMBER   THROAT  ,11(5X,A4))                  ROUT   59
C                                                                      ROUT   60
C     PRESSURE RATIOS                                                  ROUT   61
C                                                                      ROUT   62
      FMT(4) = FMT(6)                                                  ROUT   63
      CALL VARFMT (APP,NPT)                                            ROUT   64
C     WRITE (6,FMT) FR1,FB,FB,FB,(APP(J),J=1,NPT)                      ROUT/  65
      CALL OUT2                                                        ROUT   66
C                                                                      ROUT   67
      AGV = 9.80665                                                    ROUT   68
      DO 202 K=2,NPT                                                   ROUT   69
      SPIM(K) = (2.*RR*(HSUM(1)-HSUM(K)))**.5/AGV                      ROUT   70
C                                                                      ROUT   71
C  AW (A/W) IN UNITS OF SEC/ATM                                        ROUT   72
C                                                                      ROUT   73
      AW = RR*TTT(K)/(PPP(K)*          WM(K)*SPIM(K)*AGV**2)           ROUT   74
      IF(K.NE.2)GO TO 200                                              ROUT   75
      CSTR = 32.174*PPP(1)*AW                                          ROUT   76
      AEAT(2) = 1.                                                     ROUT   77
  200 VACI(K)=SPIM(K)+PPP(K)*AW                                        ROUT   78
      IF (SONVEL(K).NE.0.) VMOC(K)=SPIM(K)*AGV/SONVEL(K)               ROUT   79
      NV(K)= CSTR + .5                                                 ROUT   80
  202 CONTINUE                                                         ROUT   81
C                                                                      ROUT   82
C   MACH NUMBER                                                        ROUT   83
C                                                                      ROUT   84
      VMOC(1)=0.                                                       ROUT   85
      IF(GAMMAS(2).EQ.0.) VMOC(2)=0.                                   ROUT   86
      FMT(7) = F3                                                      ROUT   87
C     WRITE(6,FMT)(FN(I),I=1,4),(VMOC(J),J=1,NPT)                      ROUT/  88
C     WRITE (6,208)                                                    ROUT/  89
  208 FORMAT (1H )                                                     ROUT   90
      CALL LOAD1(NPT)                                                  ROUTA  90
C                                                                      ROUT   91
C   AREA RATIO                                                         ROUT   92
C                                                                      ROUT   93
      FMT(4) = FMT9X                                                   ROUT   94
      CALL VARFMT (AEAT,NPT)                                           ROUT   95
      FMT(5) = FB                                                      ROUT   96
```

```
C      WRITE(6,FMT)FA1,FA2,FB,FB,(AEAT(J),J=2,NPT)                      ROUT/ 97
C                                                                       ROUT   98
C   C*                                                                  ROUT   99
C                                                                       ROUT  100
       FMT(5) = FMT13                                                   ROUT  101
       FMT(6) = FMTI9                                                   ROUT  102
       FMT(7) = FB                                                      ROUT  103
C      WRITE(6,FMT)(FR(I),I=1,4),(NV(J),J=2,NPT)                        ROUT/104
       DO 1 I=2,NPT                                                     ROUTA104
    1  CSTAR(I)=NV(I)                                                   ROUTB104
C                                                                       ROUT  105
C   CF - THRUST COEFICIENT                                              ROUT  106
C                                                                       ROUT  107
       FMT(6) = FMT(8)                                                  ROUT  108
       FMT(7) = F3                                                      ROUT  109
       DO 212  I=2,NPT                                                  ROUT  110
  212  V(I)=32.174*SPIM(I)/CSTR                                         ROUT  111
C      WRITE(6,FMT)FC1,FB,FB,FB,(V(J),J=2,NPT)                          ROUT/112
C                                                                       ROUT  113
C   VACUUM IMPULSE                                                      ROUT  114
C                                                                       ROUT  115
       FMT(5) = FMT13                                                   ROUT  116
       FMT(7) = F1                                                      ROUT  117
C      WRITE(6,FMT)(FA(I),I=1,4),(VACI(J),J=2,NPT)                      ROUT/118
C                                                                       ROUT  119
C   SPECIFIC IMPULSE                                                    ROUT  120
C                                                                       ROUT  121
C      WRITE(6,FMT)(FI(I),I=1,4),(SPIM(J),J=2,NPT)                      ROUT/122
       CALL LOAD2(NPT)                                                  ROUTA122
C      WRITE (6,208)                                                    ROUT/123
       FMT(4) = FB                                                      ROUT  124
       FMT(5) = FMT13                                                   ROUT  125
       FMT(7) = F5                                                      ROUT  126
       IF(EQL)  GO TO 312                                               ROUT  127
C      WRITE(6,310)                                                     ROUT/128
  310  FORMAT(15HOMOLE FRACTIONS //)                                    ROUT  129
C                                                                       ROUT  130
C   MOLE FRACTIONS - FROZEN                                             ROUT  131
C                                                                       ROUT  132
       TRA = 5.E-6                                                      ROUT  133
       IF(TRACE.NE.0.)  TRA=TRACE                                       ROUT  134
       LINE = 0                                                         ROUT  135
       DO 430 K =1,NS                                                   ROUT  136
       V(LINE+1) = EN(K,NFZ)/TOTN(NFZ)                                  ROUT  137
       IF(V(LINE+1).LT.TRA)  GO TO 424                                  ROUT  138
       LINE = LINE+1                                                    ROUT  139
       Z(LINE,1) = SUB(K,1)                                             ROUT  140
       Z(LINE,2) = SUB(K,2)                                             ROUT  141
       Z(LINE,3) = SUB(K,3)                                             ROUT  142
       Z(LINE,4) = V(LINE)                                              ROUT  143
```

```
  424 IF (LINE.NE.4.AND.K.NE.NS)   GO TO 430                            ROUT 144
      IF (LINE.EQ.0)   GO TO 312                                        ROUT 145
C     WRITE (6,426) (Z(LN,1),Z(LN,2),Z(LN,3),Z(LN,4),LN=1,LINE)         ROUT/146
  426 FORMAT (1H ,4(3A4,F9.5,7X))                                       ROUT 147
      LINE = 0                                                          ROUT 148
  430 CONTINUE                                                          ROUT 149
  312 CALL OUT3                                                         ROUT 150
 1000 RETURN                                                            ROUT 151
      END                                                               ROUT 152
```

```
      SUBROUTINE FROZEN                                                 FROZ   1
C                                                                       FROZ   2
C      (FROZEN COMPOSITION EXPANSION ONLY)                              FROZ   3
C                                                                       FROZ   4
C  THE FOLLOWING DOUBLE PRECISION TYPE STATEMENTS ARE REQUIRED FOR      FROZ   5
C     IBM 360 MACHINES ONLY                                             FROZ   6
C                                                                       FROZ   7
      DOUBLE PRECISION HSUM,SSUM,CPR,DLVTP,DLVPT,GAMMAS                 FROZ   8
      DOUBLE PRECISION COEF,S,EN,ENLN,HO,DELN                           FROZ   9
      DOUBLE PRECISION SUMS,SUMH,SS                                     FROZ  10
C                                                                       FROZ  11
      LOGICAL EQL,FROZ,CONVG,SP,HP,VOL                                  FROZ  12
C                                                                       FROZ  13
      COMMON/POINTS/HSUM(13),SSUM(13),CPR(13),DLVTP(13),DLVPT(13)       FROZ  14
     1 ,GAMMAS(13),P(26),T(26),V(13),PPP(13),WM(13),SONVEL(13),TTT(13)  FROZ  15
     2 ,VLM(13),TOTN(13)                                                FROZ  16
      COMMON/SPECES/COEF(2,7,150),S(150),EN(150,13),ENLN(150),HO(150)   FROZ  17
     1 ,DELN(150),A(15,150),SUB(150,3),IUSE(150),TEMP(50,2),SLN(150)    FROZ  18
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  FROZ  19
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)FROZ  20
     2 ,HPP(2),RHO(2),VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         FROZ  21
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)FROZ  22
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE           FROZ  23
      COMMON/INDX/ IDEBUG,CONVG,TP,HP,SP,ISV, NPP, MOLES,NP,NT,NPT,NLM  FROZ  24
     1   ,NS,KMAT,IMAT,IQ1,NOF,NOMIT,IP,NEWR,NSUB,NSUP,ITM,CPCVFR,CPCVEQ FROZ  25
     2 ,IONS,NC,NSERT,JSOL,JLIQ,KASE,NREAC,IC,JS1,VOL,SHOCK,IT,NFZ,CALCHFROZ  26
     3 ,IQSAVE,LSAVE,ISUP,ISUB,ITNUM                                    FROZ  27
      COMMON/PERF/PCP(22),VMOC(13),SPIM(13),VACI(13),SUBAR(13),SUPAR(13)FROZ  28
     1   ,APP(13),AEAT(13),CSTR,EQL,FROZ,SSO,AREA,AWT                   FROZ  29
      COMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)     FROZ  30
     1   ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMTI9,FA1,FA2       FROZ  31
     2   ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,FO                      FROZ  32
      CONVG = .FALSE.                                                   FROZ  33
      TLN = ALOG(TT)                                                    FROZ  34
      DO 51  ITER=1,8                                                   FROZ  35
      SUMS = 0.                                                         FROZ  36
      SUMH = 0.                                                         FROZ  37
      JS1 = ITM                                                         FROZ  38
      NNN = NPT                                                         FROZ  39
      NPT = NFZ                                                         FROZ  40
      CALL CPHS                                                         FROZ  41
      CC = CPSUM                                                        FROZ  42
   55 NPT = NNN                                                         FROZ  43
      DO 60 J=ITM,NS                                                    FROZ  44
      IF (EN(J,NFZ).EQ.0.) GO TO 60                                     FROZ  45
      SS = S(J)                                                         FROZ  46
      PMN = PP*WM(NFZ)*EN(J,NFZ)                                        FROZ  47
      IF(IUSE(J).EQ.0)  SS=SS-ALOG(PMN)
      SUMS = SUMS+SS*EN(J,NFZ)                                          FROZ  48
      IF (CONVG)        SUMH=SUMH+HO(J)*EN(J,NFZ)                       FROZ  49
```

```
   60 CONTINUE                                                          FROZ
      IF (CONVG) GO TO 81                                               FROZ
      DLNT=(SUMS-SO)/CC                                                 FROZ
      TLN=TLN-DLNT                                                      FROZ
      IF(DLNT.LT.0.) DLNT=-DLNT                                         FROZ
      IF(DLNT.LT.0.5E-4)  CONVG=.TRUE.                                  FROZ
      TT = EXP(TLN)                                                     FROZ
   51 CONTINUE                                                          FROZ
      WRITE(6,70)                                                       FROZ
   70 FORMAT(40HOFROZEN DID NOT CONVERGE IN 8 ITERATIONS)               FROZ
      GO TO 903                                                         FROZ
   81 TTT(NPT)= TT                                                      FROZ
      SSUM(NPT)= SUMS                                                   FROZ
      HSUM(NPT)= TT*SUMH                                                FROZ
      GAMMAS(NPT)= CPSUM/(CPSUM-1./WM(NFZ))                             FROZ
      VLM(NPT) = RR*TT/(WM(NFZ)*101.325*PP)                             FROZ
      WM(NPT) = WM(NFZ)                                                 FROZ
C                                                                       FROZ
      DLVPT(NPT) = -1.                                                  FROZ
      DLVTP(NPT) = 1.                                                   FROZ
      TOTN(NPT) = TOTN(NFZ)                                             FROZ
      PPP(NPT) = PP                                                     FROZ
      CPR(NPT)= CPSUM                                                   FROZ
      IF (TT.LT.(TLOW-150.))GO TO 903                                   FROZ
      IF(NC.EQ.0) GO TO 1000                                            FROZ
      INC =0                                                            FROZ
      DO 901  I=ITM,NS                                                  FROZ
      IF(IUSE(I).EQ.0.OR.IUSE(I).EQ.-10000) GO TO 901                   FROZ
      INC = INC+1                                                       FROZ
      IF (EN(I,NFZ).EQ.0.) GO TO 901                                    FROZ
      IF(TT.LT.(TEMP(INC,1)-50.).OR.TT.GT.(TEMP(INC,2)+50.))GO TO 903   FROZ
  901 CONTINUE                                                          FROZ
      GO TO 1000                                                        FROZ
  903 TT=0.                                                             FROZ
      NPT= NPT-1                                                        FROZ
 1000 RETURN                                                            FROZ
      END                                                               FROZ
```

```
      BLOCK DATA                                                        BLOK   1
C                                                                       BLOK   2
      DIMENSION ATEM(3,50)                                              BLOK   3
C                                                                       BLOK   4
      COMMON/MISC/ENN,SUMN,TT,SO,ATOM(3,101),LLMT(15),BO(15),BOP(15,2)  BLOK   5
     1 ,TM,TLOW,TMID,THIGH,PP,CPSUM,OF,EQRAT,FPCT,R,RR,HSUBO,AC(2),AM(2)BLOK   6
     2 ,HPP(2),RH(2), VMIN(2),VPLS(2),WP(2),DATA(22),NAME(15,5)         BLOK   7
     3 ,ANUM(15,5),PECWT(15),ENTH(15),FAZ(15),RTEMP(15),FOX(15),DENS(15)BLOK   8
     4 ,RHOP,RMW(15),TLN,CR,OXF(15),ENNL,ENSAVE,ENLSAV ,TRACE           BLOK   9
      CCMMON/OUPT/FMT(30),FP(4),FT(4),FH(4),FS(4),FM(4),FV(4),FD(4)     BLOK  10
     1  ,FC(4),FG(4),FB,FMT13,F1,F2,F3,F4,F5,FL(4),FMTI9,FA1,FA2        BLOK  11
     2  ,FR1,FC1,FN(4),FR(4),FA(4),FI(4),FMT9X,F0                       BLOK  12
C                                                                       BLOK  13
      EQUIVALENCE (ATOM(1,52),ATEM)                                     BLOK  14
C                                                                       BLOK  15
C  ATOMIC SYMBOLS, WEIGHTS, AND VALENCES                                BLOK  16
C                                                                       BLOK  17
      DATA ATOM/                                                        BLOK  18
     A    2HH ,  1.00797, 1.,    2HHE,  4.0026, 0.,    2HLI,  6.939   , 1., BLOK  19
     B    2HBE,  9.0122 , 2.,    2HB ,  10.811 , 3.,   2HC ,  12.01115, 4., BLOK  20
     C    2HN ,  14.0067 , 0.,   2HO ,  15.9994,-2.,   2HF ,  18.9984 ,-1., BLOK  21
     D    2HNE,  20.183  , 0.,   2HNA,  22.9898, 1.,   2HMG,  24.312  , 2., BLOK  22
     E    2HAL,  26.9815 , 3.,   2HSI,  28.086 , 4.,   2HP ,  30.9738 , 5., BLOK  23
     F    2HS ,  32.064  , 4.,   2HCL,  35.453 ,-1.,   2HAR,  39.948  , 0., BLOK  24
     G    2HK ,  39.102  , 1.,   2HCA,  40.080 , 2.,   2HSC,  44.956  , 3., BLOK  25
     H    2HTI,  47.900  , 4.,   2HV ,  50.942 , 5.,   2HCR,  51.996  , 3., BLOK  26
     I    2HMN,  54.9380 , 2.,   2HFE,  55.847 , 3.,   2HCO,  58.9332 , 2., BLOK  27
     J    2HNI,  58.710  , 2.,   2HCU,  63.540 , 2.,   2HZN,  65.370  , 2., BLOK  28
     K    2HGA,  69.720  , 3.,   2HGE,  72.590 , 4.,   2HAS,  74.9216 , 3., BLOK  29
     L    2HSE,  78.960  , 4.,   2HBR,  79.909 ,-1.,   2HKR,  83.800  , 0., BLOK  30
     M    2HRB,  85.47   , 1.,   2HSR,  87.620 , 2.,   2HY ,  88.905  , 3., BLOK  31
     N    2HZR,  91.220  , 4.,   2HNB,  92.906 , 5.,   2HMO,  95.94   , 6., BLOK  32
     O    2HTC,  99.000  , 7.,   2HRU,101.070 , 3.,    2HRH,102.905  , 3., BLOK  33
     P    2HPD,106.400   , 2.,   2HAG,107.870 , 1.,    2HCD,112.400  , 2., BLOK  34
     Q    2HIN,114.820   , 3.,   2HSN,118.690 , 4.,    2HSB,121.750  , 3. /BLOK  35
      DATA ATEM/                                                        BLOK  36
     R    2HTE,127.600   , 4.,   2HI ,126.9044,-1.,    2HXE,131.300  , 0., BLOK  37
     S    2HCS,132.905   , 1.,   2HBA,137.340 , 2.,    2HLA,138.910  , 3., BLOK  38
     T    2HCE,140.120   , 3.,   2HPR,140.907 , 3.,    2HND,144.240  , 3., BLOK  39
     U    2HPM,145.000   , 3.,   2HSM,150.350 , 3.,    2HEU,151.960  , 3., BLOK  40
     V    2HGD,157.250   , 3.,   2HTB,158.924 , 3.,    2HDY,162.500  , 3., BLOK  41
     W    2HHO,164.930   , 3.,   2HER,167.260 , 3.,    2HTM,168.934  , 3., BLOK  42
     X    2HYB,173.040   , 3.,   2HLU,174.997 , 3.,    2HHF,178.490  , 4., BLOK  43
     Y    2HTA,180.948   , 5.,   2HW ,183.850 , 6.,    2HRE,186.200  , 7., BLOK  44
     Z    2HOS,190.200   , 4.,   2HIR,192.200 , 4.,    2HPT,195.090  , 4., BLOK  45
     A    2HAU,196.967   , 3.,   2HHG,200.590 , 2.,    2HTL,204.370  , 1., BLOK  46
     B    2HPB,207.190   , 2.,   2HBI,208.980 , 3.,    2HPO,210.000  , 2., BLOK  47
     C    2HAT,210.000   , 0.,   2HRN,222.000 , 0.,    2HFR,223.000  , 1., BLOK  48
     D    2HRA,226.000   , 2.,   2HAC,227.000 , 3.,    2HTH,232.038  , 4., BLOK  49
     E    2HPA,231.000   , 5.,   2HU ,238.030 , 6.,    2HNP,237.000  , 5., BLOK  50
```

```
     F    2HPU,242.000   , 4.,    2HAM,243.000 , 3.,    2HCM,247.000   , 3., BLOK   51
     G    2HBK,249.000   , 3.,    2HCF,251.000 , 3.,    2HES,254.000   , 0., BLOK   52
     H    2HFM,253.000   , 0.,    2HD ,2.014102, 1./                         BLOK   53
C                                                                            BLOK   54
C      INFORMATION USED IN VARIABLE OUTPUT FORMAT                            BLOK   55
C
       DATA FMT/3H(1H,4H,3A4,4H,A2,,3HF9.,2H0,,3HF9.,2H0,,3HF9.,2H0,,3HF9BLOK0056
     1.,2H0,,3HF9.,2H0,,3HF9.,2H0,,3HF9.,2H0,,3HF9.,2H0,,3HF9.,2H0,,3HF9BLOK0057
     2.,2H0,,3HF9.,2H0,,3HF9.,2H0,,3HF9.,1H0,1H)/, FB,F0,F1,F2,F3,F4,F5/BLOK0058
     31H  ,2H0,,2H1,,2H2,,2H3,,2H4,,2H5,/,FMT13/2H13/,FMT9X/3H9X,/,FMTI9BLOK0059
     4/3HI9,/                                                            BLOK0060
      DATA                           FP/4HP, A,4HTM   ,2H  ,1H /         BLOK   61
     1,FT/4HT, D,4HEG K,4H     ,2H  /,FH/4HH, C,4HAL/G,2H  ,1H /         BLOK   62
     2,FS/4HS, C,4HAL/(,4HG)(K,2H) /,FM/4HM, M,4HOL W,2HT ,1H /          BLOK   63
     3,FV/4H(DLV,4H/DLP,4H)T   ,2H  /,FD/4H(DLV,4H/DLT,2H)P,1H /         BLOK   64
     4,FC/4HCP,  ,4HCAL/,4H(G)(,2HK)/,FG/4HGAMM,4HA (S,2H) ,1H /         BLOK   65
     5,FL/4HSON ,4HVEL,,4HM/SE,2HC /                                     BLOK   66
C                                                                        BLOK   67
C      INFORMATION USED IN PERFORMANCE OUTPUT                            BLOK   68
C                                                                        BLOK   69
      DATA FR1/4HPC/P/, FC1/2HCF/,     FN/4HMACH,4H NUM,4HBER ,1H /      BLOK   70
     1,FR/4HCSTA,4HR, F,4HT/SE,2HC  /,FI/4HISP,,4H LB-,4HSEC/,2HLB/      BLOK   71
     2,FA/4HIVAC,4H,LB-,4HSEC/,2HLB /,FA1/4HAE/A/,FA2/1HT/               BLOK   72
      END                                                                BLOK   73
```

APPENDIX F
SAMPLE CASES AND SAMPLE INPUTS

SAMPLE INPUTS:

```
    RESERVOIR PRESS. FIXED--1 ZONE--FROZEN--N2H4/N204--PRF=30KPSFA,PRO=30KPSFA
               LIQUID INJECTION--RC/RT=1.0--CORE O/F=1.5--NCEC71=0
     1
  &NUM   PRMT=0.,1.,.002,.001,0.,1., RT=.1, RCRT=1., EPSC=6., ESPG=.50,
    OXFULE=1.5,19*0., PRF=30000.,19*0., PRO=30000.,19*0., AITAC=.0005,
    AIPCT=1.,19*0., RHOIFU=1.9488,19*0., TIFU=537.,19*0.,
    RHOIOX=2.7787,19*0.,TIOX=537.,19*0.,   CDINJ=.7,19*0.,
    MTOTMX=0,MODECH=2,NAORW=1,MGORLI=2,NCHA1=1,NCHA2=0,NCEC71=0,  &END
  &EQU   NCHANG=1, TOG=5400.,19*0.,MWG=21.1,19*0.,GAMG=1.23,19*0., &END
        ZONE MASS FLOWS FIXED--2 ZONE--FROZ,FROZ--MDTI(1)=.1,MDTI(2)=.025
  GAS INJECTION--RC/RT=.5--STRIATED FLOW--INJ VELOCITIES FIXED--80/20 STRIATION
     2
  &NUM   PRMT=0.,1.,.004,.001,0.,1., RT=.1, RCRT=.5, MDTI=.1,.025,18*0.,
    OXFULE=6.,.5,18*0., RIFU=2*24665.,18*0., TIFU=2*537.,18*0.,
    WIFU=170.,600.,18*0., RIOX=2*1554.,18*0., TIOX=2*537.,18*0., EPSC=4.,
    WIOX=170.,600.,18*0., AGAMGU=1.2092,1.3792,18*0., ATOGU=6089.,1460.,18*0.,
    AMWGU=12.969,3.024,18*0., MODECH=1,MGORLI=1,NCHA1=1,NCHA2=1,
    MTOTMX=0, NAORW=2, NCEC71=0, &END
  &EQU   NCHANG=1, TOG=6089.,1460.,18*0., MWG=12.969,3.024,18*0.,
    GAMG=1.2092,1.3792,18*0.,    &END
 CHAMBER STATIC PRESS. FIXED--2 ZONE--EQULIBRIUM--H2/02 AT 50  PSIA
 GAS INJECTION--RC/RT=.5--CORE O/F=6., SHEATH O/F=.5--80/20 STRIATION
     2
  &NUM  PRMT=0.,1.,.008,.001,0.,1.,RT=.2,RCRT=.5,EPSC=4.,PG=7200.,
    TIFU=2*537.,18*0.,RIFU=2*24665.,18*0.,TIOX=2*537.,18*0.,
    RIOX=2*1554.,18*0.,WIFU=150.,600.,18*0.,WIOX=150.,600.,18*0.,
    PCTMO=.8,.2,18*0.,MTOTMX=0,MODECH=3,NAORW=2,MGORLI=1,NCHA1=1,NCHA2=1,
    NCEC71=2,OXFULE=6.,.5,18*0.,  &END
 REACTANTS
 O 2.                                   00       100.0   0.0      G298.15  O
 H 2.                                   00       100.    0.       G298.15  F

 NAMELISTS
 REACTANTS
 O 2.                                   00       100.0   0.0      G298.15  O
 H 2.                                   00       100.    0.       G298.15  F

 NAMELISTS
 CHAMBER STATIC PRESS. FIXED--2 ZONE--EQULIBRIUM--H2/02 AT 50  PSIA
 GAS INJECTION--RC/RT=.5--CORE O/F=6., SHEATH O/F=.5--60/40 STRIATION
    71
  &NUM   PCTMO=.6,.4,18*0.,    &END
```

PARAMETERS IN NAMELIST NUM VARIABLES

```
&NUM
PRVT= 0.0         ,  1.0000000     , 0.19999999E-02, 0.99999993E-03, 0.0        ,  1.0000000    ,ERWT= 0.0        ,
 0.0          , 0.0          , 0.0          ,PG= 0.0          ,MDTI= 0.0        , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          ,RT= 0.99999964E-01,RCRT=  1.0000000    ,EPSC=  6.0000000    ,OXFULE=  1.5000000    , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          ,PRF=  30000.000    , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,PRO=  30000.000    , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          ,ESPG= 0.50000000    ,AITAC= 0.49999985E-03,AIPCT=  1.0000000    , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
AIPCTF= 0.0         , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,RHOIFU=  1.9487991    , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          ,RIFU= 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,TIFU=
  537.00000   , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          ,WIFU= 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          ,RHOIOX=  2.7786999    , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,RIOX= 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          ,TIOX=  537.00000    , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
WIOX= 0.0         , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,PCTMO= 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          ,CDINJ= 0.69999999    , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,AGAMGU=
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          ,AMWGU= 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          ,ATOGU= 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,
 0.0          , 0.0          , 0.0          , 0.0          , 0.0          , 0.0          ,MTOTMX=          0,MODECH=
          2,NAORW=          1,MGORLI=          2,NCHA1=          1,NCHA2=          0,NSPPG=          0,NCEC71=          0
&END
```

CASE 1

RESERVOIR PRESS. FIXED—1 ZONE--FROZEN--N2H4/N2O4--PRF=30KPSFA,PRO=30KPSFA
LIQUID INJECTION--RC/RT=1.0--CORE O/F=1.5--NCEC71=0

```
LOOP= 6    RC/RT=   0.100000E 01    MODECH= 2    MGORLI= 2    NAORW= 1    NCEC71= 0
                                    NSPPG= 0     MTCTMX= 0    LETOUT= 1   NINCH= 1

RT=  0.100000E 00    RTCALC=  0.100002E 00    PG=  0.833603E 04    EPSC=  0.600000E 01    CLP=  0.546936E 04
```

INJECTOR PARAMETERS

ZONE PSI LIMITS =0.0   0.1000E 01

ZONE	PI	TIFU TIOX	AIFU AIOX	WIFU WIOX	RHOIFU RHOIOX	RIFU RIOX	MOTIFU MOTIOX	PRF PRO
1	0.841055E 04	0.537000E 03	0.942476E-04	0.104196E 03	0.194880E 01	0.0	0.191376E-01	0.300000E 05
1		0.537000E 03	0.118393E-03	0.872596E 02	0.277870E 01	0.0	0.287064E-01	0.300000E 05

BURNED GAS PARAMETERS

POG	TOG	CPG	GAMG	TG	AG	WG	RHOG	MOTI
0.838534E 04	0.540000E 04	0.126023E 05	0.123000E 01	0.539405E 04	0.188337E 00	0.387364E 03	0.655802E-03	0.478440E-01

THROAT PLANE

R	P	PSI	T	W	M	GAM
0.0	0.546936E 04	0.0	0.498530E 04	0.323300E 04	0.850496E 00	0.123000E 01
0.100000E-02	0.546934E 04	0.988361E-04	0.498530E 04	0.323302E 04	0.850500E 00	0.123000E 01
0.200000E-02	0.546925E 04	0.395343E-03	0.498528E 04	0.323309E 04	0.850520E 00	0.123000E 01
0.300000E-02	0.546906E 04	0.889528E-03	0.498525E 04	0.323321E 04	0.850554E 00	0.123000E 01
0.400000E-02	0.546876E 04	0.158139E-02	0.498520E 04	0.323341E 04	0.850613E 00	0.123000E 01
0.500000E-02	0.546832E 04	0.247095E-02	0.498512E 04	0.323370E 04	0.850695E 00	0.123000E 01
0.599999E-02	0.546773E 04	0.355822E-02	0.498502E 04	0.323408E 04	0.850803E 00	0.123000E 01
0.799999E-02	0.546609E 04	0.632598E-02	0.498474E 04	0.323519E 04	0.851119E 00	0.123000E 01
0.999999E-02	0.546374E 04	0.988494E-02	0.498434E 04	0.323675E 04	0.851563E 00	0.123000E 01
0.120000E-01	0.546064E 04	0.142354E-01	0.498381E 04	0.323880E 04	0.852147E 00	0.123000E 01
0.140000E-01	0.545675E 04	0.193779E-01	0.498314E 04	0.324139E 04	0.852886E 00	0.123000E 01
0.160000E-01	0.545200E 04	0.253130E-01	0.498234E 04	0.324454E 04	0.853785E 00	0.123000E 01
0.180000E-01	0.544637E 04	0.320414E-01	0.498138E 04	0.324827E 04	0.854848E 00	0.123000E 01
0.200000E-01	0.543982E 04	0.395639E-01	0.498025E 04	0.325262E 04	0.856089E 00	0.123000E 01
0.220000E-01	0.543231E 04	0.478814E-01	0.497896E 04	0.325760E 04	0.857511E 00	0.123000E 01
0.240000E-01	0.542381E 04	0.569950E-01	0.497751E 04	0.326323E 04	0.859120E 00	0.123000E 01
0.260000E-01	0.541429E 04	0.669058E-01	0.497587E 04	0.326955E 04	0.860923E 00	0.123000E 01
0.280000E-01	0.540369E 04	0.776150E-01	0.497405E 04	0.327656E 04	0.862929E 00	0.123000E 01
0.300000E-01	0.539200E 04	0.891241E-01	0.497204E 04	0.328430E 04	0.865142E 00	0.123000E 01
0.319999E-01	0.537916E 04	0.101435E 00	0.496982E 04	0.329280E 04	0.867573E 00	0.123000E 01
0.339999E-01	0.536514E 04	0.114548E 00	0.496739E 04	0.330206E 04	0.870226E 00	0.123000E 01
0.359999E-01	0.534989E 04	0.128466E 00	0.496475E 04	0.331213E 04	0.873114E 00	0.123000E 01
0.379999E-01	0.533336E 04	0.143191E 00	0.496188E 04	0.332304E 04	0.876242E 00	0.123000E 01
0.399999E-01	0.531550E 04	0.158723E 00	0.495877E 04	0.333482E 04	0.879624E 00	0.123000E 01
0.419999E-01	0.529626E 04	0.175066E 00	0.495541E 04	0.334749E 04	0.883265E 00	0.123000E 01
0.439999E-01	0.527557E 04	0.192221E 00	0.495178E 04	0.336112E 04	0.887185E 00	0.123000E 01
0.459999E-01	0.525338E 04	0.210189E 00	0.494788E 04	0.337572E 04	0.891391E 00	0.123000E 01
0.479999E-01	0.522961E 04	0.228974E 00	0.494368E 04	0.339134E 04	0.895896E 00	0.123000E 01
0.499999E-01	0.520418E 04	0.248576E 00	0.493918E 04	0.340803E 04	0.900715E 00	0.123000E 01
0.519999E-01	0.517702E 04	0.268998E 00	0.493435E 04	0.342584E 04	0.905866E 00	0.123000E 01
0.539999E-01	0.514804E 04	0.290241E 00	0.492917E 04	0.344485E 04	0.911369E 00	0.123000E 01
0.559999E-01	0.511713E 04	0.312307E 00	0.492363E 04	0.346508E 04	0.917238E 00	0.123000E 01
0.579999E-01	0.508420E 04	0.335197E 00	0.491769E 04	0.348661E 04	0.923495E 00	0.123000E 01
0.599999E-01	0.504913E 04	0.358913E 00	0.491133E 04	0.350952E 04	0.930166E 00	0.123000E 01
0.619999E-01	0.501180E 04	0.383455E 00	0.490452E 04	0.353390E 04	0.937275E 00	0.123000E 01
0.639999E-01	0.497208E 04	0.408823E 00	0.489722E 04	0.355980E 04	0.944849E 00	0.123000E 01
0.659999E-01	0.492982E 04	0.435018E 00	0.488941E 04	0.358735E 04	0.952920E 00	0.123000E 01
0.679998E-01	0.488485E 04	0.462038E 00	0.488104E 04	0.361663E 04	0.961523E 00	0.123000E 01
0.699998E-01	0.483701E 04	0.489882E 00	0.487207E 04	0.364777E 04	0.970695E 00	0.123000E 01
0.719998E-01	0.478611E 04	0.518548E 00	0.486244E 04	0.368088E 04	0.980475E 00	0.123000E 01
0.739998E-01	0.473193E 04	0.548032E 00	0.485210E 04	0.371612E 04	0.990915E 00	0.123000E 01
0.759997E-01	0.467426E 04	0.578328E 00	0.484099E 04	0.375362E 04	0.100206E 01	0.123000E 01
0.779997E-01	0.461284E 04	0.609432E 00	0.482903E 04	0.379355E 04	0.101398E 01	0.123000E 01
0.799997E-01	0.454741E 04	0.641333E 00	0.481614E 04	0.383611E 04	0.102672E 01	0.123000E 01
0.819997E-01	0.447765E 04	0.674021E 00	0.480225E 04	0.388151E 04	0.104037E 01	0.123000E 01
0.839996E-01	0.440326E 04	0.707483E 00	0.478722E 04	0.392998E 04	0.105502E 01	0.123000E 01
0.859996E-01	0.432387E 04	0.741702E 00	0.477096E 04	0.398178E 04	0.107074E 01	0.123000E 01
0.879996E-01	0.423909E 04	0.776656E 00	0.475333E 04	0.403721E 04	0.108766E 01	0.123000E 01
0.899996E-01	0.414850E 04	0.812321E 00	0.473417E 04	0.409657E 04	0.110589E 01	0.123000E 01
0.919995E-01	0.405163E 04	0.848665E 00	0.471330E 04	0.416028E 04	0.112557E 01	0.123000E 01
0.939995E-01	0.394798E 04	0.885650E 00	0.469051E 04	0.422874E 04	0.114687E 01	0.123000E 01
0.959995E-01	0.383699E 04	0.923230E 00	0.466557E 04	0.430243E 04	0.116997E 01	0.123000E 01
0.979995E-01	0.371807E 04	0.961350E 00	0.463818E 04	0.438192E 04	0.119509E 01	0.123000E 01
0.999994E-01	0.359057E 04	0.999942E 00	0.460802E 04	0.446784E 04	0.122251E 01	0.123000E 01
0.101999E 00	0.345382E 04	0.103892E 01	0.457468E 04	0.456090E 04	0.125251E 01	0.123000E 01
THROAT WALL						
0.100002E 00	0.359037E 04	0.100000E 01	0.460796E 04	0.446796E 04	0.122255E 01	0.123000E 01

PERFORMANCE

```
THROAT ISPVAC,LBF-SEC/LBM  CSTAR,FT/SEC
         0.212052E 03     0.550632E 04

THROAT ISPVID,LBF-SEC/LBM  CSTID,FT/SEC
         0.211126E 03     0.545206E 04

MO,LBF-SEC/FT          MOID,LBF-SEC/FT
  0.478440E-01           0.483202E-01

CD,DISCHARGE COEFFICIENT=   0.990145E 00

THROAT ISPVAC/ISPVID=   0.100438E 01

CSTAR/CSTID=   0.100995E 01
```

CASE 2

CHAMBER STATIC PRESS. FIXED--2 ZONE--EQULIBRIUM--H2/O2 AT 50 PSIA
GAS INJECTION--RC/RT=.5--CORE O/F=6., SHEATH O/F=.5--80/20 STRIATION

LOOP= 3	RC/RT= 0.500000E 00	MODECH= 3	MGORLI= 1	NAORW= 2	NCEC71= 2
		NSPPG= 0	MTOTMX= 0	LETOUT= 2	NINCH= 2

RT= 0.200000E 00 RTCALC= 0.200004E 00 PG= 0.720000E 04 EPSC= 0.400000E 01 CLP= 0.510585E 04

INJECTOR PARAMETERS

ZONE PSI LIMITS =0.0 0.8000E 00
ZONE PSI LIMITS =0.8000 0.1000E 01

ZONE	PI	TIFU / TIOX	AIFU / AIOX	WIFU / WIOX	RHOIFU / RHOIOX	RIFU / RIOX	MDTIFU / MDTIOX	PRF / PRO
1	0.732444E 04	0.537000E 03	0.164505E 00	0.150000E 03	0.552992E-03	0.246650E 05	0.136455E-01	0.293374E 05
1		0.537000E 03	0.621870E-01	0.150000E 03	0.877706E-02	0.155400E 04	0.818728E-01	0.303067E 05
2	0.732444E 04	0.537000E 03	0.479806E-01	0.600000E 03	0.552992E-03	0.246650E 05	0.159197E-01	0.300841E 05
2		0.537000E 03	0.151149E-02	0.600000E 03	0.877706E-02	0.155400E 04	0.795986E-02	0.421579E 05

BURNED GAS PARAMETERS.

POG	TOG	CPG	GAMG	TG	AG	WG	RHOG	MDTI
0.727946E 04	0.576760E 04	0.361197E 05	0.112110E 01	0.576076E 04	0.424293E 00	0.702770E 03	0.320337E-03	0.955183E-01
0.735471E 04	0.146013E 04	0.598062E 05	0.137920E 01	0.145162E 04	0.784664E-01	0.100890E 04	0.301643E-03	0.238796E-01

THROAT PLANE

R	P	PSI	T	W	M	GAM
0.0	0.510585E 04	0.0	0.561116E 04	0.388867E 04	0.789642E 00	0.111959E 01
0.400000E-02	0.510568E 04	0.385705E-03	0.561115E 04	0.388885E 04	0.789680E 00	0.111959E 01
0.799999E-02	0.510500E 04	0.154287E-02	0.561109E 04	0.388960E 04	0.789838E 00	0.111958E 01
0.120000E-01	0.510360E 04	0.347169E-02	0.561097E 04	0.389114E 04	0.790162E 00	0.111958E 01
0.160000E-01	0.510140E 04	0.617256E-02	0.561078E 04	0.389357E 04	0.790673E 00	0.111958E 01
0.200000E-01	0.509832E 04	0.964611E-02	0.561051E 04	0.389696E 04	0.791386E 00	0.111958E 01
0.240000E-01	0.509432E 04	0.138932E-01	0.561017E 04	0.390136E 04	0.792312E 00	0.111958E 01
0.320000E-01	0.508345E 04	0.247125E-01	0.560923E 04	0.391334E 04	0.794833E 00	0.111957E 01
0.400000E-01	0.506850E 04	0.386420E-01	0.560794E 04	0.392981E 04	0.798297E 00	0.111956E 01
0.480000E-01	0.504925E 04	0.556973E-01	0.560628E 04	0.395102E 04	0.802758E 00	0.111954E 01
0.560000E-01	0.502543E 04	0.758979E-01	0.560422E 04	0.397725E 04	0.808277E 00	0.111952E 01
0.640000E-01	0.499671E 04	0.992673E-01	0.560175E 04	0.400888E 04	0.814930E 00	0.111950E 01
0.719999E-01	0.496271E 04	0.125833E 00	0.559881E 04	0.404634E 04	0.822810E 00	0.111947E 01
0.799999E-01	0.492291E 04	0.155625E 00	0.559538E 04	0.409018E 04	0.832033E 00	0.111944E 01
0.879998E-01	0.487670E 04	0.188679E 00	0.559139E 04	0.414108E 04	0.842741E 00	0.111941E 01
0.959998E-01	0.482332E 04	0.225033E 00	0.558679E 04	0.419988E 04	0.855110E 00	0.111937E 01
0.104000E 00	0.476186E 04	0.264716E 00	0.558105E 04	0.426557E 04	0.869117E 00	0.111932E 01
0.112000E 00	0.469123E 04	0.307741E 00	0.557417E 04	0.433964E 04	0.885048E 00	0.111926E 01
0.120000E 00	0.461009E 04	0.354127E 00	0.556626E 04	0.442477E 04	0.903354E 00	0.111919E 01
0.128000E 00	0.451668E 04	0.403897E 00	0.555715E 04	0.452274E 04	0.924424E 00	0.111912E 01
0.136000E 00	0.440886E 04	0.457057E 00	0.554664E 04	0.463584E 04	0.948747E 00	0.111903E 01
0.144000E 00	0.428390E 04	0.513594E 00	0.553446E 04	0.476692E 04	0.976935E 00	0.111893E 01
0.151999E 00	0.413839E 04	0.573451E 00	0.551920E 04	0.491905E 04	0.101007E 01	0.111881E 01
0.159999E 00	0.396808E 04	0.636495E 00	0.550069E 04	0.509684E 04	0.104905E 01	0.111867E 01
0.167999E 00	0.376746E 04	0.702514E 00	0.547866E 04	0.530658E 04	0.109511E 01	0.111850E 01
0.175999E 00	0.352957E 04	0.771150E 00	0.545018E 04	0.555840E 04	0.115126E 01	0.111830E 01
0.177999E 00	0.346329E 04	0.788654E 00	0.544225E 04	0.562857E 04	0.116690E 01	0.111824E 01
0.179999E 00	0.339231E 04	0.806683E 00	0.118565E 04	0.571049E 04	0.109924E 01	0.138556E 01
0.180999E 00	0.335338E 04	0.816284E 00	0.118177E 04	0.574985E 04	0.110868E 01	0.138563E 01
0.181499E 00	0.333385E 04	0.821031E 00	0.117982E 04	0.576961E 04	0.111342E 01	0.138566E 01
0.181999E 00	0.331411E 04	0.825777E 00	0.117785E 04	0.578956E 04	0.111821E 01	0.138570E 01
0.182499E 00	0.329417E 04	0.830523E 00	0.117587E 04	0.580972E 04	0.112305E 01	0.138573E 01
0.182999E 00	0.327396E 04	0.835285E 00	0.117385E 04	0.583016E 04	0.112796E 01	0.138577E 01
0.183999E 00	0.323285E 04	0.844825E 00	0.116975E 04	0.587172E 04	0.113793E 01	0.138584E 01
0.184999E 00	0.319068E 04	0.854407E 00	0.116555E 04	0.591436E 04	0.114816E 01	0.138592E 01
0.185999E 00	0.314746E 04	0.864020E 00	0.116124E 04	0.595804E 04	0.115865E 01	0.138599E 01
0.186999E 00	0.310318E 04	0.873661E 00	0.115665E 04	0.600326E 04	0.116975E 01	0.138607E 01
0.187999E 00	0.305780E 04	0.883325E 00	0.115183E 04	0.604989E 04	0.118136E 01	0.138615E 01
0.188999E 00	0.301129E 04	0.893012E 00	0.114688E 04	0.609766E 04	0.119325E 01	0.138623E 01
0.190999E 00	0.291476E 04	0.912438E 00	0.113663E 04	0.619684E 04	0.121794E 01	0.138641E 01
0.192999E 00	0.281336E 04	0.931912E 00	0.112542E 04	0.630234E 04	0.124479E 01	0.138659E 01
0.194999E 00	0.270681E 04	0.951404E 00	0.111342E 04	0.641393E 04	0.127351E 01	0.138678E 01
0.196999E 00	0.259488E 04	0.970879E 00	0.110036E 04	0.653275E 04	0.130472E 01	0.138698E 01
0.200999E 00	0.235391E 04	0.100960E 01	0.107090E 04	0.679351E 04	0.137506E 01	0.138743E 01
THROAT WALL						
0.200004E 00	0.241596E 04	0.100000E 01	0.107864E 04	0.672579E 04	0.135659E 01	0.138731E 01

PERFORMANCE

```
THROAT ISPVAC,FT/SEC  CSTAR,FT/SEC
       0.927952E 04   0.767759E 04

THROAT ISPVID,FT/SEC  CSTID,FT/SEC
       0.920196E 04   0.753465E 04

MO,LBF-SEC/FT         MOID,LBF-SEC/FT
  0.119398E 00          0.121663E 00

CD,DISCHARGE COEFFICIENT=   0.981381E 00

THROAT ISPVAC/ISPVID=   0.100843E 01

CSTAR/CSTID=   0.101897E 01
```

CASE 3

CHAMBER STATIC PRESS. FIXED--2 ZONE--EQULIBRIUM--H2/O2 AT 50 PSIA
GAS INJECTION--RC/RT=.5--CORE O/F=6., SHEATH O/F=.5--60/40 STRIATION

LOOP= 3   RC/RT= 0.500000E 00   MODECH= 3   MGORLI= 1   NAORW= 2   NCECT1= 2
NSPPG= 0   MTOTMX= 0   LETOUT=71   NINCH= 2

RT= 0.200000E 00   RTCALC= 0.200074E 00   PG= 0.720000E 04   EPSC= 0.400000E 01   CLP= 0.498080E 04

INJECTOR PARAMETERS

ZONE PSI LIMITS =0.0 0.6000E 00
ZONE PSI LIMITS =0.6000 0.1000E 01

ZONE	PI	TIFU TIOX	AIFU AIOX	WIFU WIOX	RHOIFU RHOIOX	RIFU RIOX	MDTIFU MDTIOX	PRF PRO
1	0.731038E 04	0.537000E 03	0.124879E 00	0.150000E 03	0.551930E-03	0.246650E 05	0.103387E-01	0.293374E 05
1		0.537000E 03	0.472075E-01	0.150000E 03	0.876020E-02	0.155400E 04	0.620321E-01	0.303067E 05
2	0.731038E 04	0.537000E 03	0.971282E-01	0.600000E 03	0.551930E-03	0.246650E 05	0.321648E-01	0.300841E 05
2		0.537000E 03	0.305975E-02	0.600000E 03	0.876020E-02	0.155400E 04	0.160824E-01	0.421579E 05

BURNED GAS PARAMETERS

POG	TOG	CPG	GAMG	TG	AG	WG	RHOG	MDTI
0.727150E 04	0.576760E 04	0.361197E 05	0.112110E 01	0.576144E 04	0.338855E 00	0.666798E 03	0.320299E-03	0.723708E-01
0.734444E 04	0.146013E 04	0.598062E 05	0.137920E 01	0.145218E 04	0.164059E 00	0.975317E 03	0.301527E-03	0.482472E-01

THROAT PLANE

R	P	PSI	T	W	M	GAM
0.0	0.498080E 04	0.0	0.560038E 04	0.402641E 04	0.818619E 00	0.111949E 01
0.400000E-02	0.498063E 04	0.386688E-03	0.560036E 04	0.402660E 04	0.818658E 00	0.111949E 01
0.799999E-02	0.497991E 04	0.154630E-02	0.560030E 04	0.402739E 04	0.818825E 00	0.111949E 01
0.120000E-01	0.497844E 04	0.348051E-02	0.560017E 04	0.402901E 04	0.819164E 00	0.111949E 01
0.160000E-01	0.497613E 04	0.618818E-02	0.559997E 04	0.403155E 04	0.819700E 00	0.111948E 01
0.200000E-01	0.497290E 04	0.967040E-02	0.559969E 04	0.403511E 04	0.820448E 00	0.111948E 01
0.240000E-01	0.496871E 04	0.139279E-01	0.559933E 04	0.403972E 04	0.821418E 00	0.111948E 01
0.320000E-01	0.495731E 04	0.247731E-01	0.559835E 04	0.405228E 04	0.824061E 00	0.111947E 01
0.400000E-01	0.494163E 04	0.387344E-01	0.559700E 04	0.406955E 04	0.827694E 00	0.111946E 01
0.480000E-01	0.492145E 04	0.558259E-01	0.559525E 04	0.409178E 04	0.832370E 00	0.111944E 01
0.560000E-01	0.489649E 04	0.760652E-01	0.559310E 04	0.411928E 04	0.838155E 00	0.111942E 01
0.640000E-01	0.486639E 04	0.994735E-01	0.559050E 04	0.415243E 04	0.845129E 00	0.111940E 01
0.719999E-01	0.483075E 04	0.126075E 00	0.558743E 04	0.419168E 04	0.853386E 00	0.111937E 01
0.799999E-01	0.478906E 04	0.155895E 00	0.558371E 04	0.423703E 04	0.862980E 00	0.111934E 01
0.879998E-01	0.474071E 04	0.188952E 00	0.557900E 04	0.428775E 04	0.873887E 00	0.111930E 01
0.959998E-01	0.468496E 04	0.225256E 00	0.557356E 04	0.434622E 04	0.886462E 00	0.111926E 01
0.104000E 00	0.462088E 04	0.264838E 00	0.556731E 04	0.441344E 04	0.900918E 00	0.111920E 01
0.112000E 00	0.454732E 04	0.307713E 00	0.556014E 04	0.449060E 04	0.917512E 00	0.111914E 01
0.120000E 00	0.446286E 04	0.353897E 00	0.555191E 04	0.457920E 04	0.936566E 00	0.111907E 01
0.128000E 00	0.436573E 04	0.403395E 00	0.554244E 04	0.468108E 04	0.958476E 00	0.111900E 01
0.136000E 00	0.425375E 04	0.456195E 00	0.553152E 04	0.479854E 04	0.983736E 00	0.111891E 01
0.144000E 00	0.412420E 04	0.512250E 00	0.551765E 04	0.493387E 04	0.101332E 01	0.111880E 01
0.151999E 00	0.397374E 04	0.571453E 00	0.550131E 04	0.509093E 04	0.104775E 01	0.111867E 01
0.155999E 00	0.398771E 04	0.602705E 00	0.123126E 04	0.521445E 04	0.985459E 00	0.138470E 01
0.157999E 00	0.383998E 04	0.619297E 00	0.122715E 04	0.526230E 04	0.996023E 00	0.138478E 01
0.158999E 00	0.381549E 04	0.627588E 00	0.122505E 04	0.528671E 04	0.100141E 01	0.138483E 01
0.159999E 00	0.379062E 04	0.635921E 00	0.122291E 04	0.531159E 04	0.100690E 01	0.138487E 01
0.160999E 00	0.376525E 04	0.644296E 00	0.122056E 04	0.533698E 04	0.101274E 01	0.138491E 01
0.161999E 00	0.373932E 04	0.652729E 00	0.121815E 04	0.536292E 04	0.101873E 01	0.138496E 01
0.163999E 00	0.368585E 04	0.669734E 00	0.121318E 04	0.541642E 04	0.103107E 01	0.138505E 01
0.165999E 00	0.362999E 04	0.686949E 00	0.120799E 04	0.547231E 04	0.104396E 01	0.138515E 01
0.167999E 00	0.357168E 04	0.704358E 00	0.120257E 04	0.553066E 04	0.105741E 01	0.138525E 01
0.169999E 00	0.351076E 04	0.721955E 00	0.119690E 04	0.559160E 04	0.107147E 01	0.138536E 01
0.171999E 00	0.344712E 04	0.739732E 00	0.119099E 04	0.565528E 04	0.108616E 01	0.138547E 01
0.173999E 00	0.338059E 04	0.757680E 00	0.118448E 04	0.572235E 04	0.110208E 01	0.138558E 01
0.177999E 00	0.323921E 04	0.794047E 00	0.117022E 04	0.586630E 04	0.113663E 01	0.138583E 01
0.181999E 00	0.308221E 04	0.830960E 00	0.115443E 04	0.602475E 04	0.117510E 01	0.138611E 01
0.185999E 00	0.291123E 04	0.868298E 00	0.113625E 04	0.620047E 04	0.121884E 01	0.138641E 01
0.189999E 00	0.272347E 04	0.905894E 00	0.111530E 04	0.639649E 04	0.126902E 01	0.138675E 01
0.197999E 00	0.229126E 04	0.980938E 00	0.106306E 04	0.686199E 04	0.139378E 01	0.138755E 01
0.205999E 00	0.177499E 04	0.105357E 01	0.998479E 03	0.742632E 04	0.154803E 01	0.138852E 01
THROAT WALL						
0.200074E 00	0.216509E 04	0.100000E 01	0.104745E 04	0.699925E 04	0.143107E 01	0.138778E 01

PERFORMANCE

```
THROAT ISPVAC,FT/SEC  CSTAR,FT/SEC
       0.924010E 04   0.761168E 04

THROAT ISPVID,FT/SEC  CSTID,FT/SEC
       0.917499E 04   0.747535E 04

MO,LBF-SEC/FT         MOID,LBF-SEC/FT
  0.120618E 00          0.122818E 00

CD,DISCHARGE COEFFICIENT=   0.982089E 00

THROAT ISPVAC/ISPVID=   0.100710E 01

CSTAR/CSTID=   0.101824E 01
```

CASE 4

ZONE MASS FLOWS FIXED--2 ZONE--FROZ,FROZ--MDTI(1)=.1,MDTI(2)=.025
GAS INJECTION--RC/RT=.5--STRIATED FLOW--INJ VELOCITIES FIXED--80/20 STRIATION

ZONE	EST.GAMMA	EST.MW	EST.TO
1	0.120920E 01	0.129690E 02	0.608900E 04
2	0.137920E 01	0.302400E 01	0.146000E 04

LOOP= 2 RC/RT= 0.500000E 00 MODECH= 1 MGORLI= 1 NAORW= 2 NCEC71= 0
NSPPG= 0 MTOTMX= 0 LETOUT= 2 NINCH= 2

RT= 0.100000E 00 RTCALC= 0.999724E-01 PG= 0.296703E 05 EPSC= 0.400000E 01 CLP= 0.211065E 05

INJECTOR PARAMETERS

ZONE PSI LIMITS =0.0 0.8000E 00
ZONE PSI LIMITS =0.8000E 000.1000E 01

ZONE	PI	TIFU TIOX	AIFU AIOX	WIFU WIOX	RHOIFU RHOIOX	RIFU RIOX	MDTIFU MDTIOX	PRF PRO
1	0.302083E 05	0.537000E 03	0.368453E-01	0.170000E 03	0.228071E-02	0.246650E 05	0.142857E-01	0.302755E 05
1		0.537000E 03	0.139285E-01	0.170000E 03	0.361993E-01	0.155400E 04	0.857142E-01	0.312758E 05
2	0.302083E 05	0.537000E 03	0.121794E-01	0.600000E 03	0.228071E-02	0.246650E 05	0.166667E-01	0.310461E 05
2		0.537000E 03	0.383678E-03	0.600000E 03	0.361993E-01	0.155400E 04	0.833333E-02	0.435060E 05

BURNED GAS PARAMETERS

POG	TOG	CPG	GAMG	TG	AG	WG	RHOG	MDTI
0.300212E 05	0.608900E 04	0.221607E 05	0.120920E 01	0.607663E 04	0.106041E 00	0.740483E 03	0.127354E-02	0.100000E 00
0.303260E 05	0.146000E 04	0.598042E 05	0.137920E 01	0.145125E 04	0.196550E-01	0.102296E 04	0.124339E-02	0.250000E-01

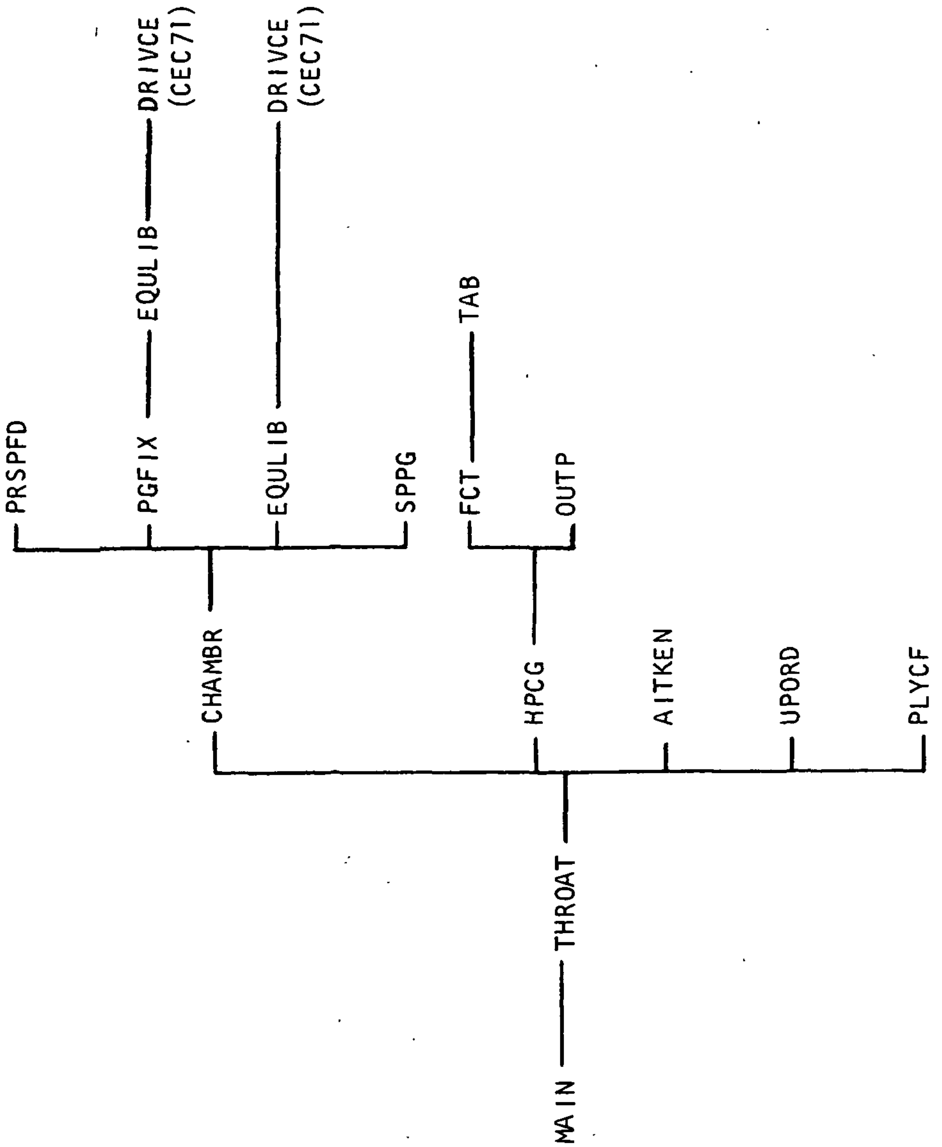
THROAT PLANE

R	P	PSI	T	W	M	GAM
0.0	0.211065E 05	0.0	0.572893E 04	0.399492E 04	0.775152E 00	0.120920E 01
0.200000E-02	0.211058E 05	0.385942E-03	0.572890E 04	0.399499E 04	0.775187E 00	0.120920E 01
0.400000E-02	0.211029E 05	0.154382E-02	0.572877E 04	0.399576E 04	0.775346E 00	0.120920E 01
0.599999E-02	0.210969E 05	0.347385E-02	0.572848E 04	0.399732E 04	0.775659E 00	0.120920E 01
0.799999E-02	0.210874E 05	0.617642E-02	0.572804E 04	0.399980E 04	0.776181E 00	0.120920E 01
0.999999E-02	0.210741E 05	0.965216E-02	0.572741E 04	0.400325E 04	0.776891E 00	0.120920E 01
0.120000E-01	0.210569E 05	0.139020E-01	0.572661E 04	0.400771E 04	0.777812E 00	0.120920E 01
0.160000E-01	0.210101E 05	0.247285E-01	0.572440E 04	0.401989E 04	0.780328E 00	0.120920E 01
0.200000E-01	0.209458E 05	0.386679E-01	0.572136E 04	0.403660E 04	0.783778E 00	0.120920E 01
0.240000E-01	0.208629E 05	0.557360E-01	0.571744E 04	0.405906E 04	0.788215E 00	0.120920E 01
0.280000E-01	0.207604E 05	0.759524E-01	0.571257E 04	0.408450E 04	0.793708E 00	0.120920E 01
0.320000E-01	0.206368E 05	0.993406E-01	0.570667E 04	0.411647E 04	0.800314E 00	0.120920E 01
0.360000E-01	0.204904E 05	0.125927E 00	0.569965E 04	0.415409E 04	0.808125E 00	0.120920E 01
0.400000E-01	0.203192E 05	0.155743E 00	0.569138E 04	0.419798E 04	0.817257E 00	0.120920E 01
0.440000E-01	0.201205E 05	0.188820E 00	0.568171E 04	0.424871E 04	0.827837E 00	0.120920E 01
0.480000E-01	0.198911E 05	0.225194E 00	0.567045E 04	0.430703E 04	0.840033E 00	0.120920E 01
0.520000E-01	0.196270E 05	0.264899E 00	0.565736E 04	0.437391E 04	0.854063E 00	0.120920E 01
0.559999E-01	0.193234E 05	0.307969E 00	0.564212E 04	0.445044E 04	0.870180E 00	0.120920E 01
0.599999E-01	0.189742E 05	0.354437E 00	0.562435E 04	0.453805E 04	0.888712E 00	0.120920E 01
0.639998E-01	0.185721E 05	0.404324E 00	0.560354E 04	0.463855E 04	0.910076E 00	0.120920E 01
0.679998E-01	0.181076E 05	0.457639E 00	0.557904E 04	0.475416E 04	0.934806E 00	0.120920E 01
0.719998E-01	0.175692E 05	0.514369E 00	0.554998E 04	0.488774E 04	0.963583E 00	0.120920E 01
0.759997E-01	0.169420E 05	0.574464E 00	0.551519E 04	0.504303E 04	0.997330E 00	0.120920E 01
0.799997E-01	0.162072E 05	0.637811E 00	0.547304E 04	0.522495E 04	0.103728E 01	0.120920E 01
0.839996E-01	0.153411E 05	0.704205E 00	0.542179E 04	0.544004E 04	0.108512E 01	0.120920E 01
0.879996E-01	0.143131E 05	0.773289E 00	0.535662E 04	0.569737E 04	0.114329E 01	0.120920E 01
0.889995E-01	0.140265E 05	0.790918E 00	0.533791E 04	0.576969E 04	0.115983E 01	0.120920E 01
0.899994E-01	0.137208E 05	0.809015E 00	0.117396E 04	0.584917E 04	0.113362E 01	0.137920E 01
0.904994E-01	0.135546E 05	0.818571E 00	0.117003E 04	0.588917E 04	0.114329E 01	0.137920E 01
0.907493E-01	0.134711E 05	0.823302E 00	0.116805E 04	0.590932E 04	0.114817E 01	0.137920E 01
0.909993E-01	0.133867E 05	0.828034E 00	0.116603E 04	0.592969E 04	0.115313E 01	0.137920E 01
0.912493E-01	0.133014E 05	0.832766E 00	0.116398E 04	0.595030E 04	0.115815E 01	0.137920E 01
0.914993E-01	0.132150E 05	0.837510E 00	0.116190E 04	0.597120E 04	0.116326E 01	0.137920E 01
0.919992E-01	0.130394E 05	0.847013E 00	0.115763E 04	0.601378E 04	0.117371E 01	0.137920E 01
0.924992E-01	0.128594E 05	0.856550E 00	0.115322E 04	0.605754E 04	0.118452E 01	0.137920E 01
0.929992E-01	0.126750E 05	0.866111E 00	0.114865E 04	0.610250E 04	0.119568E 01	0.137920E 01
0.934991E-01	0.124862E 05	0.875694E 00	0.114392E 04	0.614868E 04	0.120722E 01	0.137920E 01
0.944991E-01	0.120949E 05	0.894908E 00	0.113394E 04	0.624490E 04	0.123149E 01	0.137920E 01
0.954991E-01	0.116844E 05	0.914170E 00	0.112323E 04	0.634667E 04	0.125751E 01	0.137920E 01
0.964991E-01	0.112539E 05	0.933447E 00	0.111170E 04	0.645443E 04	0.128548E 01	0.137920E 01
0.974991E-01	0.108024E 05	0.952708E 00	0.109925E 04	0.656974E 04	0.131563E 01	0.137920E 01
0.994990E-01	0.983281E 04	0.991020E 00	0.107120E 04	0.681940E 04	0.138361E 01	0.137920E 01
0.101499E 00	0.876979E 04	0.102872E 01	0.103802E 04	0.710436E 04	0.146427E 01	0.137920E 01
THROAT WALL						
0.999724E-01	0.958939E 04	0.100000E 01	0.106379E 04	0.688381E 04	0.140158E 01	0.137920E 01

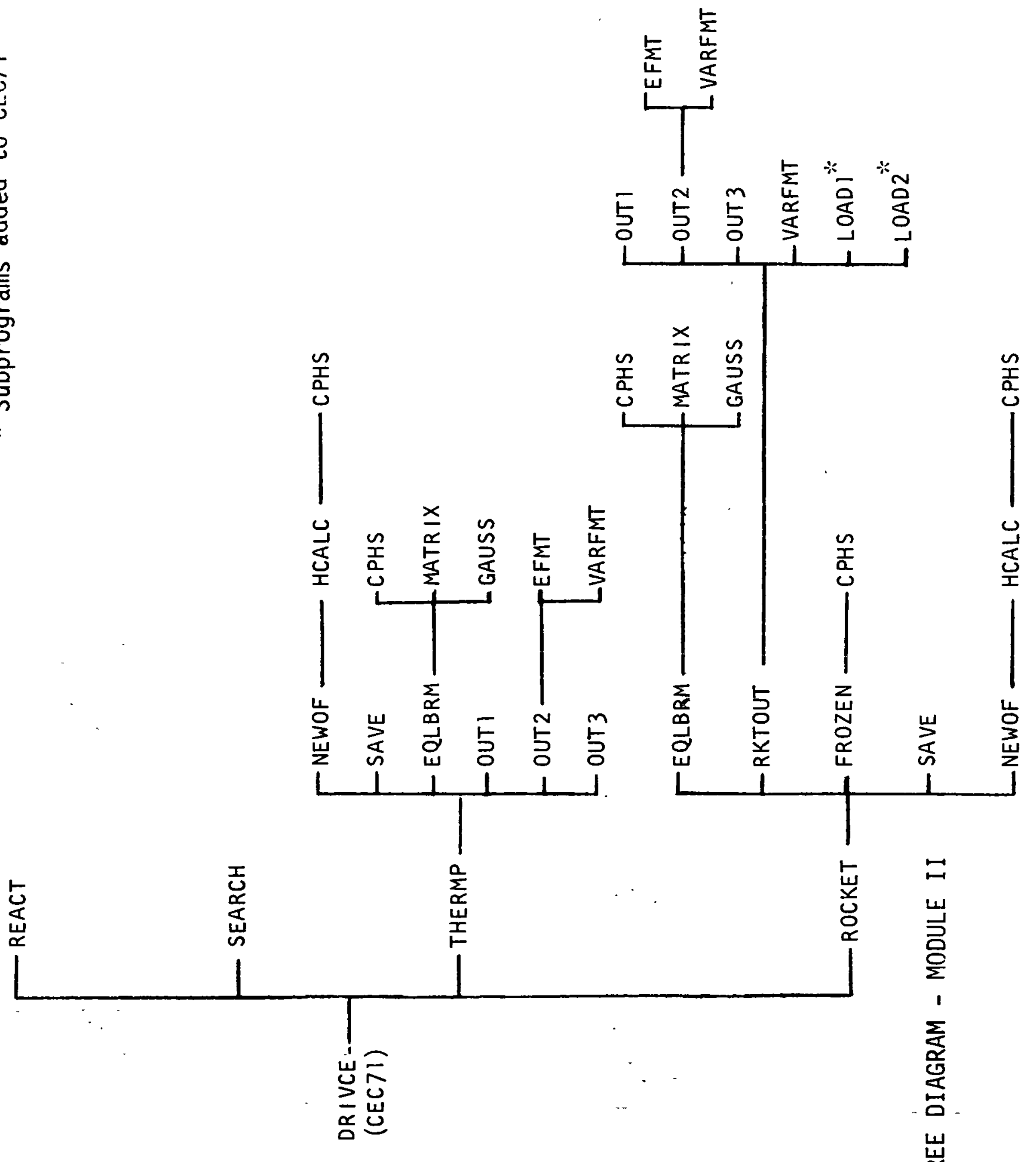
PERFORMANCE

```
THROAT ISPVAC,LBF-SEC/LBM  CSTAR,FT/SEC
          0.289113E 03     0.755629E 04

THROAT ISPVID,LBF-SEC/LBM  CSTID,FT/SEC
          0.286263E 03     0.738404E 04

MO,LBF-SEC/FT          MOID,LBF-SEC/FT
  0.125000E 00           0.127916E 00

CD,DISCHARGE COEFFICIENT=   0.977205E 00

THROAT ISPVAC/ISPVID=   0.100995E 01

CSTAR/CSTID=   0.102333E 01
```

APPENDIX G

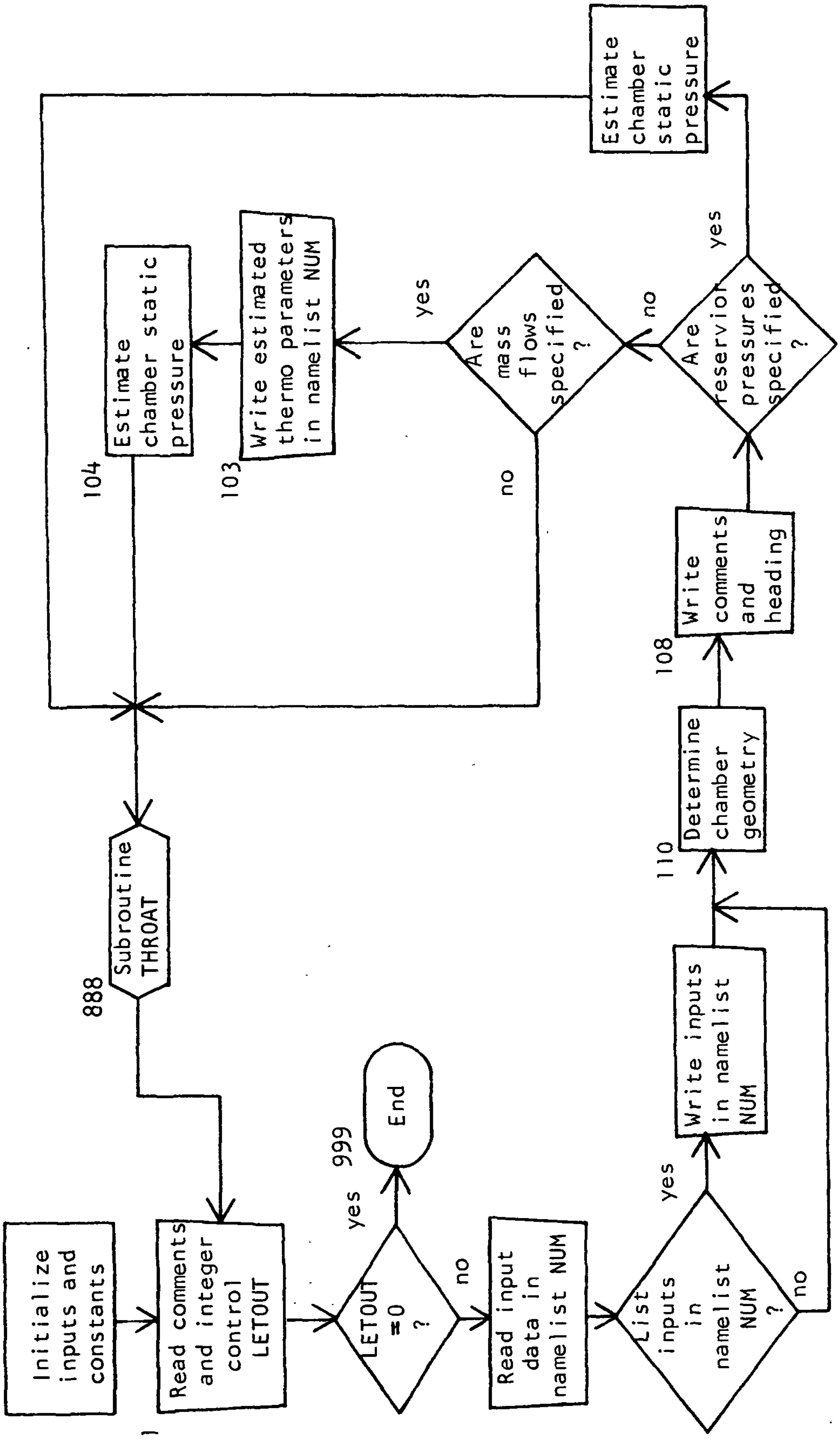


ROUTINE TREE DIAGRAMS - MODULE I

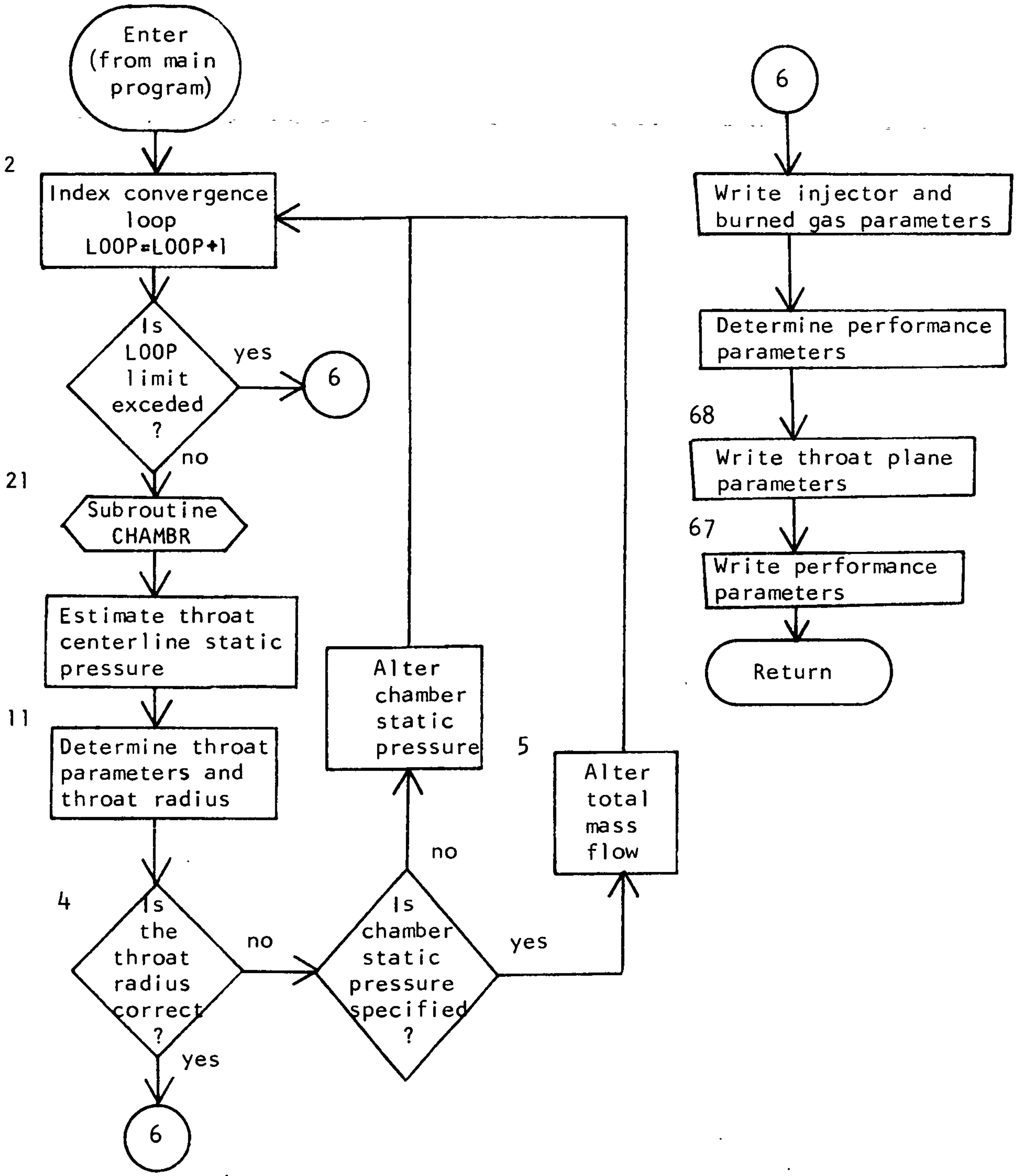


* Subprograms added to CEC71

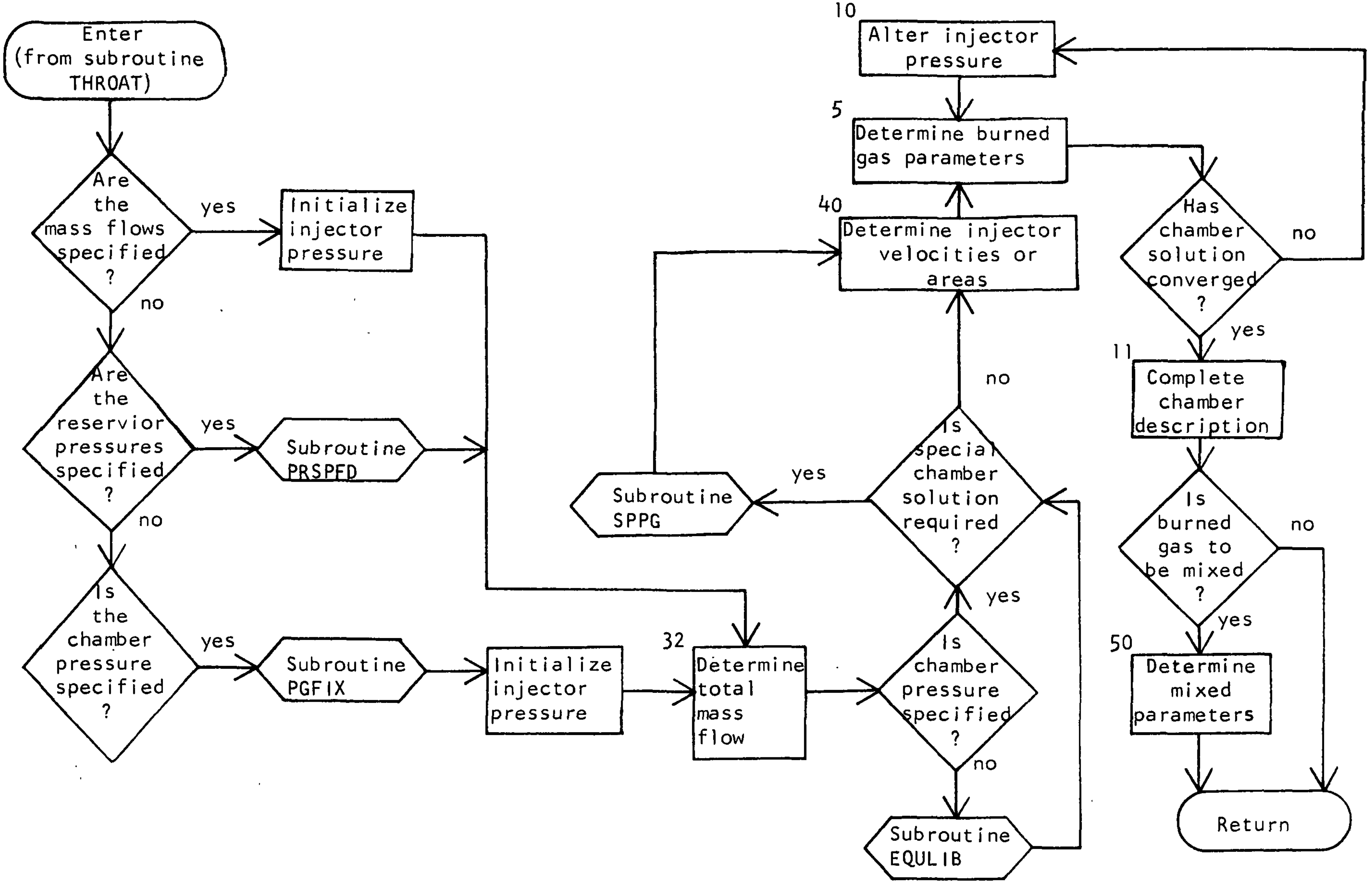
ROUTINE TREE DIAGRAM - MODULE II



Flow diagram for main routine



Flow diagram for subroutine THROAT



Flow diagram for subroutine CHAMBR

REFERENCES

1. Gordon, Sanford; and McBride, Bonnie J.: Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouget Detonations. NASA SP-273, 1971.

2. Back, L. H.; Cuffel, R. F.; and Massier, P. F.: Laminarization of a Turbulent Boundary Layer in Nozzle Flow. AIAA J., Vol. 7, No. 4, April 1969, pp. 730-733.

3. Back, L. H.; and Seban, P. A.: Flow and Heat Transfer in a Turbulent Boundary Layer with Large Acceleration Parameter. Proceedings of the Heat Transfer Fluid Mechanics Institute, 1967, pp. 410-426.

4. Norton, David J.: Subsonic, Transonic, and Supersonic Nozzle Flow by the Inverse Technique. Journal of Spacecraft and Rockets, Vol. 9, No. 6, June 1972, pp. 457-459.

5. Cuffel, R. F.; Back L. H.; and Massier, P. F.: Transonic Flowfield in a Supersonic Nozzle with Small Throat Radius of Curvature. AIAA J., Vol. 7, No. 7, July 1969, pp. 1364-1366.

6. Back, L. H.; and Cuffel, R. F.: Flow Coefficients for Supersonic Nozzles with Comparatively Small Radius of Curvature Throats. Journal of Spacecraft and Rockets, Vol. 8, No. 2, Feb. 1971, pp. 196-198.

7. Norton, D. J.; and Shelton, S.: Performance of Rocket Nozzles with Low Radius of Curvature Ratios. JPL Space Programs Summary No. 37-55, Vol. III, Reb. 1969, Pasadena, California, pp. 167-169.

8. Norton, David J.: Generalized One-Dimensional, Compound Compressible Nozzle Flow. Journal of Spacecraft and Rockets, Vol. 7, No. 3, March 1970, pp. 338-341.

9. Anon: IBM Application Program, H20-0205-3, System/360 Scientific Subroutine Package (360A-CM-03X) Version III, Programers Manual. IBM Corporation, 1969, pp. 337-341.

10. Nickerson, Gary R.: Striated Transonic Flow in a Converging-Diverging Nozzle. Dynamic Science Document No. SN-195, Feb. 1971.

11. Serra, Raymond A.: Determination of Internal Gas Flows by a Transient Numerical Technique. AIAA J., Vol. 10, No. 5, May 1972, pp. 603-611.

TABLE I - PROGRAM INPUT FOR

PRESELECTED THERMOCHEMICAL

INPUT DATA

Comment
Card

Comment
Card

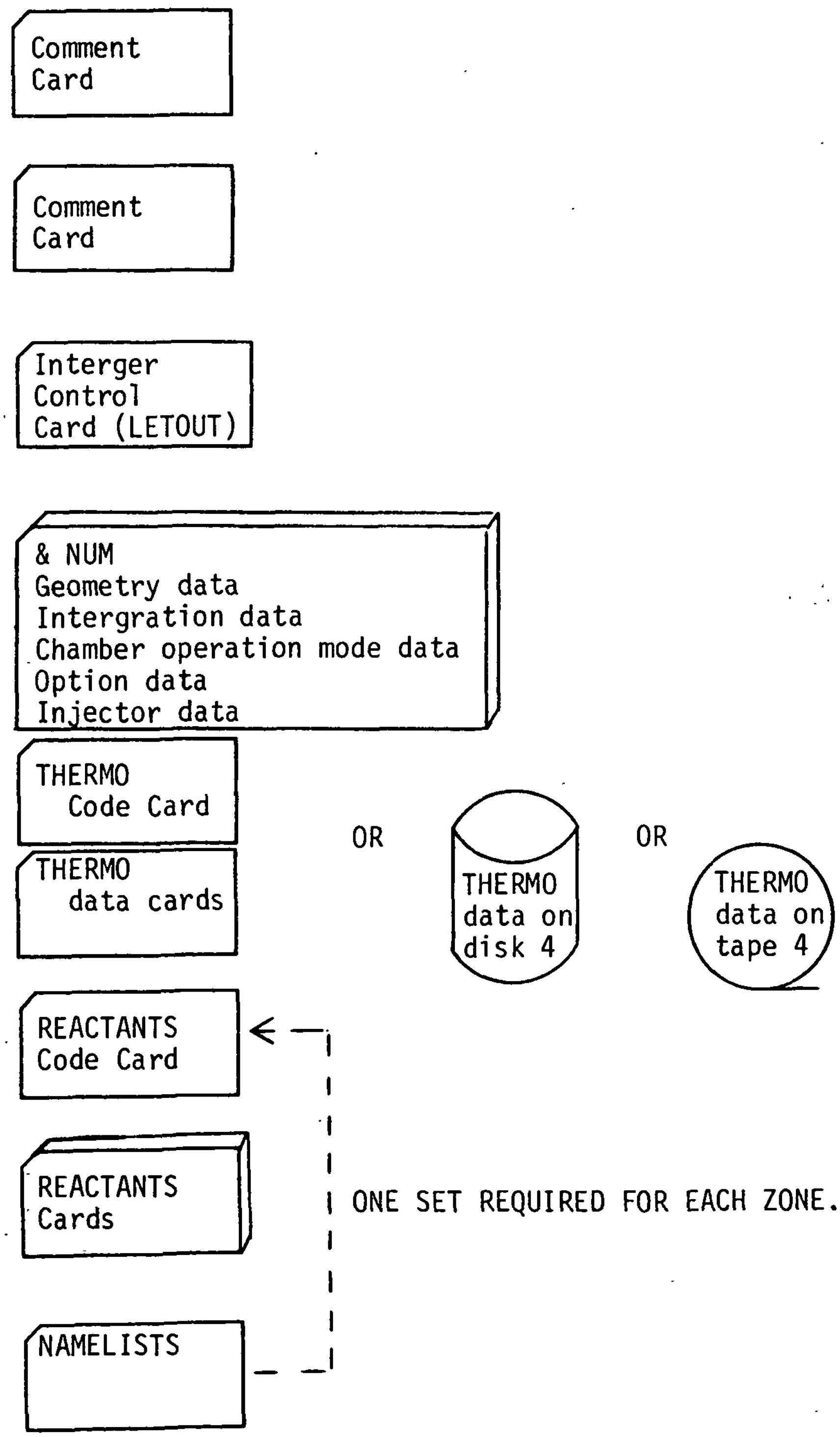
Integer
Control
Card (LETOUT)

& NUM
Geometry data
Integration data
Chamber operation mode data
  or mass generation data
Option data
Injector data
Initial estimations data
   for variable chamber pressure

& EQU
TOG
GAMG
MWG
NCHANG

TABLE II - PROGRAM INPUT FOR

INTERNALLY GENERATED THERMOCHEMICAL DATA



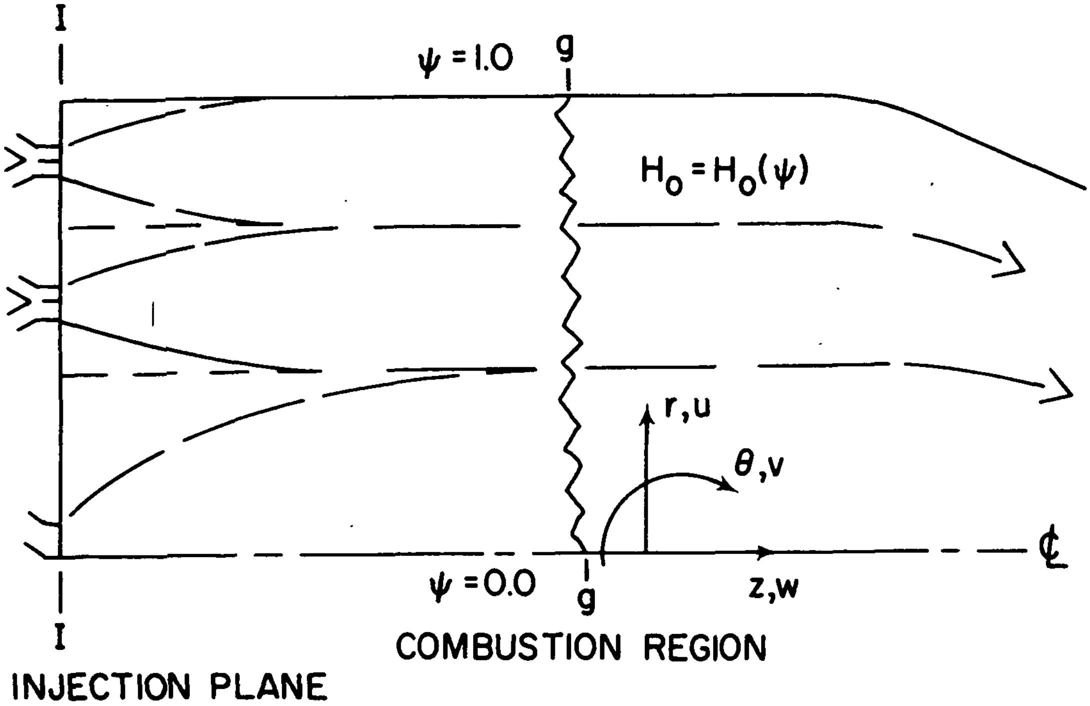


Figure 1a. Schematic of Nozzle Chamber.

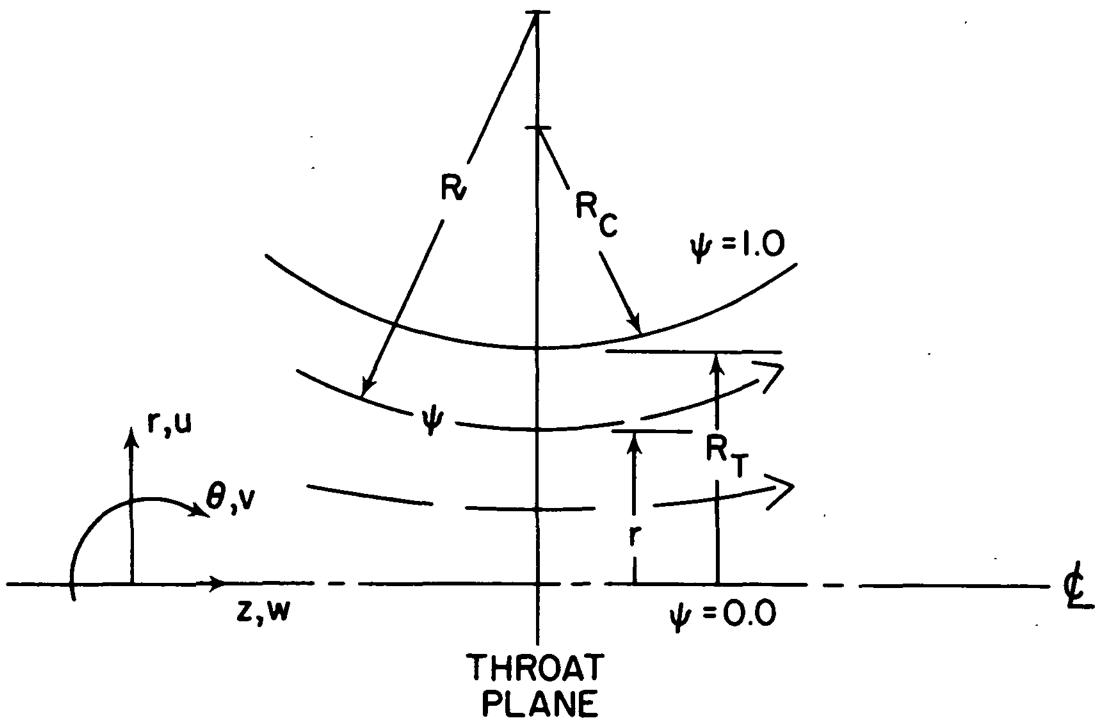


Figure 1b. Schematic of Nozzle Throat.

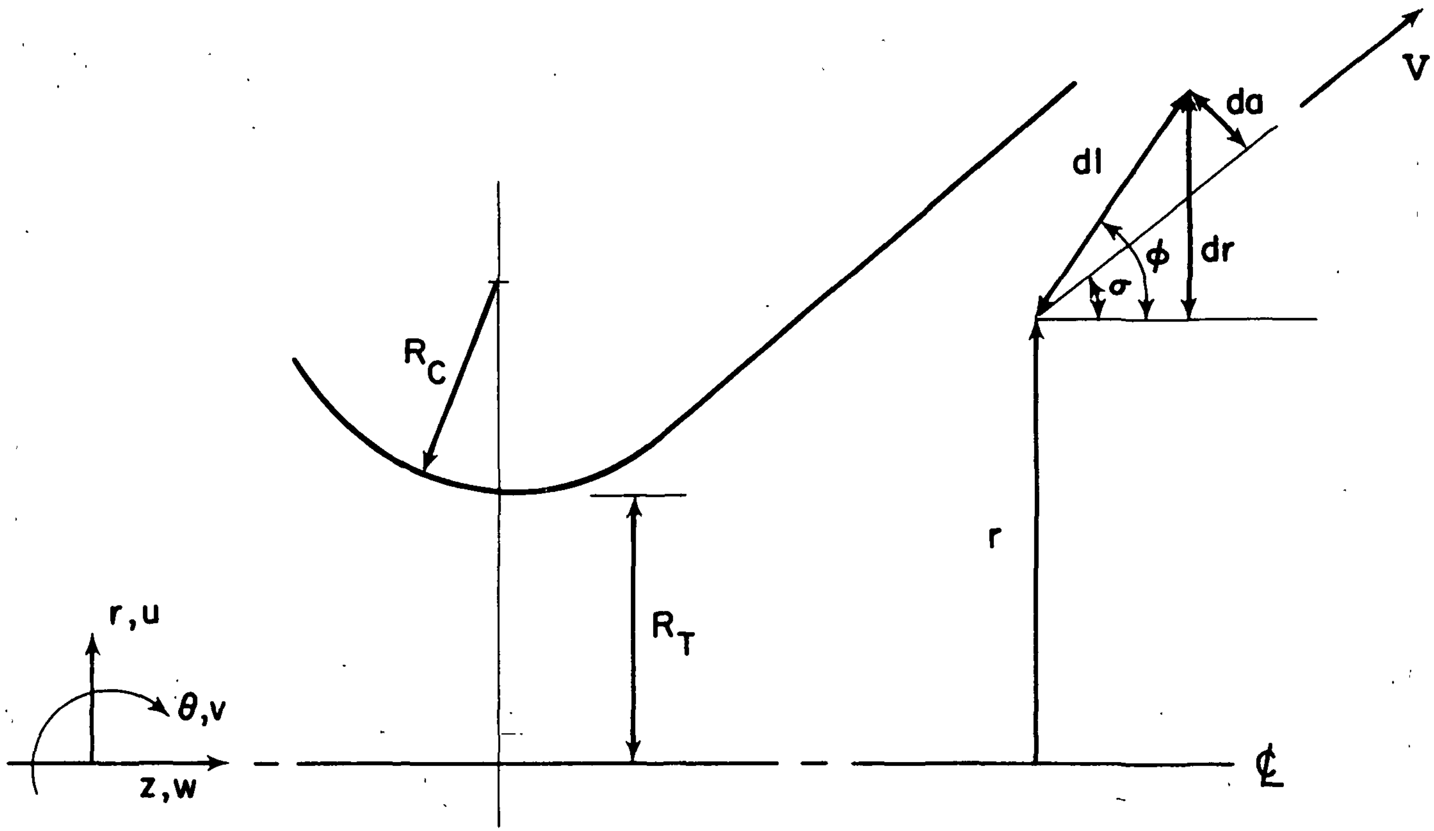


Figure 2. Schematic of Axisymmetric Nozzle for Derivation of Mass Flow.

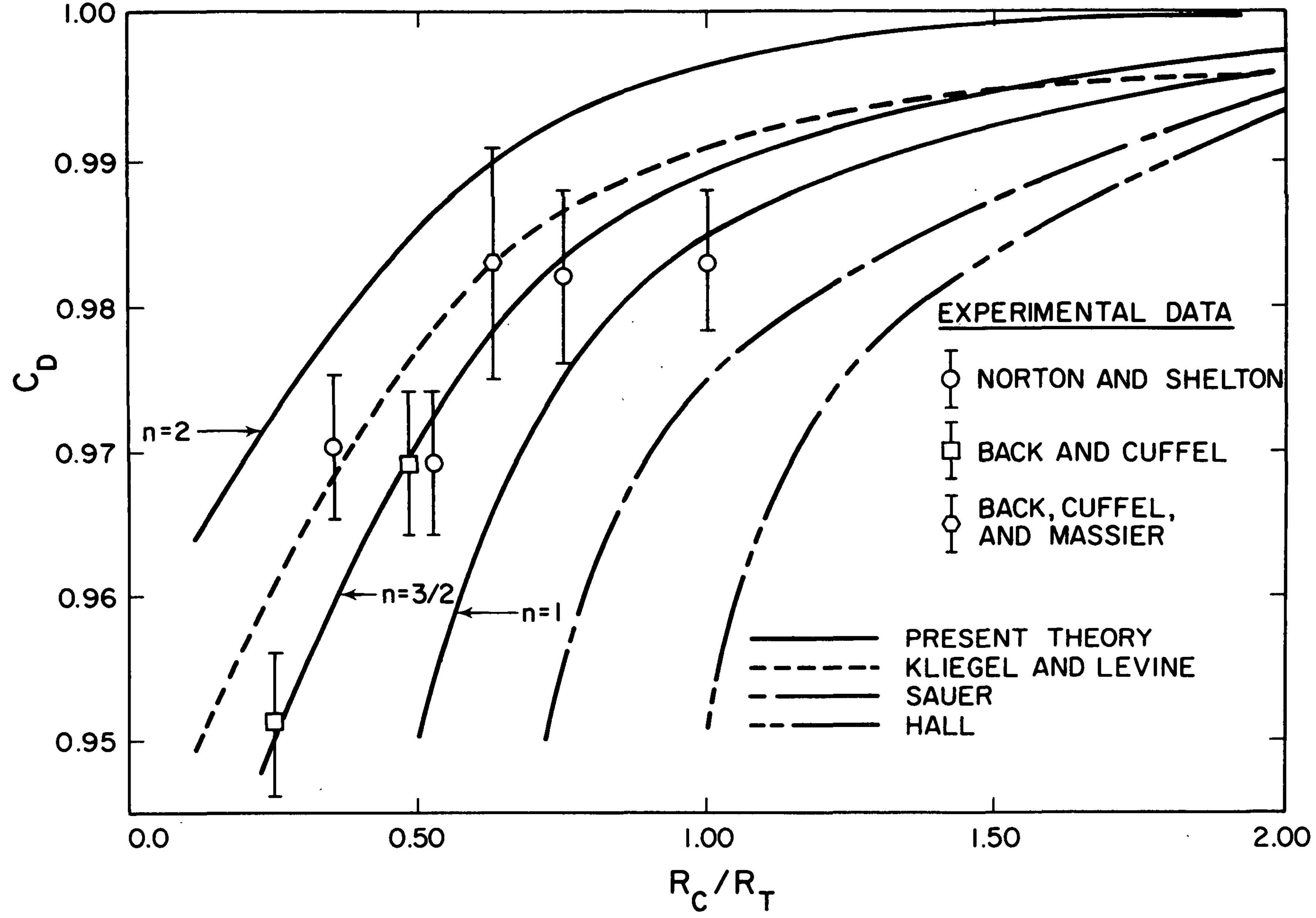


Figure 3. Discharge Coefficient Comparison Between Experimental and Analytical Results.

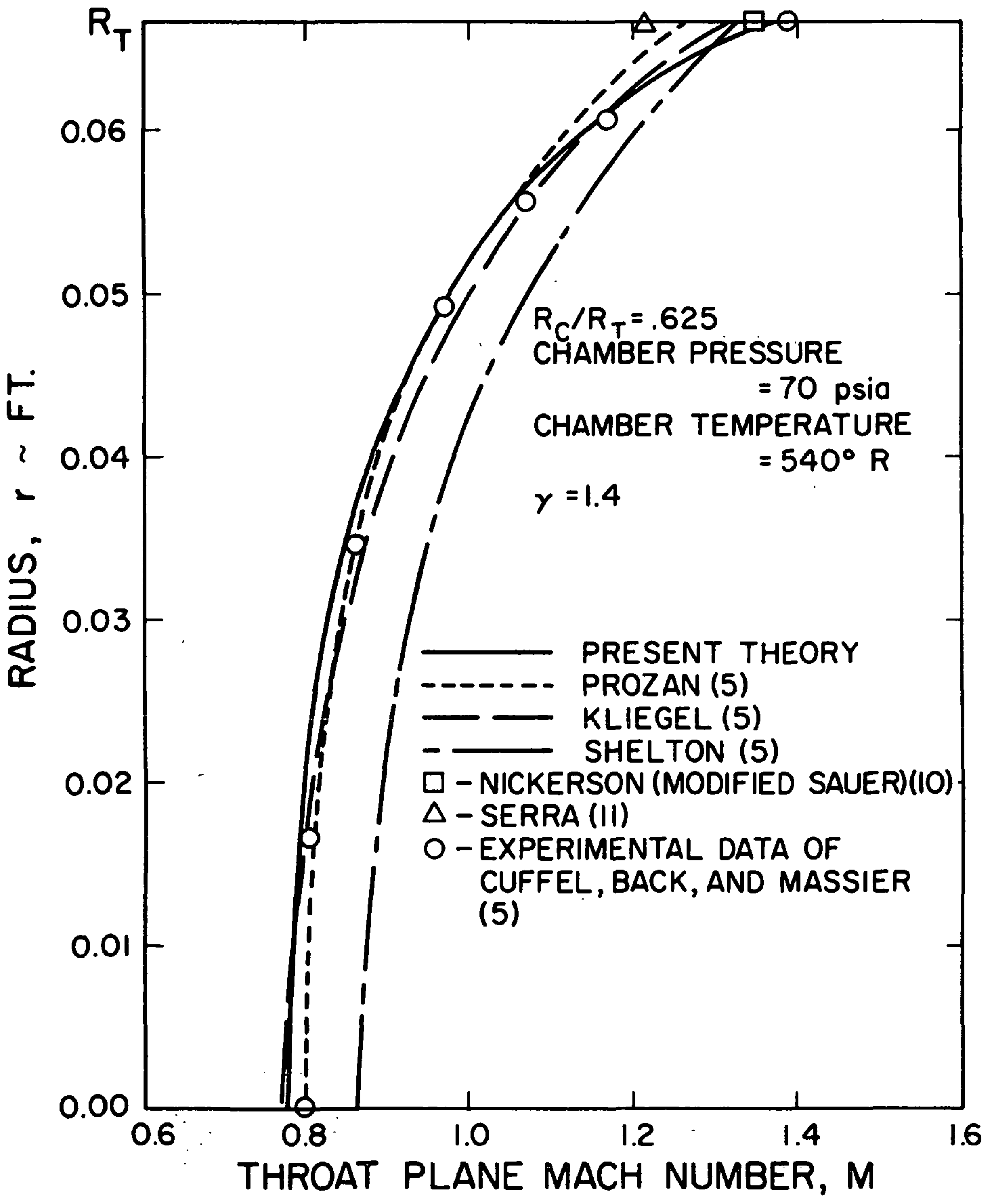


Figure 4. Throat Plane Mach Number Distribution Comparison Between Experimental and Analytical Results.

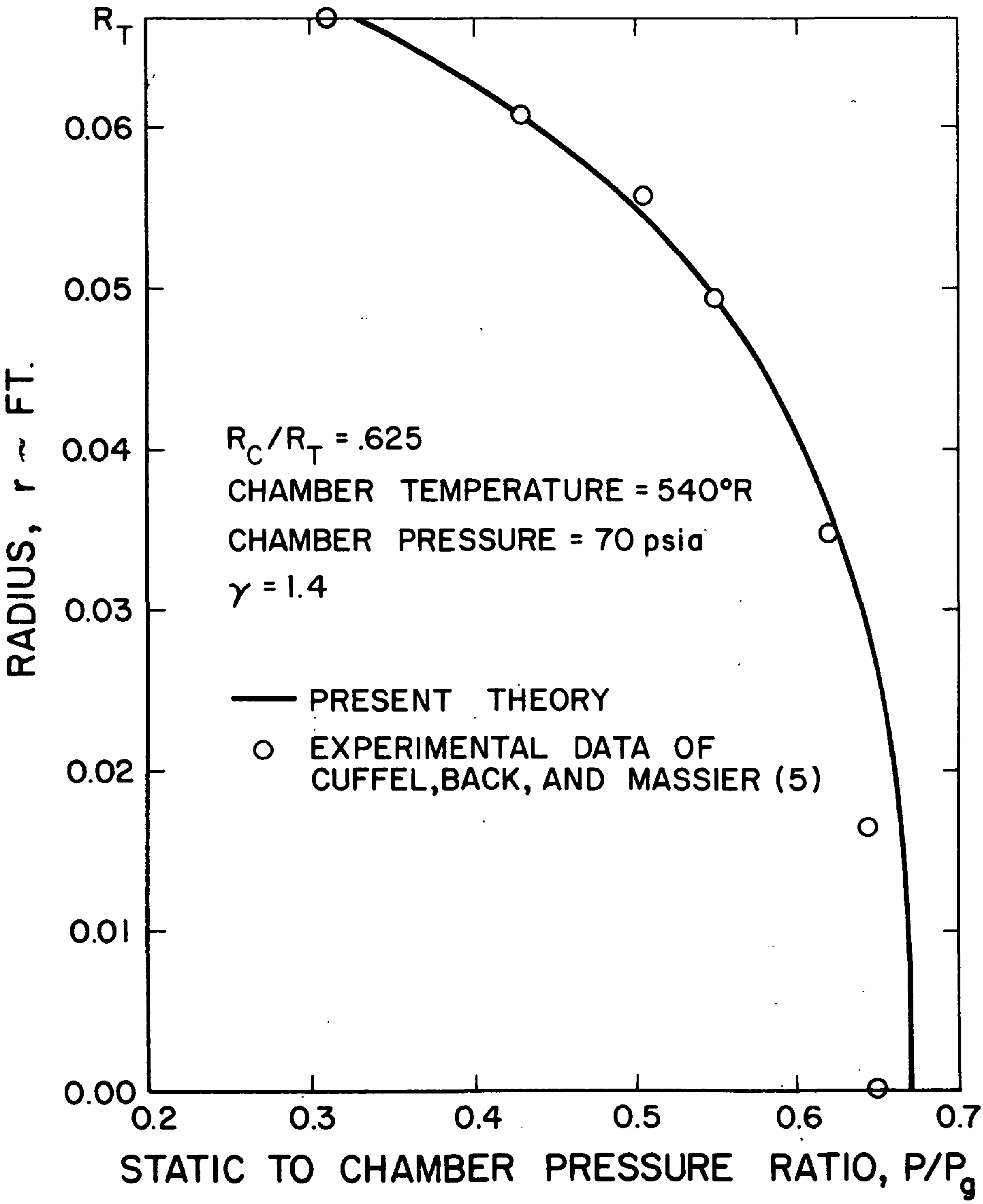


Figure 5. Throat Plane Pressure Ratio Distribution Comparison Between Experimental and Analytical Results.

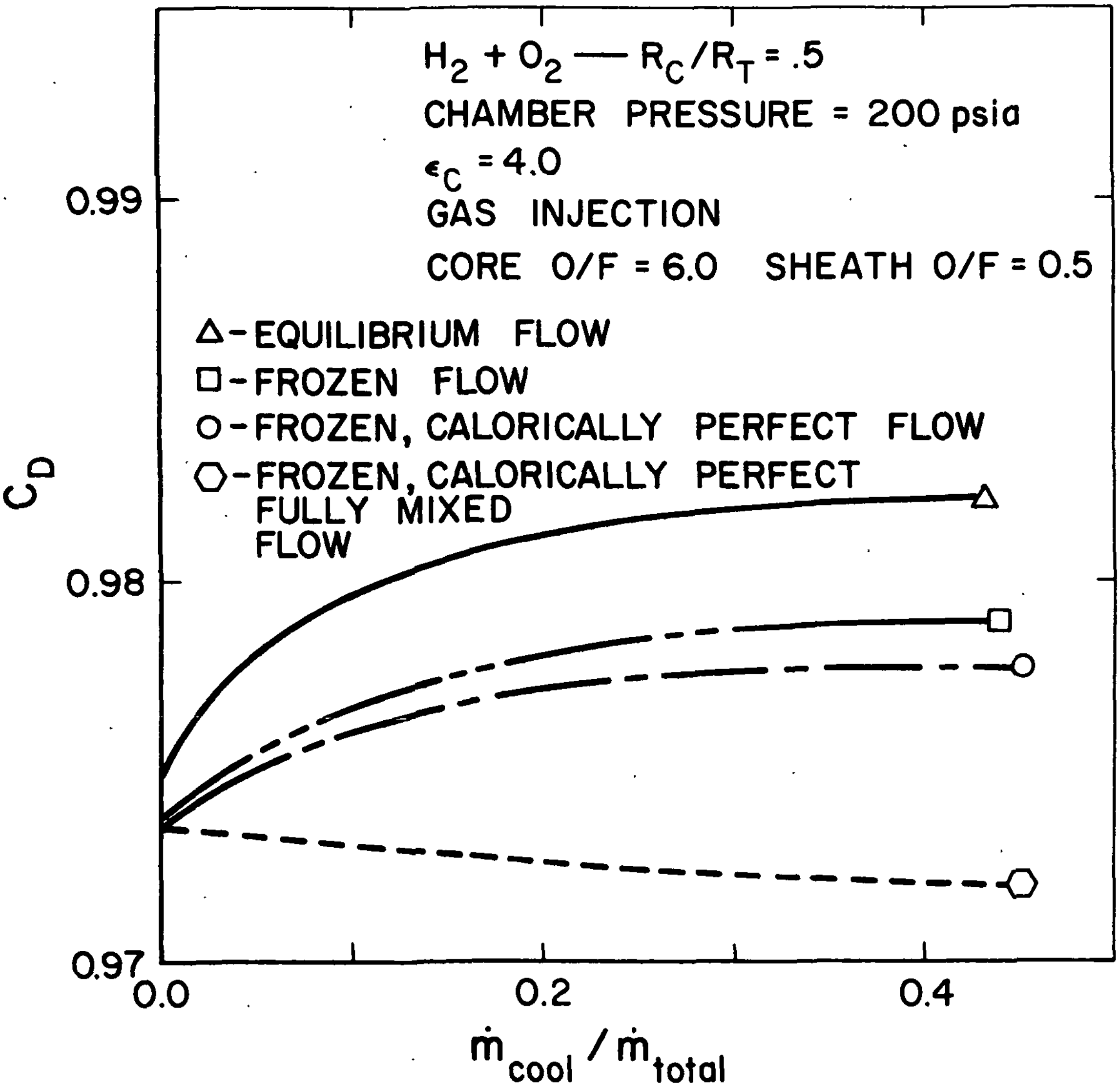


Figure 6. Variation of Discharge Coefficient with Cooling Mass Ratio for R_C/R_T = 0.5.

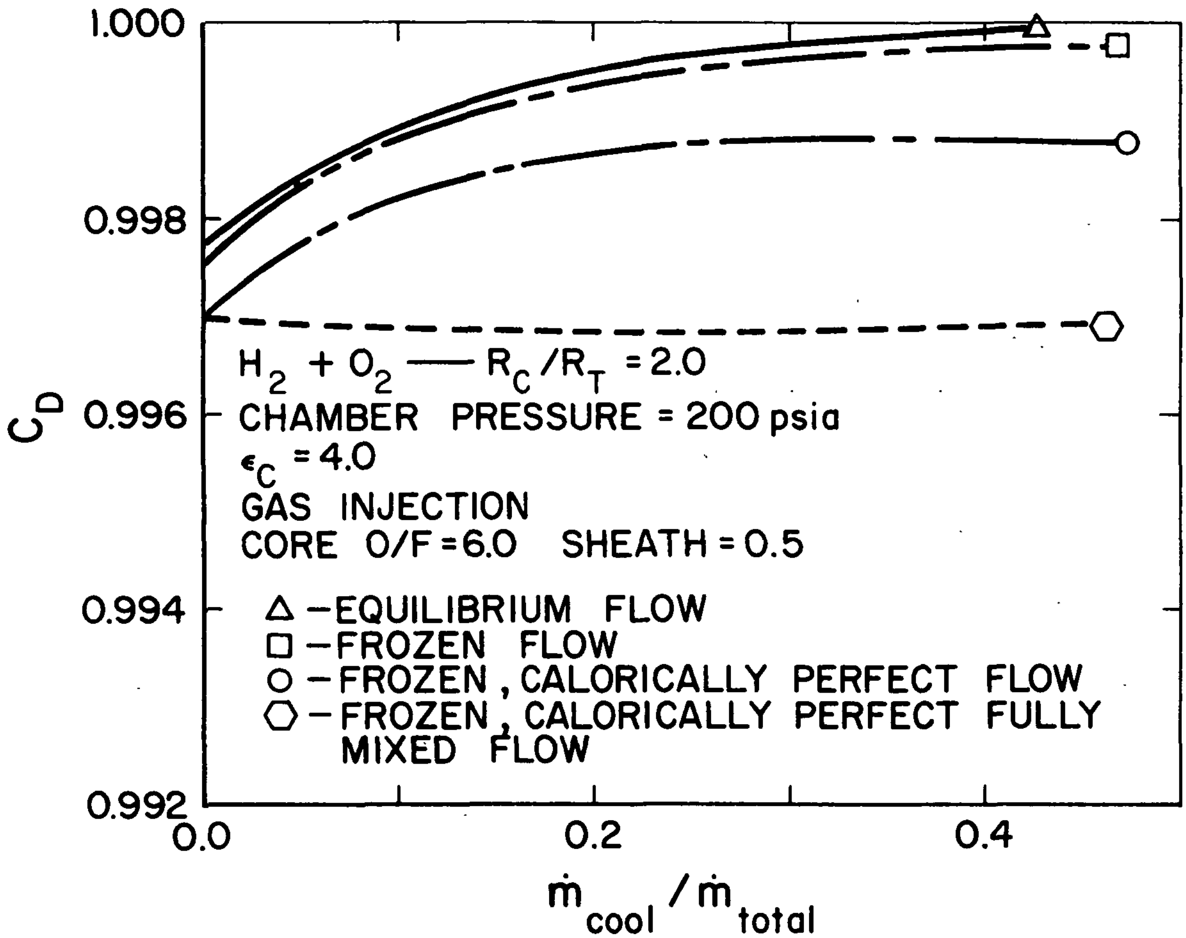


Figure 7. Variation of Discharge Coefficient with Cooling Mass Ratio for $R_C/R_T = 2.0$.

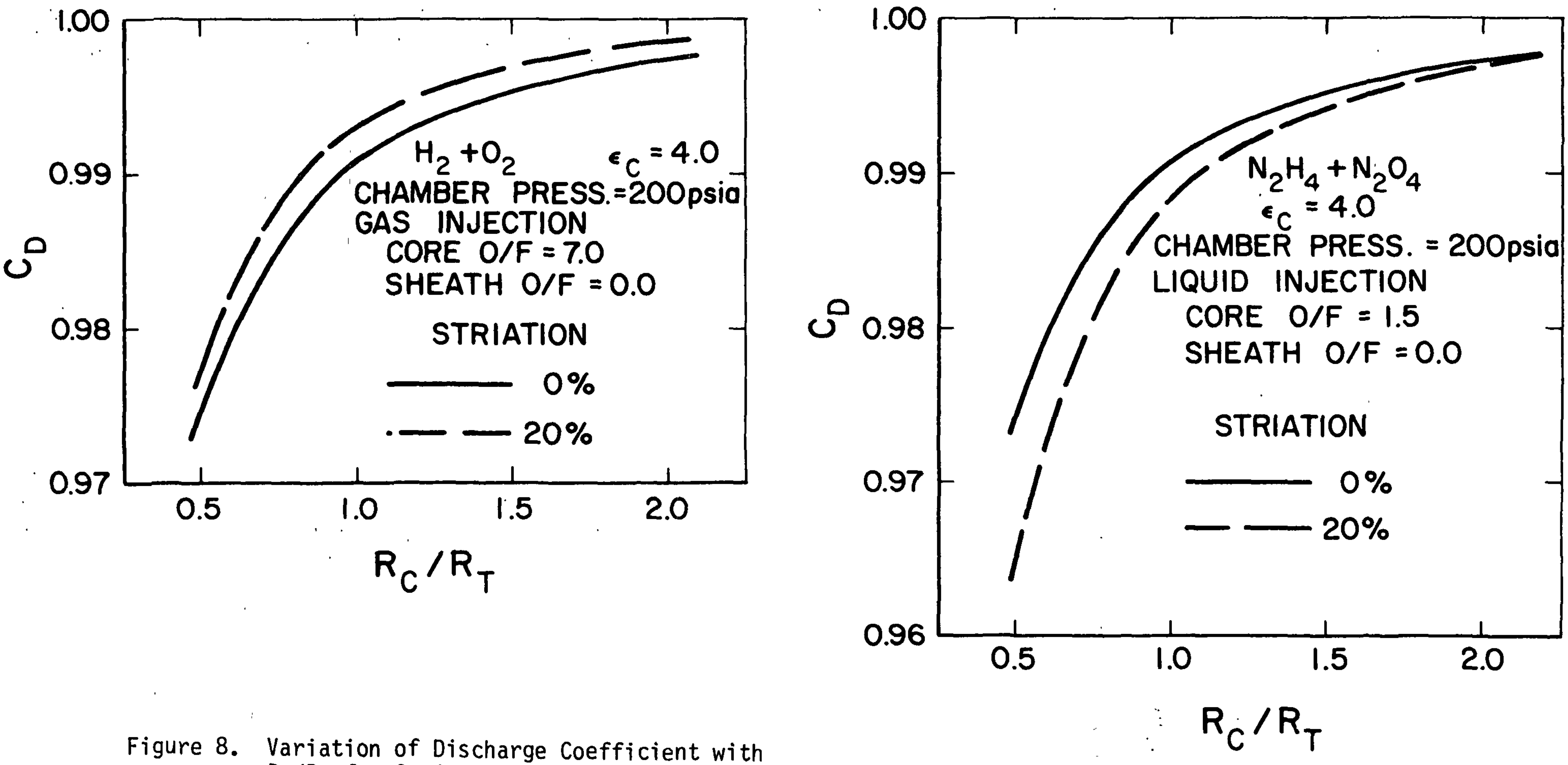


Figure 8. Variation of Discharge Coefficient with R_C/R_T for Striated and Unstriated Flow Using Different Propellants.

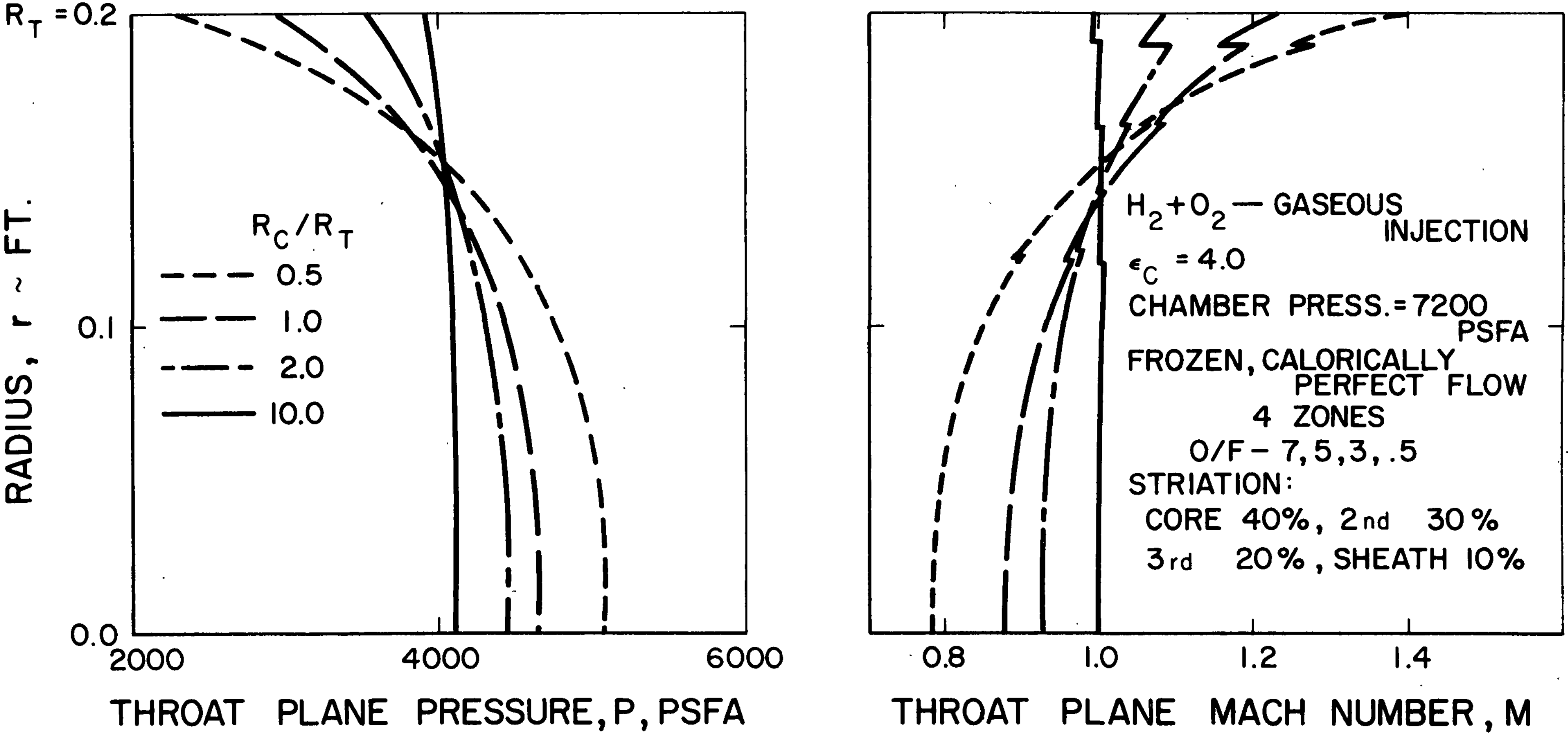


Figure 9. Variation of Throat Plane Pressure and Mach Number with R_C/R_T for Four Zone Flow.

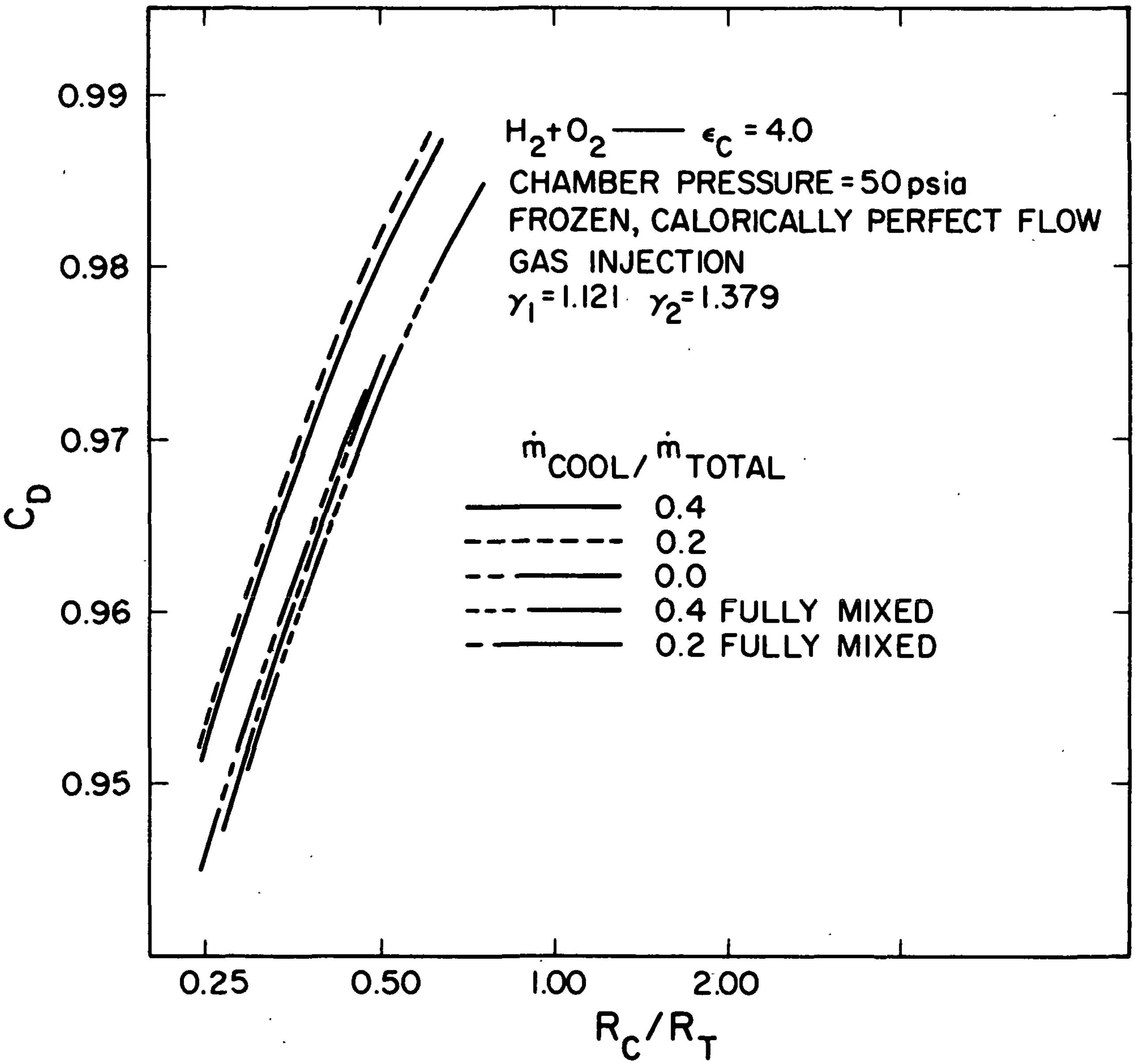


Figure 10. Discharge Coefficient for Low R_C/R_T with Various Amounts of Cooling Mass Flow.

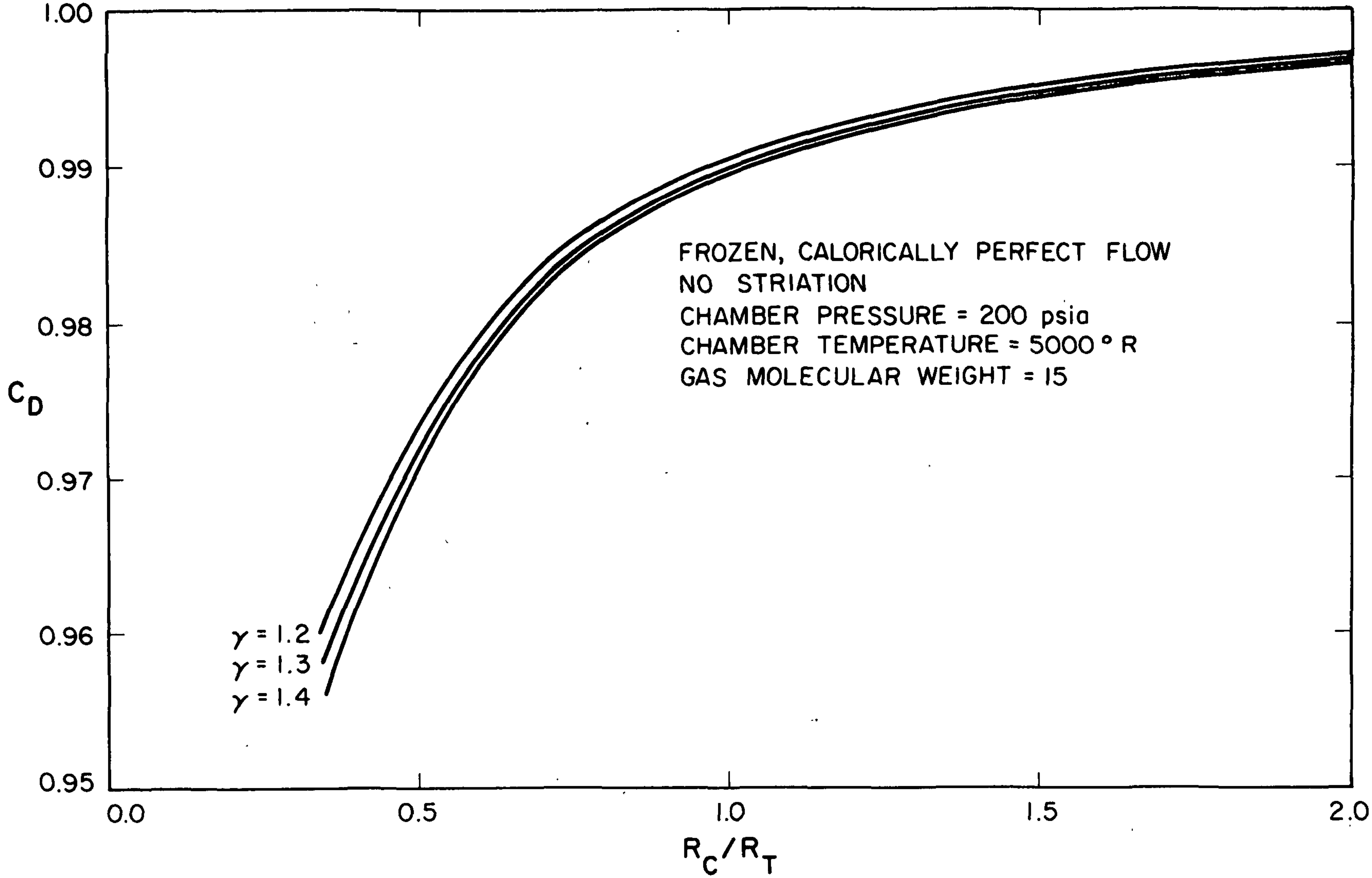


Figure 11. Discharge Coefficient for Ratio of Specific Heats = 1.2, 1.3, and 1.4.

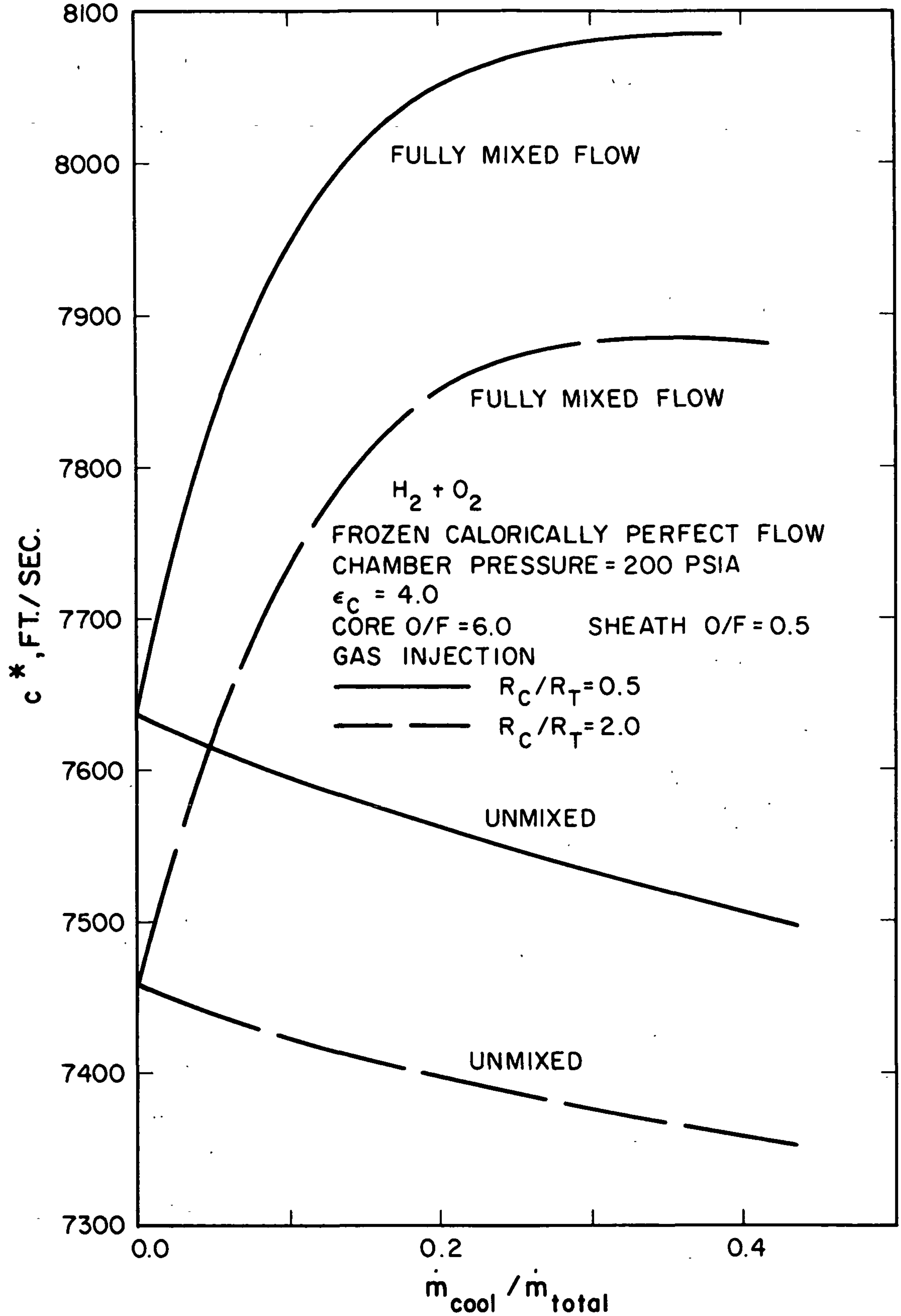


Figure 12. Characteristic Velocity For Mixed and Unmixed Flow With Chamber Static Pressure Fixed.

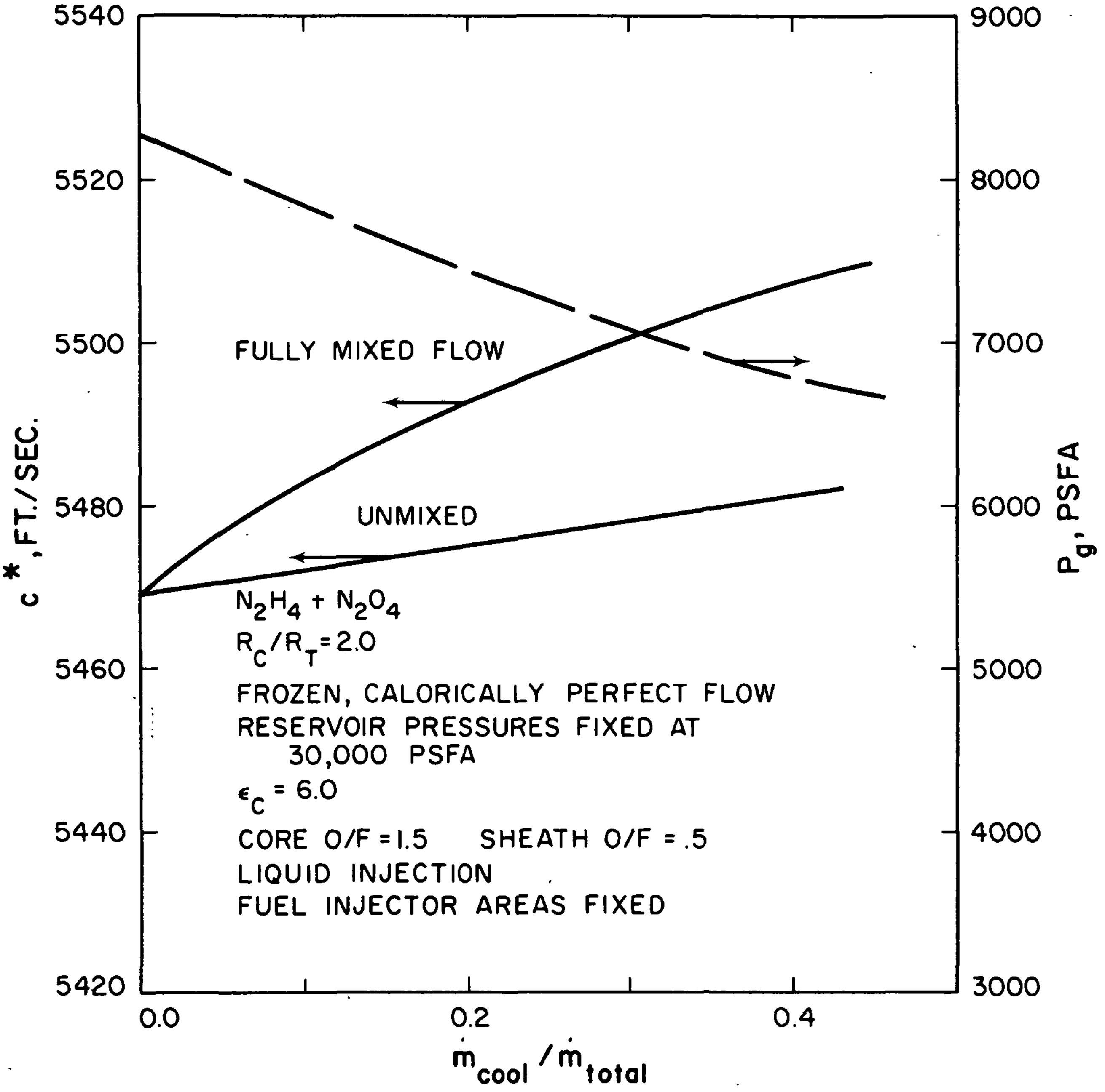


Figure 13. Characteristic Velocity For Mixed and Unmixed Flow With Reservoir Pressures Fixed.