TECHNICAL NOTE R-127

TECHNICAL NOTE R-127

A FORTRAN PROGRAM FOR HEAT TRANSFER ANALYSIS OF THE J-2 ENGINE

GPO PRICE $ ______

CFSTI PRICE(S) $ ______

Hard copy (HC) 4.00

Microfiche (MF) .75

ff 653 July 65

by J. G. Floyd
B. G. Smith
H. H. Seidel

N66 30300 (ACCESSION NUMBER) ______ (THRU) 1

107 (PAGES) ______ (CODE) 1

CR-67353 (NASA CR OR TMX OR AD NUMBER) ______ (CATEGORY) 28

March 1965

RESEARCH LABORATORIES

BROWN ENGINEERING COMPANY, INC

HUNTSVILLE, ALABAMA

[redacted]

TECHNICAL NOTE R-127

A FORTRAN PROGRAM FOR HEAT TRANSFER ANALYSIS OF THE J-2 ENGINE

January, 1965

Prepared For

PROPULSION DIVISION
P&VE LABORATORY
GEORGE C. MARSHALL SPACE FLIGHT CENTER

By

RESEARCH LABORATORIES
BROWN ENGINEERING COMPANY, INC.

Contract NAS8-5289

Prepared By

J. G. Floyd
B. G. Smith
H. H. Seidel [redacted]

ABSTRACT

30300

This report describes an IBM 7040 FORTRAN computer program for determining the heat transfer and pressure drop across the jacket of the regeneratively cooled J-2 engine. Thermochemical and heat transfer analyses were performed and are provided in two separate decks, Thermochemical Analysis Program of the J-2 Engine", Program No. SP-172, and "Heat Transfer Computation Program of the J-2 Engine", Program No. SP-173. The entire program is incorporated in "A FORTRAN Program for Heat Transfer Analysis of the J-2 Engine", Program No. SP-148.

Approved:

R. C. Watson

R. C. Watson, Jr.
Director of Research

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FORTRAN SYMBOLS

A	Effective heated surface area of tube wall
AC	Average interpolated cross-sectional area between two stations
AL	Local cross-sectional area of tube
CPI	Specific heat at constant pressure
DELH	Enthalpy difference from chamber to exit
DELPF	Pressure drop of coolant
DL	Hydraulic diameter
DLC	Average interpolated hydraulic diameter between two stations
RHO	Density of coolant
GAMMA	Ratio of specific heats at the chamber
GAMMAE	Ratio of specific heats at the exits
HC	Enthalpy at the chamber
HCAL	Film coefficient on hot gas side
HLCAL	Film coefficient on coolant side
PC	Chamber pressure
PI	Pressure at the station
PINI	Coolant inlet manifold pressure
PINJ	Fuel chamber injector pressure
SC	Entropy at the chamber
SE	Entropy at the exit
TC	Chamber temperature
TF	Adiabatic wall temperature
TGI	Local static gas temperature

LIST OF FORTRAN SYMBOLS (Cont.)

TRYS	Temperature rise of coolant across the station elements
TWGS	Wall temperature, hot gas side
TWLS	Wall temperature, coolant side
WDOTF	Fuel flow rate
WDOTO	Oxidizer flow rate
ZISPI	Specific impulse
ZK	Thermal conductivity
ZLP	Tube length element
ZM	Molecular weight
ZMR	Mixture ratio
ZMU	Dynamic viscosity

INTRODUCTION

The program described herein determines the heat transfer and pressure drop across the jacket of the regeneratively cooled J-2 engine.

In order to perform the necessary calculations to complete the program, it was important to develop two distinct phases; the thermochemical portion and the heat transfer portion.

The thermochemical part of the program provides thermodynamic parameters along the inside wall of the nozzle of the J-2 engine. These parameters are pressure, specific heat, mole fractions of the products, temperature, molecular weight, and specific impulse at exit positions or stations along the wall.

The heat transfer program utilizes the thermodynamic parameters to determine the pressure and temperature change of the coolant across the jacket of the nozzle.

In the thermochemical program a subroutine was required to determine the mole fractions of the species (H_2O, H_2, H, OH, O, O_2) and also an interpolation subroutine.

DESCRIPTION OF THERMOCHEMICAL EQUATIONS

For determining the chamber conditions the following equations were used. The chamber pressure is calculated by Equation 1 using the initially guessed values of T_c, M, and γ.

$$p_c = \dot{\omega}_F \frac{[(\dot{\omega}_o/\dot{\omega}_f)+1](R_o T_c/M g)^{\frac{1}{2}}}{A_t(\gamma)^{\frac{1}{2}}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{2(\gamma-1)}}} \qquad (1)$$

where

- $\dot{\omega}_f$ - the fuel flow rate,
- $\dot{\omega}_o$ - the oxidizer flow rate,
- R_o - universal gas constant,
- T_c - the temperature in the chamber,
- M - the molecular weight,
- g - the gravitational constant,
- A_t - the cross-sectional area at the throat, and
- γ - the ratio of specific heats.

To calculate the correct value of temperature at the corresponding chamber pressure p_c, it is necessary to make an enthalpy balance.

By iterating on temperature and going through the CONV subroutine in order to determine the mole fractions of the species H_2O, H_2, H, OH, O, O_2, the chamber pressure equation will be satisfied. With each iteration in temperature, it is necessary to interpolate for new values of enthalpy for the species $[(H^\circ - H^\circ_{298})_{H_2O}$, $(H^\circ - H^\circ_{298})_{H_2}$ ... 6 species$]$. The enthalpy balance equation is as follows:

$$n_{H_2O}(H^\circ_T - H^\circ_{298})_{H_2O} + n_{H_2}(H^\circ_T - H^\circ_{298})_{H_2} + n_H(H^\circ_T - H^\circ_{298})_H$$

$$+ n_{OH}(H^\circ_T - H^\circ_{298})_{OH} + n_O(H^\circ_T - H^\circ_{298})_O + n_{O_2}(H^\circ_T - H^\circ_{298})_{O_2} =$$

$$- (-57.798\, n_{H_2O} + 52.089 n_H + 9.33\, n_{OH} + 59.55\, n_O) \tag{2}$$

$$+ \left(-3.08 \frac{100}{32} \frac{\dot{\omega}_o}{\dot{\omega}_o + \dot{\omega}_F} - 1.887 \frac{100}{2.016} \frac{\dot{\omega}_F}{\dot{\omega}_o + \dot{\omega}_F}\right)$$

When these two terms are equal within ±0.01, the chamber temperature is correct.

The molecular weight is determined by $M = 100/n_T$,

where n_T - the total number of moles . (3)

With the chamber temperature the values of specific heat, C_p, must be interpolated for all species.

The ratio of specific heats can be calculated by the equation

$$\gamma = \frac{n_{H_2O}\, C_{p\,H_2O}^{\;T} + n_{H_2}\, C_{p\,H_2}^{\;T} + n_H\, C_{p\,H}^{\;T} + n_{OH}\, C_{p\,OH}^{\;T} + n_O\, C_{p\,O}^{\;T} + n_{O_2}\, C_{p\,O_2}^{\;T}}{n_{H_2O}\, C_{p\,H_2O}^{\;T} + n_{H_2}\, C_{p\,H_2}^{\;T} + n_H\, C_{p\,H}^{\;T} + n_{OH}\, C_{p\,OH}^{\;T} + n_O\, C_{p\,O}^{\;T} + n_{O_2}\, C_{p\,O_2}^{\;T} - 1.987 n_T} \tag{4}$$

If the present value of chamber pressure is within ±0.01 of the previous value, all variables are within tolerance; if not, the program uses the newly calculated values of T_c, M, γ, and p_c and returns to the enthalpy balance for another pass through the program.

When all convergence requirements are satisfied, chamber entropy is calculated by first interpolating for values of entropy S for all species at the corresponding T_c.

Then the entropy of the gas mixture in the chamber is

$$S_c = n_{H_2O} S^\circ_{H_2O}{}^{T_c} + n_{H_2} S^\circ_{H_2}{}^{T_c} + n_H S^\circ_H{}^{T_c} + n_{OH} S^\circ_{OH}{}^{T_c} + n_O S^\circ_O{}^{T_c}$$

$$+ n_{O_2} S^\circ_{O_2}{}^{T_c} - 1.987\, n_T \left(\frac{n_{H_2O}}{n_T} \ln \frac{n_{H_2O}}{n_T} + \frac{n_{H_2}}{n_T} \ln \frac{n_{H_2}}{n_T} \right. \tag{5}$$

$$\left. + \frac{n_H}{n_T} \ln \frac{n_H}{n_T} + \frac{n_{OH}}{n_T} \ln \frac{n_{OH}}{n_T} + \frac{n_O}{n_T} \ln \frac{n_O}{n_T} + \frac{n_{O_2}}{n_T} \ln \frac{n_{O_2}}{n_T} + \ln P_c \right).$$

After the chamber conditions have been calculated the program is ready to calculate the exit conditions.

The exit pressure ratios are determined by the following equation:

$$\left\{ \frac{2\gamma}{\gamma - 1} \left[1 - \left(\frac{1}{p_c/p_e} \right)^{\frac{\gamma-1}{\gamma}} \right] \right\}^{\frac{1}{2}} = \frac{(\gamma)^{\frac{1}{2}} \left[2/(\gamma+1) \right]^{\frac{\gamma+1}{2(\gamma-1)}}}{A_e/A_t} \left(\frac{p_c}{p_e} \right)^{\frac{1}{\gamma}} . \tag{6}$$

This implicit equation has two solutions of p_c/p_e for only one area ratio. The solution will be obtained by the trial and error method using a starting pressure ratio of 1.01 for the convergent part of the nozzle down to the throat and 2.0 for the divergent part of the nozzle from the throat down to the nozzle end.

$$\gamma_e = \gamma_c$$

(γ at the exit is equal to γ at the chamber for the initial guess.)

A_e/A_t - area ratio at the exit .

The preceding equation is iterated until the convergence tolerance is within ±0.01.

The pressure at the exit is determined by

$$p_e = \frac{1}{(p_c/p_e)} p_c \quad . \tag{7}$$

When γ has converged to within ±0.0001, the conditions are satisfied. If convergence is not within tolerance, the program returns to find a new p_c/p_e and the process continues.

After all convergence conditions are satisfied, the molecular weight, specific heat, and specific impulse are calculated at the station by the following equations:

Molecular Weight

$$M(i) = 100/n_T \tag{8}$$

Specific Heat

$$C_{P(i)} = \frac{1}{M\, n_T}\left(n_{H_2O}\, C_{p\,H_2O} + n_{H_2}\, C_{p\,H_2} + n_H\, C_{p\,H} + n_{OH}\, C_{p\,OH} + n_O\, C_{p\,O} + n_{O_2}\, C_{p\,O_2}\right) \tag{9}$$

where n_{H_2O}, n_{H_2}, ..., n_{O_2} are the moles of the species and $C_{p\,H_2O}$, $C_{p\,H_2}$, ... , $C_{p\,O_2}$ are the specific heats at constant pressure of the species at the corresponding station and temperature.

Now it is necessary to interpolate for new values of enthalpy for the species at the chamber and the exits, respectively. The enthalpy difference from the chamber to the exit is

$$\Delta H = H_c - H_e \quad .$$

From this the specific impulse is

$$Isp(i) = 9.33(10.0\ \Delta H)^{\frac{1}{2}} \quad .$$

DESCRIPTION OF HEAT TRANSFER EQUATIONS

Some variables of the following equations are subscripted either by i or j or both. These subscripts denote the station under consideration and the specie, respectively.

The equation described below is used to determine the dynamic viscosity of the gas.

$$\mu = \frac{M}{\sum_{j=1}^{6} \frac{n_{ij} M_{ij}}{\mu_{ij}}} \tag{10}$$

where

M - total molecular weight of gases,

n_{ij} - number of moles of the jth species at the station i,

M_{ij} - molecular weight of the species at the station, and

μ_{ij} - viscosity of the species at the station.

To determine the thermal conductivity of the gas, the following equation was employed:

$$k = \mu\left(C_p + 1.25 \frac{1.987}{M}\right) \tag{11}$$

where

μ - dynamic viscosity,

C_p - specific heat at constant pressure, and

M - molecular weight.

The adiabatic wall temperature is expressed by

$$T_f = T_g + Isp^2 \frac{9.81}{854.0}\left(\frac{\mu}{k\,C_p}\right)^{\frac{1}{2}} \quad , \tag{12}$$

where

T_g - main stream gas temperature, and

Isp - specific impulse .

The specific heat at the station is calculated by

$$C_p = \sum \frac{n_{ij}\, C_{p\,ij}}{M_i} \tag{13}$$

where

n_{ij} - number of moles of the jth specie at the station i, and

$C_{p\,ij}$ - specific heat of the specie at the station.

The film coefficient on the hot gas side is represented by

$$h = 0.0162\, C_p^{0.82}\, k^{0.18} \left(\frac{p\, Isp\; 9.81\; M}{847.9\; T_g}\right)^{0.82} \frac{1}{[D_t(A_e/A_t)^{\frac{1}{2}}]^{0.18}} \left(\frac{T_f}{T_{wg}}\right)^{0.35} \tag{14}$$

where

D_t - diameter at the throat,

A_e/A_t - area ratio, and

T_{wg} - wall temperature on hot gas side.

The cross-sectional area of the coolant tube is determined in the following manner:

$$A_\ell = \pi N^2 + 2N(H - 2N) \tag{15}$$

where

N - tube parameter, and

H - tube parameter .

The hydraulic diameter of the coolant tube is represented by

$$D_\ell = \frac{2\pi N^2 + 4HN - 8N^2}{\pi N + H - 2N} \quad . \tag{16}$$

The film coefficient on the coolant side is determined by

$$h_\ell = 0.021 k_\ell^{0.67}\left(\frac{\dot{\omega}_f}{n A_\ell}\right)^{0.8} \frac{1}{D_\ell^{0.2}} \frac{1}{\mu_\ell^{0.47}} C_{p\ell}^{0.33}\left[\frac{\mu_\ell C_{p\ell}/k_\ell}{\mu_{w\ell} C_{pw\ell}/k_{w\ell}}\right]^{0.25} \tag{17}$$

where

k_ℓ - thermal conductivity of the coolant,

n - number of tubes,

A_ℓ - cross-sectional area of the coolant tube,

D_ℓ - hydraulic diameter of the coolant tube,

μ_ℓ - viscosity of the coolant

$C_{p\ell}$ - specific heat of the coolant,

$\mu_{w\ell}$ - viscosity of the coolant in the neighborhood of the wall,

$C_{pw\ell}$ - specific heat of coolant in the neighborhood of the wall, and

$k_{w\ell}$ - thermal conductivity of coolant in the neighborhood of the wall.

The wall temperature on the coolant side is calculated by

$$T_{w\ell} = T_\ell + \frac{(T_f - T_\ell)}{1 + h_\ell[(e_m/k_m) + (1/h)]} \tag{18}$$

where

T_ℓ - coolant temperature,

T_f - adiabatic wall temperature,

h_ℓ - coolant film coefficient,

e_m - wall thickness,

k_m - thermal conductivity of wall material, and

h - hot gas side film coefficient .

The wall temperature for the hot gas side is determined by

$$T_{wg} = T_f - \frac{T_f - T_\ell}{1 + h[(e_m/k_m) + (1/h_\ell)]} \quad . \tag{19}$$

To determine the tube length from one station to another along the wall of the nozzle, it is necessary to use the nozzle contour coordinates X(axial) and Z(radial). These positions along the wall are designated by i for the station presently being considered and i - 1 for the previous station. The tube length from one station to another is calculated by

$$L_p = \left[\left|(X_{i-1} - X_i)^2\right| + \left|(Z_{i-1} - Z_i)^2\right| \right]^{\frac{1}{2}} \quad . \tag{20}$$

To average the cross-sectional area of the tube, it is necessary to calculate

$$A_c = (Z_i H_i + N_{i-1} H_{i-1}) - 0.4292(N_i^2 + N_{i-1})^2 \quad . \tag{21}$$

To average the hydraulic diameter of the tube, let

$$D_{\ell c} = \frac{4.0[N_i H_i + N_{i-1} H_{i-1} - 0.4292(N_i^2 + N_{i-1}^2)]}{\pi(N_i + N_{i-1}) + H_i + H_{i-1} - 2(N_i + N_{i-1})} \quad . \tag{22}$$

The effective heated surface area of the wall is represented by

$$A = \frac{2.2}{2}(N_i + N_{i-1}) L_p \quad . \tag{23}$$

The temperature change of the coolant is calculated by

$$T_{rys} = \frac{n A}{2 \dot{\omega}_f} \left[h_{i-1}(T_{f\,i-1} - T_{wg\,i-1}) + h_i(T_{f\,i}\ T_{wg\,i}) \right] \quad . \tag{23}$$

The total then equals

$$T_L = T_{L\,in} + \sum T_{rys} \quad . \tag{24}$$

The pressure drop across a tube element between two stations is determined by the following relation. When Equation 25 is equalized by iterating on an initially guessed ΔP, the pressure drop is calculated.

$$\frac{1}{\{[9.81 \cdot 10^5 \ \rho \ n^2 A_c^{\,2} (D_{\ell c}/L_p) \, \Delta P] / 2 \dot{\omega}_f\}^{\frac{1}{2}}} = 2.28$$

$$+ 4 \log \left\{ \left[\frac{9.81 \cdot 10^5 \ \rho \ n^2 A_c^{\,2} (D_{\ell c}/L_p) \, \Delta P}{2 \dot{\omega}_f^{\,2}} \right]^{\frac{1}{2}} \frac{\dot{\omega}_f D_{\ell c}}{n A_c \mu} \right\} \tag{25}$$

$$- 4 \log \left\{ \frac{e}{D} \left[\frac{9.81 \cdot 10^5 \ \rho \ n^2 A_c^{\,2} (D_{\ell c}/L_p) \, \Delta P}{2 \dot{\omega}_f^{\,2}} \right]^{\frac{1}{2}} \frac{\dot{\omega}_f D_{\ell c}}{n A_c \mu} + 4.67 \right\} .$$

RESULTS

In order to establish the accuracy of this program, some test runs were conducted for comparing the results with actual test firing data. Figures 1, 2, and 3 show the chamber pressure, main fuel injector pressure and the main fuel injector temperature as a function of time during the build-up transient. It can be seen that the calculated points are in good agreement with the measured values.

For representing a mixture ratio range from 1 to 6 at different fuel flow rates, a whole series of computer runs have been made and the results plotted on graphs. Figure 4 shows the main fuel injector pressure depending on fuel flow rate with different mixture ratios as parameters. Figure 5 represents the main fuel injector temperature depending on fuel flow rate with different mixture ratios as parameters. Figure 6 determines the chamber pressure as a function of mixture ratio with different fuel flow rates as parameters. Figure 7 gives the chamber pressure against fuel flow rates at different mixture ratios as parameters. Figure 8 represents the relationship of the main fuel injector pressure depending on mixture ratio and fuel flow rate.

In order to satisfy the requirement of the project, it was necessary to obtain curve fit equations to represent, at different mixture ratios, the main fuel injector pressure and main fuel injector temperature as a function of the fuel flow rate as plotted on Figures 4 and 5.

In order to show the accuracy between the original calculated points in Figures 4 and 5 and the points obtained by the curve fit equation, Figures 9 and 10 have been established.

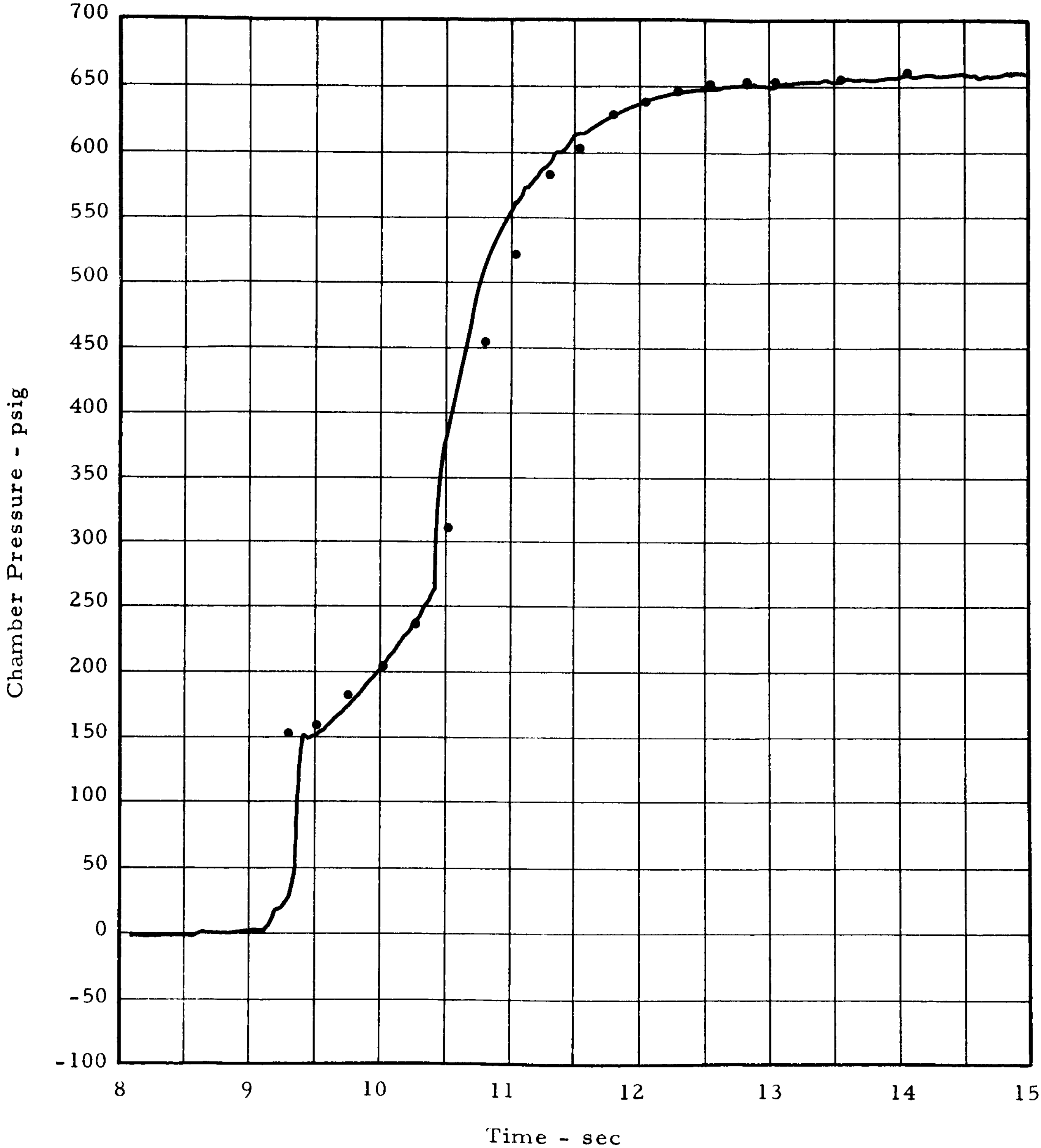

Figure 1. Chamber Pressure Measured at a Test Firing
(The dots are calculated points.)

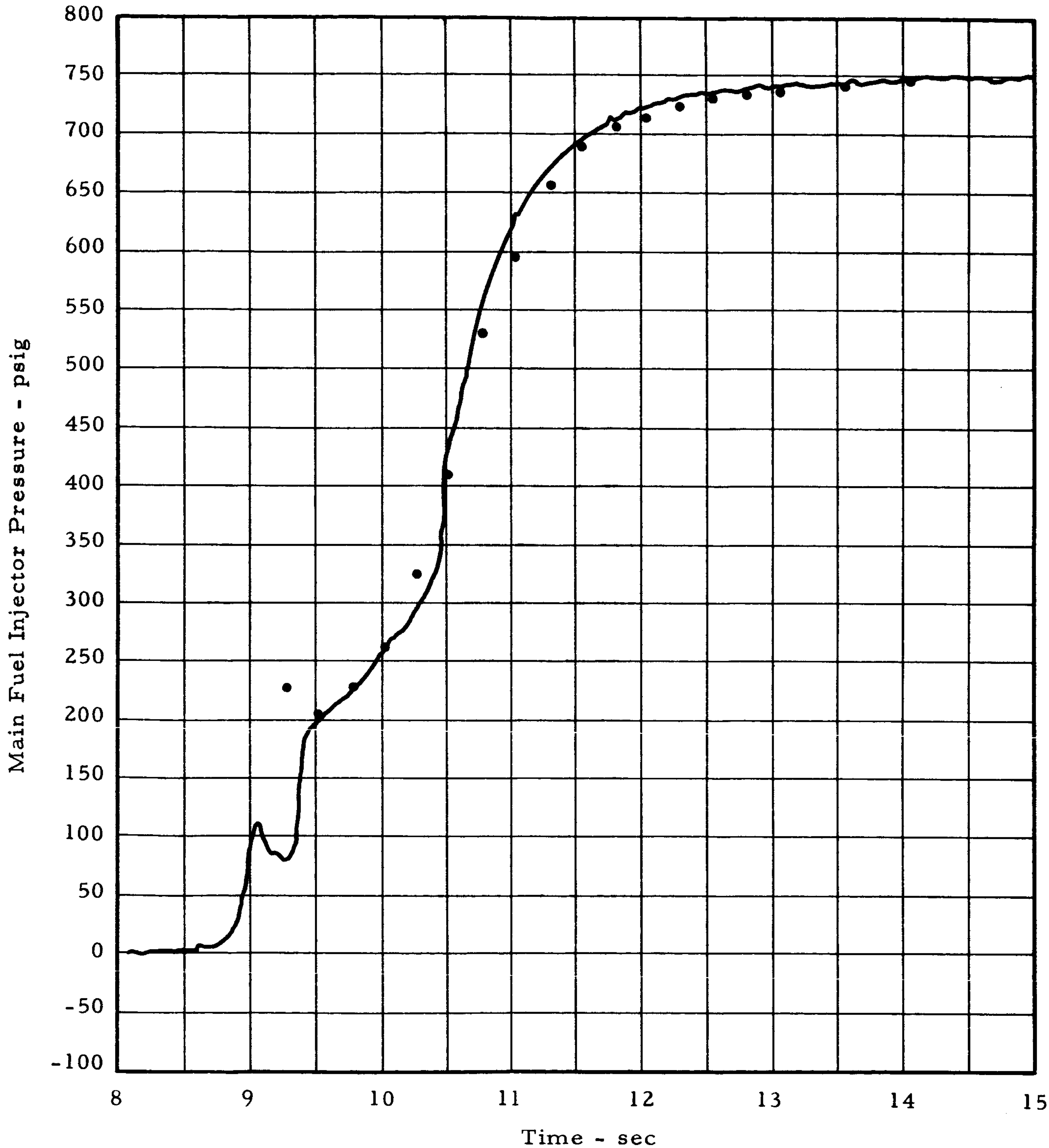

Figure 2. Main Fuel Injector Pressure Measured at a Test Firing (The dots are calculated points.)

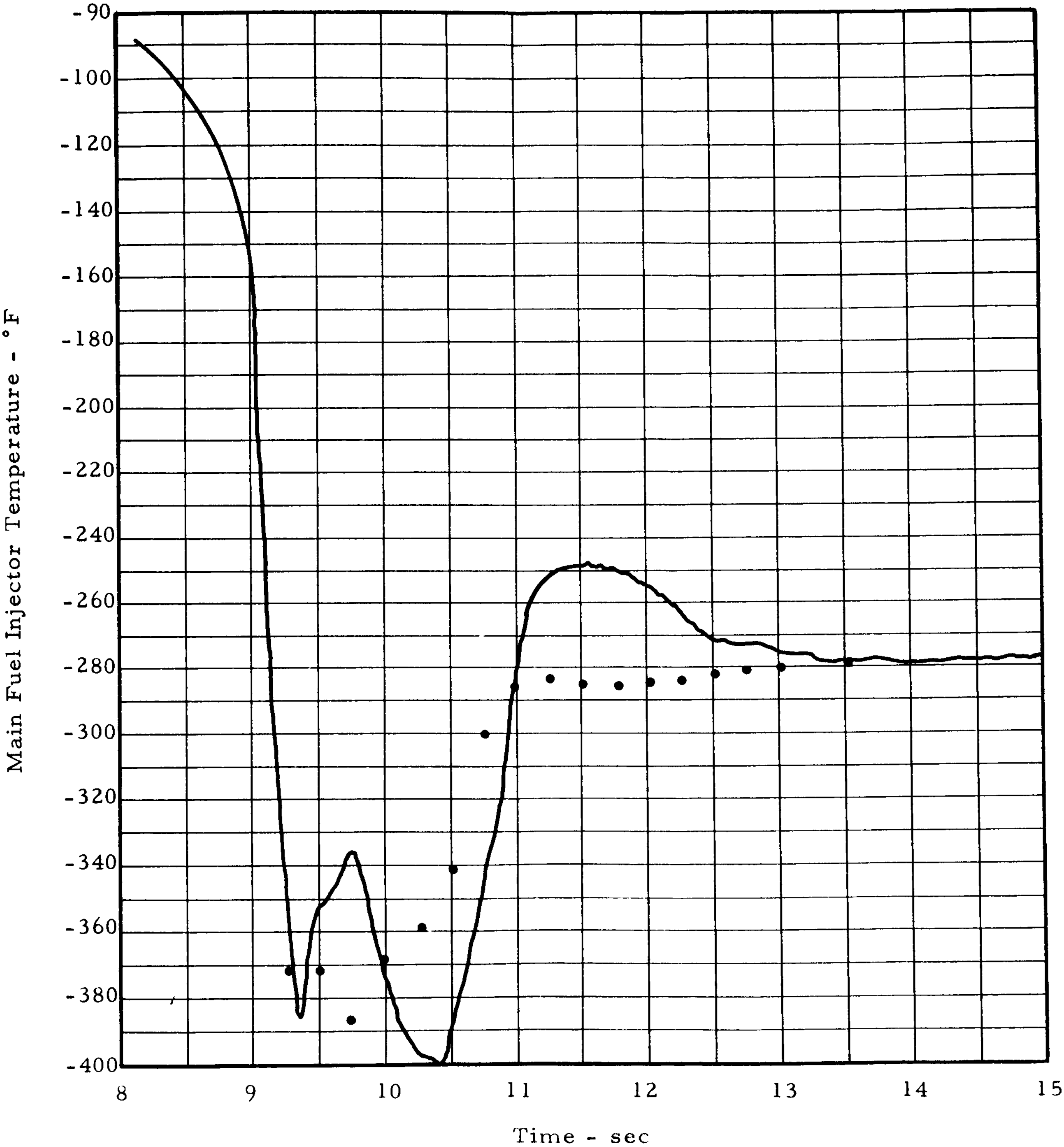

Figure 3. Main Fuel Injector Temperature Measured at a Test Firing (The dots are calculated points.)

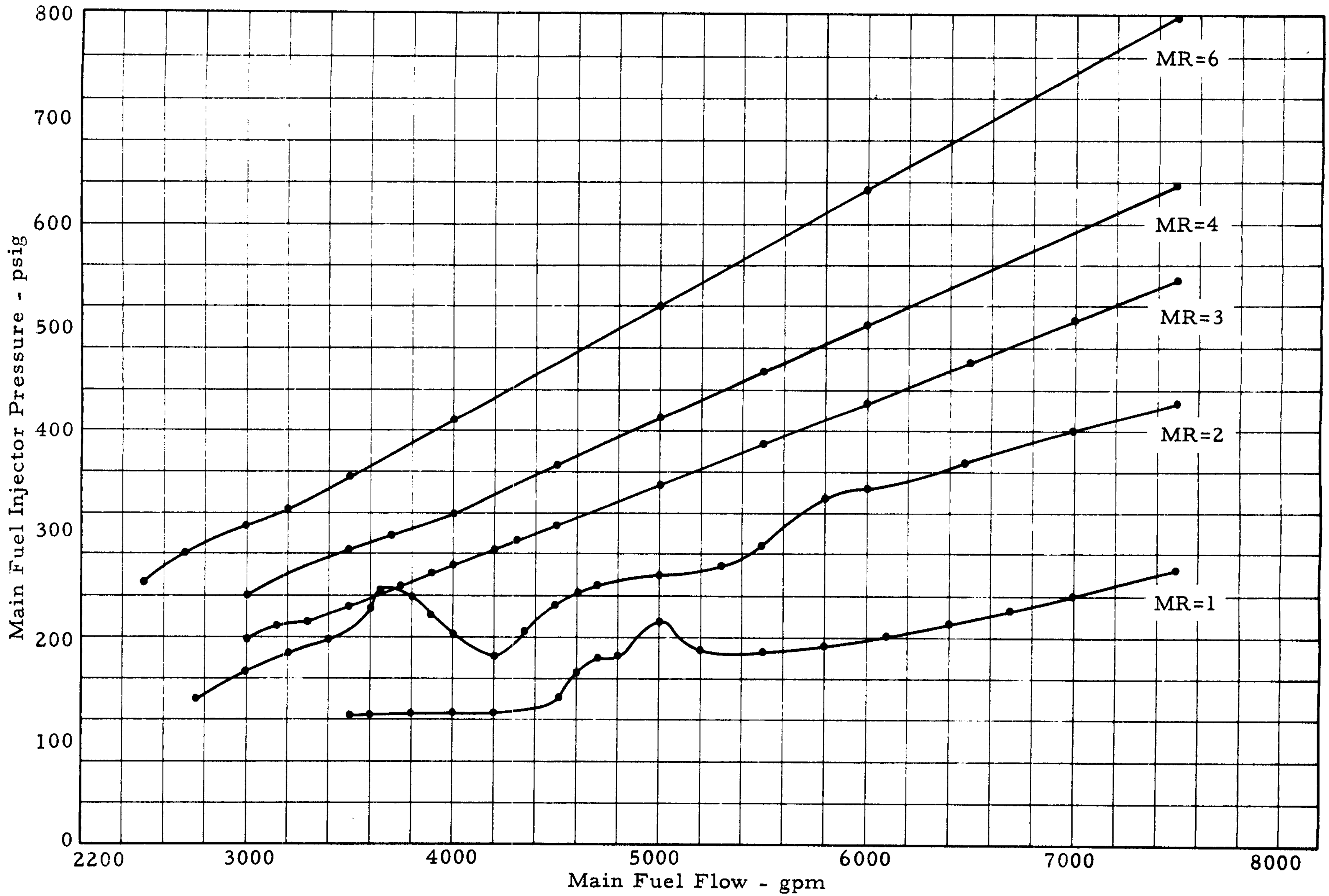

Figure 4. Calculated Main Fuel Injector Pressure Depending on Fuel Flow

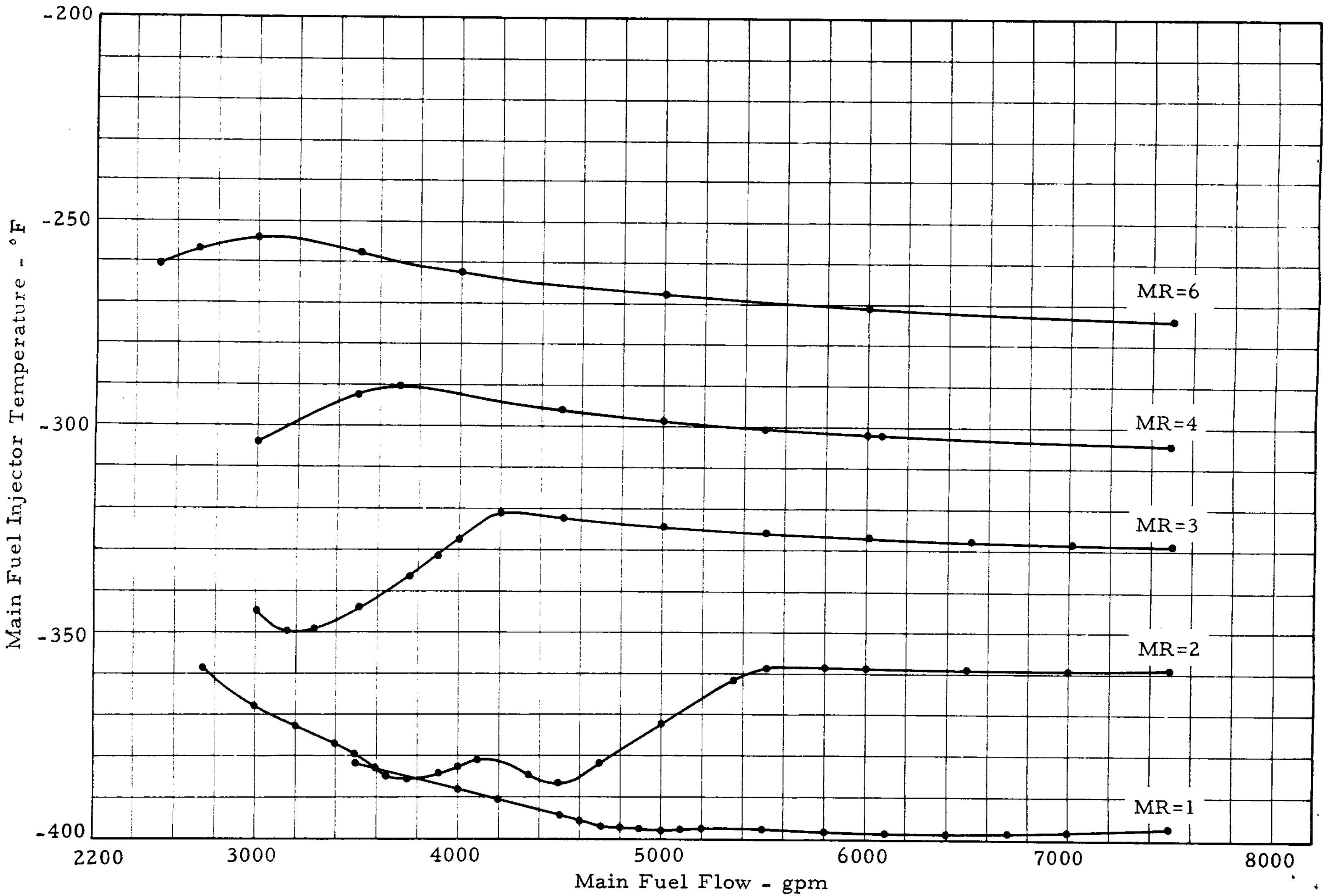

Figure 5. Calculated Main Fuel Injector Temperature Depending on Fuel Flow

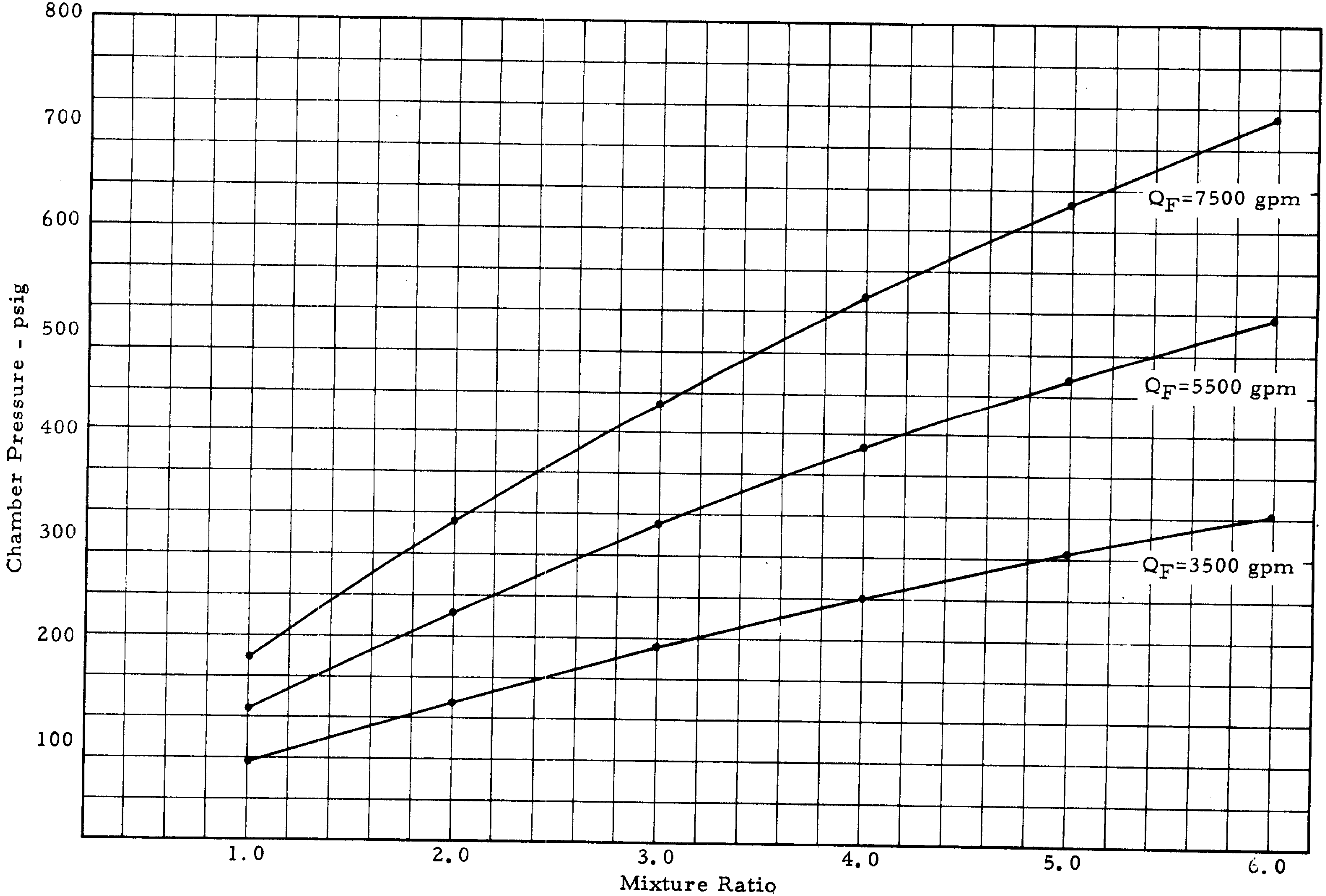

Figure 6. Calculated Chamber Pressure Depending on Mixture Ratio

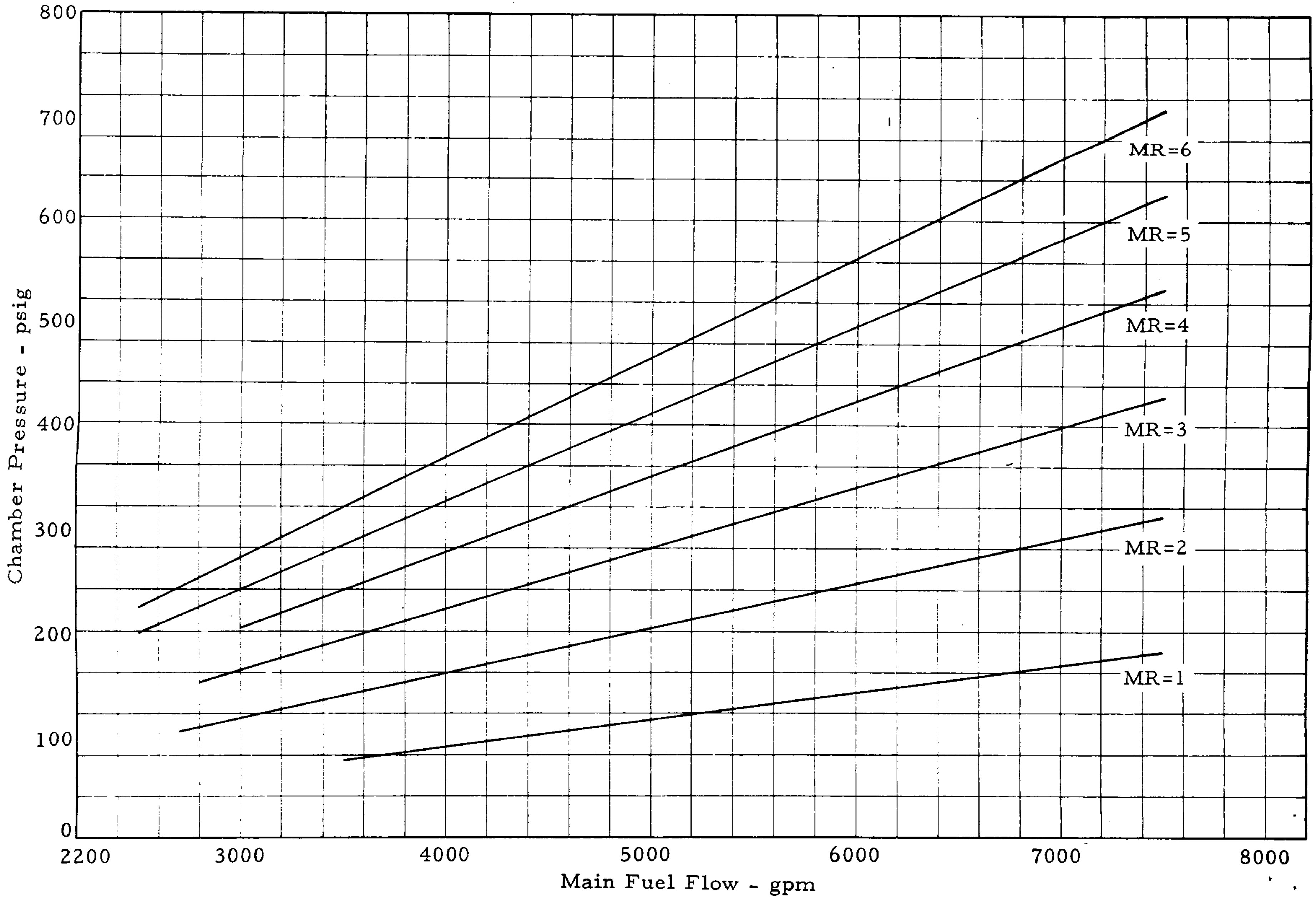

Figure 7. Calculated Chamber Pressure Depending on Fuel Flow

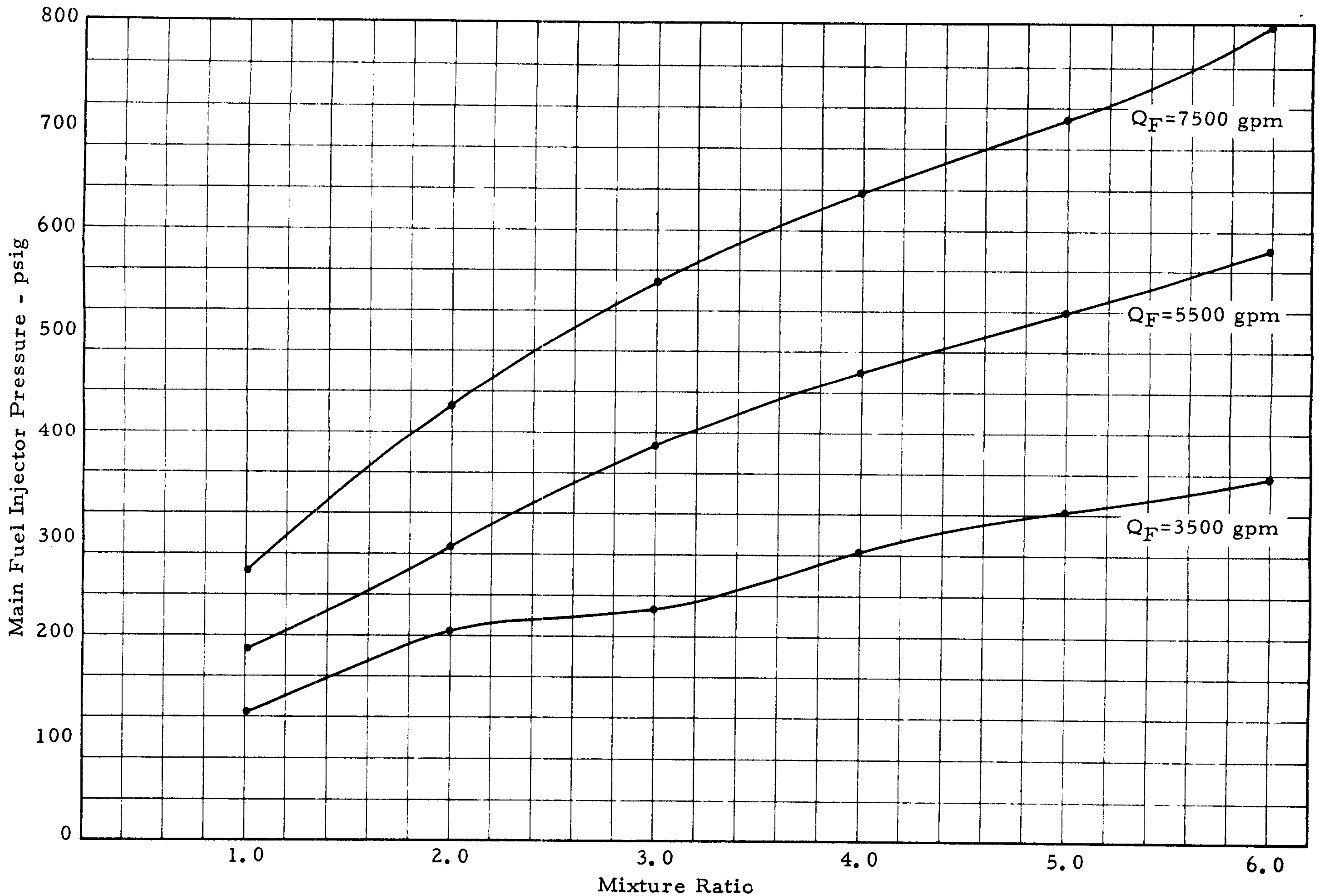

Figure 8. Calculated Main Fuel Injector Pressure Depending on Mixture Ratio

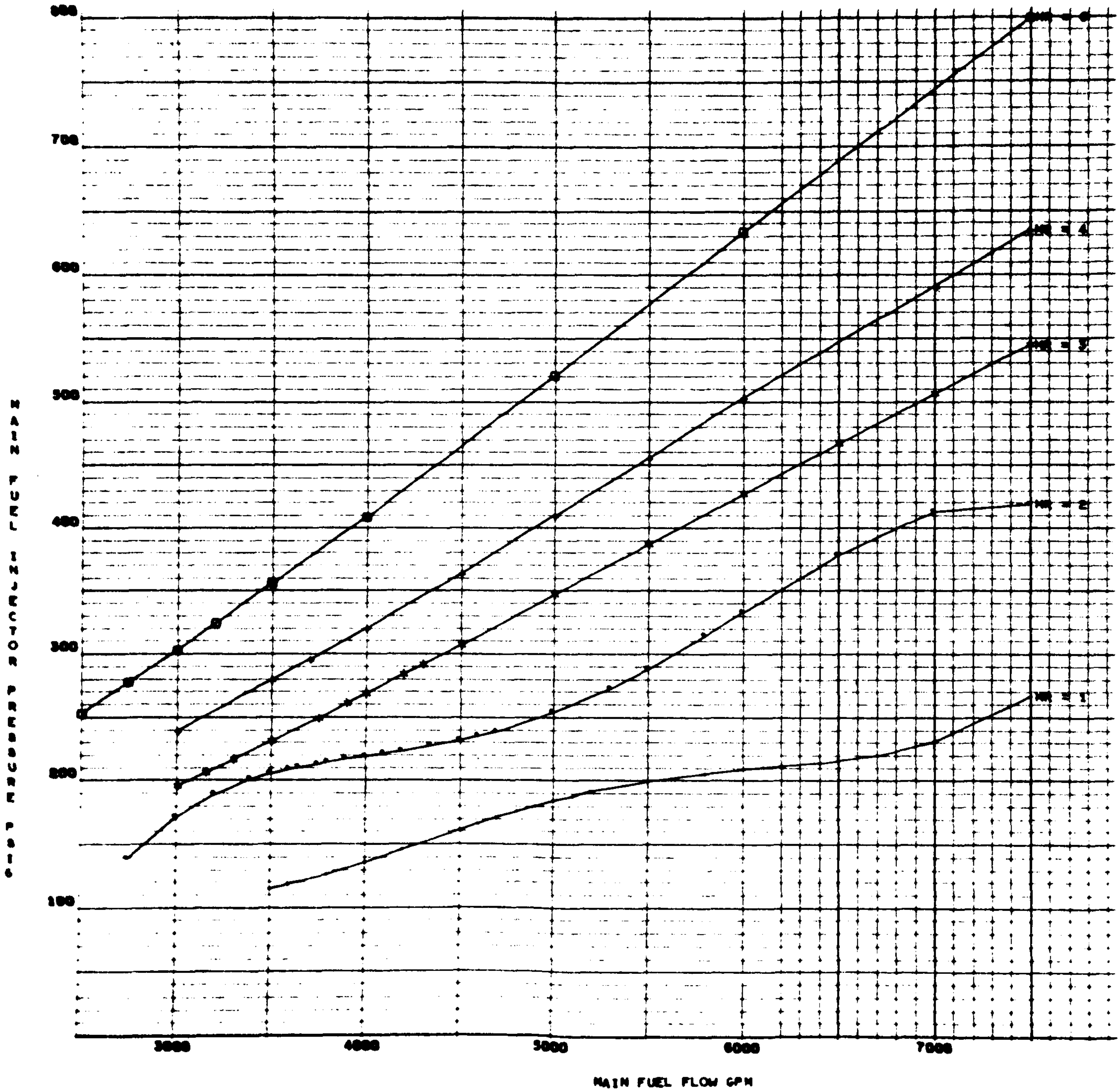

Figure 9. Curve Fitted Main Fuel Injector Pressure Depending on Fuel Flow

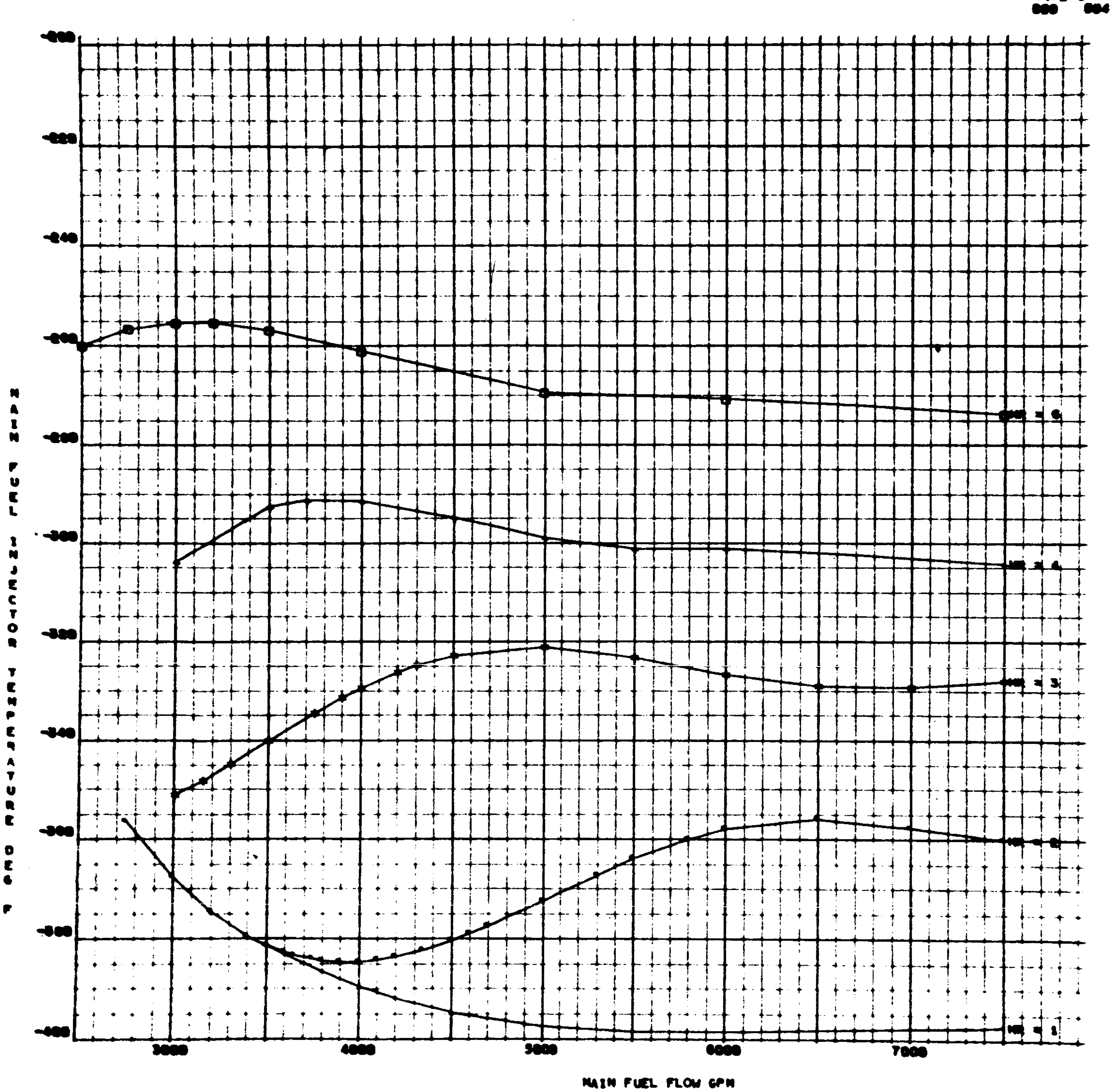

Figure 10. Curve Fitted Main Fuel Injector Temperature Depending on Fuel Flow

Polynomial equations for the pressure and temperature of the hydrogen at the injector at different mixture ratios are as follows:

MR = 1

$$P_{inj} = 1766.8362 - 1.4727344 \cdot Q_F + 0.46546086 \cdot 10^{-3} \cdot Q_F^2$$

$$- 0.61659711 \cdot 10^{-7} \cdot Q_F^3 + 0.29632222 \cdot 10^{-11} \cdot Q_F^4$$

$$T_{inj} = -197.23232 - 0.080679908 \cdot Q_F + 0.3546776 \cdot 10^{-5} \cdot Q_F^2$$

$$+ 0.24545954 \cdot 10^{-8} \cdot Q_F^3 - 0.39784219 \cdot 10^{-12} \cdot Q_F^4$$

$$+ 0.18064644 \cdot 10^{-16} \cdot Q_F^5 \quad .$$

MR = 2

$$P_{inj} = -2822.742 + 2.7243554 \cdot Q_F - 0.93611755 \cdot 10^{-3} \cdot Q_F^2$$

$$+ 0.15020497 \cdot 10^{-6} \cdot Q_F^3 - 0.10770095 \cdot 10^{-10} \cdot Q_F^4$$

$$+ 0.26024543 \cdot 10^{-15} \cdot Q_F^5$$

$$T_{inj} = - 125.65396 - 0.03017783 \cdot Q_F - 0.71595675 \cdot 10^{-4} \cdot Q_F^2$$

$$+ 0.27774318 \cdot 10^{-7} \cdot Q_F^3 - 0.36669264 \cdot 10^{-11} \cdot Q_F^4$$

$$+ 0.16453136 \cdot 10^{-15} \cdot Q_F^5 \quad .$$

MR = 3

$$P_{inj} = 234.94359 - 0.15915683 \cdot Q_F + 0.79991948 \cdot 10^{-4} \cdot Q_F^2$$

$$- 0.1352253 \cdot 10^{-7} \cdot Q_F^3 + 0.11607286 \cdot 10^{-11} \cdot Q_F^4$$

$$- 0.40622326 \cdot 10^{-16} \cdot Q_F^5$$

$$T_{inj} = 490.49316 - 0.96931922 \cdot Q_F + 0.41537367 \cdot 10^{-3} \cdot Q_F^2$$
$$- 0.82875269 \cdot 10^{-7} \cdot Q_F^3 + 0.78391667 \cdot 10^{-11} \cdot Q_F^4$$
$$- 0.28461455 \cdot 10^{-15} \cdot Q_F^5 \quad .$$

<u>MR = 4</u>

$$P_{inj} = -43.986172 + 0.13726359 \cdot Q_F - 0.2581631 \cdot 10^{-4} \cdot Q_F^2$$
$$+ 0.46817281 \cdot 10^{-8} \cdot Q_F^3 - 0.2759249 \cdot 10^{-12} \cdot Q_F^4$$

$$T_{inj} = - 1096.9887 + 0.64548287 \cdot Q_F - 0.18792121 \cdot 10^{-3} \cdot Q_F^2$$
$$+ 0.23530442 \cdot 10^{-7} \cdot Q_F^3 - 0.1076003 \cdot 10^{-11} \cdot Q_F^4 \quad .$$

<u>MR = 6</u>

$$P_{inj} = - 0.88819456 + 0.10923849 \cdot Q_F - 0.66651119 \cdot 10^{-5} \cdot Q_F^2$$
$$+ 0.17194829 \cdot 10^{-8} \cdot Q_F^3 - 0.11674367 \cdot 10^{-12} \cdot Q_F^4$$

$$T_{inj} = - 555.97099 + 0.27550206 \cdot Q_F - 0.89368338 \cdot 10^{-4} \cdot Q_F^2$$
$$+ 0.12064853 \cdot 10^{-7} \cdot Q_F^3 - 0.58364974 \cdot 10^{-12} \cdot Q_F^4 \quad .$$

CONCLUSIONS

From the results of the test, it is seen that the program yields values in good agreement with the original test firing results. It is also seen that the curve fit equations yield curves in good agreement with the original plots.

REFERENCES

1. Seidel, H. H., "Thermodynamic and Heat Transfer Analysis of the J-2 Rocket Engine", Brown Engineering Company, Inc., Technical Note R-126, December, 1964

2. Barrère, M., A. Jaumotte, B. Fraeijs de Veubeke, and J. Vandenkerckhove, Rocket Propulsion, Elsevier Publishing Company, Amsterdam, London, New York, and Princeton, 1960

APPENDIX A

List of FORTRAN Symbols for Subroutines

Subroutine CONV

Subroutine TABLE1

LIST OF FORTRAN SYMBOLS FOR SUBROUTINES

C1	Hydrogen balance
C2	Oxygen balance
C3	Concentration constant - K_{p1} for specie H
C4	Concentration constant - K_{p2} for specie H_2O
C5	Concentration constant - K_{p3} for specie OH
C6	Concentration constant - K_{p4} for specie O
DNO2	Differential of XNO2
GAMMA	Ratio of PT/XNT
P	Calculated pressure (GAMMA · XNT)
PT	Pressure
T	Temperature
XNH	Number of moles of hydrogen atom
XNH2	Number of moles of hydrogen
XNH2O	Number of moles of water
XNNO	Calculated oxygen balance
XNO	Number of moles of oxygen atom
XNO2	Number of moles of oxygen
XNOH	Number of moles of hydroxyl
XNT	Total number of moles

SUBROUTINE CONV

For a given pressure and temperature this subroutine calculates the equilibrium composition of the combustion products of the J-2 engine. The method of solution, "The Errors Distributed Among the Minimum Number of Species", is outlined in <u>Rocket Propulsion</u> (see Reference 1).

The theory outlined in <u>Rocket Propulsion</u> is briefly as follows:

<u>Step 1</u>

Assume the value of a minimum number of species. Since the value of the oxidizing atoms does not exceed two, it is easier to assume the value of the oxidizing elements. Accordingly, the system can be reduced to second-degree equations. Choose the value of the ratio (GAMMA = PT/XNT) instead of the value of pressure.

<u>Step 2</u>

Solve for the number of moles. Since the oxygen balance is known, compare the calculated with the known. If they agree within the tolerance, then perform step 4. If not, perform step 3.

<u>Step 3</u>

Solve for the differential of XNO2. Algebraically apply this to XNO2 and perform step 2 again.

<u>Step 4</u>

Final answers are obtained when the calculated pressure and the known pressure agree within a specified tolerance. If this condition is not met, perform step 5.

<u>Step 5</u>

Calculate a new ration PT/XNT using your new value of N. Start again at step 2.

The chemical species in this subroutine are determined by the following equations:

$$C1 \text{ (hydrogen balance)} = 2\,XNH2O + 2\,XNH2 + XNH + XNOH \tag{A-1}$$

$$C2 \text{ (oxygen balance)} = XNH2O + XNOH + XNO + 2\,XNO2 \tag{A-2}$$

$$XNT = XNH2O + XNH2 + XNH + XNOH + XNO + XNO2 \tag{A-3}$$

$$C3 = K_{P1} = [XNH/(XNH2)^{\frac{1}{2}}] \cdot (GAMMA)^{\frac{1}{2}} \tag{A-4}$$

$$C4 = K_{P2} = 1/K_{PH2O} = \{[XNH2 \cdot (XNO2)^{\frac{1}{2}}] \,/\, XNH2O\} \cdot (GAMMA)^{\frac{1}{2}} \tag{A-5}$$

$$C5 = K_{P3} = XNOH/(XNO2 \cdot XNH2)^{\frac{1}{2}} \tag{A-6}$$

$$C6 = K_{P4} = [XNO/(XNO2)^{\frac{1}{2}}] \cdot (GAMMA)^{\frac{1}{2}} \quad . \tag{A-7}$$

The subroutine has the following calling sequence:

```
SUBROUTINE CONV (T, PT, C1, C2, XNH2O, XNO, XNO2,
                 XNH, XNOH, XNH2, XNT) .
```

T(temperature), PT(pressure), C1(hydrogen balance), and C2(oxygen balance) are input variables while the rest of the arguments are output variables.

The subroutine solves for the concentration constants C3, C4, C5, and C6 by interpolating in the concentration constants tables, using temperature as the independent variable. Then, following the theory outlined previously, the subroutine performs the following steps.

Step 1

Set GAMMA equal to 0.5 and XNO2 equal to 1.

Step 2

Solve for XNH by manipulating equations A-4, A-5, A-6, and A-7 to form the quadratic equation:

$$A = \frac{2 \cdot GAMMA}{C3^2} \cdot \left[1 + \frac{(GAMMA \cdot XNO2)^{\frac{1}{2}}}{C4}\right]$$

$$B = \frac{C5 \cdot (XNO2)^{\frac{1}{2}} \cdot GAMMA}{C3} + 1$$

$$C = -C1$$

which gives

$$XNH = \frac{-B + (B^2 - 4AC)^{\frac{1}{2}}}{2A} .$$

By substituting assigned values into the equations used to form the coefficients of the quadratic equation above, it is easy to solve for XNH2, XNH2O, and XNOH. XNNO (the calculated oxygen balance) can be solved for from Equation A-2 and XNT from Equation A-3. If XNNO and C2 agree within the specified tolerance, proceed with step 4. If not, then perform step 3.

Step 3

Solve for the differential of XNO2:

$$DNO2 = \frac{2 \cdot XNO2 \cdot (C2 - XNNO)}{XNH2) \cdot (1 - 2 \cdot AA) + XNOH \cdot (1 - AA) + 4 \cdot XNO2 + XNO}$$

where

$$AA = \frac{2 \cdot XNH2O + XNOH}{C1 + 2 \cdot (XNH2O + XNH2)}$$

Algebraically apply DNO2 to XNO2 and then return to step 2.

Step 4

Calculate pressure from the equation P = GAMMA · XNT. Compare P with the known pressure PT. If they agree within the specified tolerance, then the calculated values for XNH2O, XNO, XNO2, XNH, XNOH, XNH2, and XNT are sufficiently accurate. Control then returns to the main program. If the tolerance is not satisfied, perform step 5.

Step 5

Calculate a new GAMMA:

$$\text{GAMMA} = \frac{\text{PT}}{\text{XNT}}$$

and return to step 2.

SUBROUTINE TABLE1

This routine is used to make a linear interpolation between independent and dependent table values. The following equation was employed. The answer is a function of the X table, and K is the determining constant.

$$ANS(X) = \frac{Y_i(X) - Y_{i-1}(X)}{X_i - X_{i-1}} (K - X_{i-1}) + Y_{i-1}(X)$$

where

X - independent table,

Y - dependent table, and

K - constant .

APPENDIX B

Inputs to the Program

Output of the Program

List of FORTRAN Statements

INPUTS TO THE PROGRAM

Initial conditions are read into the computer from a series of input cards.

Card #1 --- D_t

D_t, throat diameter (cm)

Card #2 --- e_m, K_m, HNFACT

e_m, wall material thickness (cm)

K_m, thermal conductivity of material (cal/cm· sec·

HNFACT, parameter changing factor

Card #3 --- NOST, NOZ, NOTZ, NOFZ

NOST, number of stations considered

NOZ, number of stations from the end of nozzle to the throat

NOTZ, number of stations from throat to chamber

NOFZ, number of stations from manifold entrance to nozzle end .

The program requires a series of tables consisting of standard properties.

ZMII VALUES --- Molecular weight of species

ITAB --- Number of values per tables

Table #1 --- TMP Values (temperature, °K)

Table #2 --- H Values (viscosity, gm/cm sec)

Table #3 --- OH Values (viscosity, gm/cm sec)

Table #4 --- H2 Values (viscosity, gm/cm sec)

Table #5 --- H2O Values (viscosity, gm/cm sec)

Table #6 --- O1 Values (viscosity, gm/cm sec)

Table #7 --- O2 Values (viscosity, gm/cm sec)

Table #8 --- CPH Values (specific heat, cal/mole °K)

Table #9 --- CPOH Values (specific heat, cal/mole °K)

Table #10 --- CPH2 Values (specific heat, cal/mole °K)

Table #11 --- CPH20 Values (specific heat, cal/mole °K)

Table #12 --- CPO1 Values (specific heat, cal/mole °K)

Table #13 --- CPO2 Values (specific heat, cal/mole °K)

Table #14 --- ZH Values (tube parameter, in)

Table #15 --- ZN Values (tube parameter, in)

Table #16 --- X Values (nozzle coordinate, in)

Table #17 --- Z Values (nozzle coordinate, in)

Table #18 --- Temperature (°K)

IRAD --- Number values in tables

Table #19 --- MU Values (viscosity of H2 millipoise) at 1 atm

Table #20 --- MU Values (viscosity of H2 millipoise) at 10 atm

Table #21 --- MU Values (viscosity of H2 millipoise) at 20 atm

Table #22 --- MU Values (viscosity of H2, gm/cm sec) at 30 atm

Table #23 --- MU Values (viscosity of H2, gm/cm sec) at 60 atm

Table #24 --- MU Values (viscosity of H2, gm/cm sec) at 100 atm

Table #25 --- K Values (conductivity of H2, BTU/hr-ft-°R) at 1 atm

Table #26 --- K Values (conductivity of H2, BTU/hr-ft-°R) at 10 atm

Table #27 --- K Values (conductivity of H2, BTU/hr-ft-°R) at 20 atm

Table #28 --- K Values (conductivity of H2, BTU/hr-ft-°R) at 30 atm

Table #29 --- K Values (conductivity of H2, BTU/hr-ft-°R) at 60 atm

Table #30 --- K Values (conductivity of H2, BTU/hr-ft-°R) at 100 atm

Table #31 --- Cp Values (specific heat of H2, cal/gm-°K) at 1 atm

Table #32 --- Cp Values (specific heat of H2, cal/gm-°K) at 10 atm

Table #33 --- Cp Values (specific heat of H2, cal/gm-°K) at 20 atm

Table #34 --- Cp Values (specific heat of H2, cal/gm-°K) at 30 atm

Table #35 --- Cp Values (specific heat of H2, cal/gm-°K) at 60 atm

Table #36 --- Cp Values (specific heat of H2, cal/gm-°K) at 100 atm

Table #37 --- Gamma Values (specific wt. of H2, gm/cm^3) at 1 atm

Table #38 --- Gamma Values (specific wt. of H2, gm/cm^3) at 10 atm

Table #39 --- Gamma Values (specific wt. of H2, gm/cm^3) at 20 atm

Table #40 --- Gamma Values (specific wt. of H2, lb/ft^3) at 30 atm

Table #41 --- Gamma Values (specific wt. of H2, lb/ft^3) at 60 atm

Table #42 --- Gamma Values (specific wt. of H2, lb/ft^3) at 100 atm

NOVAL --- Number of values in the following tables

Table #43 --- TEMP Values (temperature, °K)

Table #44 --- H298H Values (enthalpy, cal/mole)

Table #45 --- H298OH Values (enthalpy, cal/mole)

Table #46 --- H298H2 Values (enthalpy, cal/mole)

Table #47 --- H298H2O Values (enthalpy, cal/mole)

Table #48 --- H298O Values (enthalpy, cal/mole)

Table #49 --- H298O2 Values (enthalpy, cal/mole)

Table #50 --- CPTH Values (specific heat, cal/mole -°K)

Table #51 --- CPTOH Values (specific heat, cal/mole -°K)

Table #52 --- CPTH2 Values (specific heat, cal/mole -°K)

Table #53 --- CPTH2O Values (specific heat, cal/mole - °K)

Table #54 --- CPTO Values (specific heat, cal/mole -°K)

Table #55 --- CPTO2 Values (specific heat, cal/mole -°K)

Table #56 --- SH Values (entropy, cal/mole -°K)

Table #57 --- SOH Values (entropy, cal/mole -°K)

Table #58 --- SH2 Values (entropy, cal/mole -°K)

Table #59 --- SH2O Values (entropy, cal/mole -°K)

Table #60 --- SO Values (entropy, cal/mole -°K)

Table #61 --- SO2 Values (entropy, cal/mole -°K)

Table #62 --- KPH Values (concentration constant of H)

Table #63 --- KPOH Values (concentration constant of OH)

Table #64 --- KPH2O Values (concentration constant of H2O)

Table #65 --- KPO Values (concentration constant of O)

NOAEAT, IAREA --- The number of area ratios to be read in and the number from the initial start to the throat.

Table #66 --- AEATI (area ratios)

IMR --- The number of oxidizer and fuel cards to follow

WDOTO, WDOTF --- The oxidizer and fuel flow card (gal/min)

OUTPUT OF THE PROGRAM

The quantities on the sample printout are defined as follows:

$\dot{\omega}_o$	-	Oxidizer flow (gal/min)
$\dot{\omega}_f$	-	Fuel flow (gal/min)
MR	-	Mixture ratio
P_c	-	Chamber pressure (psig)
T_c	-	Chamber temperature (°F)
P_{inj}	-	Main fuel injector pressure (psig)
T_{inj}	-	Main fuel injector temperature (°F)

Each case at one mixture and one fuel flow rate ran approximately one hour.

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
      DIMENSION  PI(100), TGI(100), ZMI(100), CPI(100),
     1 AEATI(100),ZISPI(100), HT(100),OHT(100), H2T(100), H2OT(100),
     2 O1T(100), O2T(100), TMP(100), H(45), OH(45), H2(45),
     3  H2O(45), O1(45), O2(45), ZMII(6),          CPII(40),
     4                                        ZKLC(2), CPLC(2),
     5   ZMULC(2) , A(100), TWGI(100)     ,
     6 ZMU(100), ZK(100),  TF(100),          HCAL(100), TWL(2),
     7 HLCAL(100), CPH(45), CPOH(45), CPH2(45), CPH2O(45), CPO1(45),
     8 CPO2(45), ZN(100), ZH(100), AL(100), DL(100), X(100), Z(100)
      DIMENSION ZKL30(100), CPL30(100), ZMUL30(100), TMPL(100), ZKL60
     1 (100), CPL60(100), ZMUL60(100), ZKL99(100), CPL99(100),
     2 ZMUL99(100), GAM30(100), GAM60(100), GAM99(100)
     3, ZMUL1(100), ZMUL10(100), ZMUL20(100), ZKL1(100), ZKL10(100),
     4 ZKL20(100), CPL1(100), CPL20(100), GAM1(100),      CPL10(100),
     5 GAM10(100), GAM20(100)
      DIMENSION H298H(100),H298OH(100),H298H2(100),H29H2O(100),
     1 H298O(100),H298O2(100),          CPTH(100),CPTOH(100),CPTH2(100),
     2 CPTH2O(100),CPTO(100),CPTO2(100),SH(100),SOH(100),SH2(100),
     3 SH2O(100),SO(100),SO2(100)
      DIMENSION CPIS(100), TGIS(100)
      DIMENSION XAXIS(50), YAXIST(50), YAXISP(50)
      COMMON /TABLE/ TEMP(100),ZKPH(100),ZKPH2O(100),ZKPOH(100),
     1 ZKPO(100)
C
C     READ  NO OF STATIONS MAXIMUM 100  *     *     *     *     *     *     **
```

```
C
      READ  (5, 1612)  DT
      READ  (5,1612)  EM, ZKM, HNFACT
 1612 FORMAT  (4E15.8)
      READ (5,1615) NOST, NOZ   , NOTZ, NOFZ
 1615 FORMAT (4I3)
 1610 FORMAT (I3)
C
C     READ INPUT FROM THERMOCHEMICAL PROGRAM      *    *    *    *    **
C
 1967 FORMAT (8F10.0)
 1620 FORMAT (5E15.8)
C     READ SPECIAL MOLE FRACTION   *    *    *    *    *    *    *    **
      READ (5,1620) (ZMII(I), I=1,6)
C     READ NO OF INPUTS FOR TABLES      *    *    *    *    *    *    **
      READ (5,1610)  ITAB
C
C     READ TABLE VALUES FOR COMPOUNDS   *    *    *    *    *    *    **
C
      READ (5,1620) (TMP(I),I=1, ITAB)
      READ (5,1620) (H(I), I=1, ITAB)
      READ (5,1620) (OH(I), I=1, ITAB)
      READ (5,1620) (H2(I),I=1, ITAB)
      READ (5,1620) (H2O(I), I=1, ITAB)
      READ (5,1620) (O1(I), I=1, ITAB)
```

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
      READ (5,1620)( O2(I), I=1, ITAB)
C
C     SPECIAL TABLE FOR CHAMBER PRESSURE CALCULATION (CPII) *   *   **
C
      READ (5,1620) (CPH(I), I=1,ITAB)
      READ (5,1620) (CPOH(I), I=1,ITAB)
      READ (5,1620) (CPH2(I), I=1,ITAB)
      READ (5,1620) (CPH2O(I), I=1, ITAB)
      READ (5,1620) (CPO1(I), I=1, ITAB)
      READ (5,1620) (CPO2(I), I=1, ITAB)
      DO 7677  I=1, ITAB
      CPH(I)= CPH(I) * 1.98726
      CPOH(I)= CPOH(I) * 1.98726
      CPH2(I)=  CPH2(I) * 1.98726
      CPH2O(I)= CPH2O(I) * 1.98726
      CPO1(I)= CPO1(I) * 1.98726
      CPO2(I)= CPO2(I) * 1.98726
 7677 CONTINUE
C
C     READ N AND H VALUES FOR CALCULATION AND INTERPOLATION *   *   **
C
      READ (5,1620) (ZH(I), I= 1, NOST)
      READ (5,1620) (ZN(I), I= 1, NOST)
      READ (5,1620) (X(I), I= 1, NOST)
      READ (5,1620) (Z(I), I= 1, NOST)
```

```
      DO 6765 I= 1, NOST
      ZH(I)= ZH(I) - HNFACT * 2.
      ZN(I)= ZN(I) - HNFACT
 6765 CONTINUE
      DO 356 I= 1, NOST
      ZH(I)= ZH(I) * 2.54
      ZN(I)= ZN(I) * 2.54
      X(I)= X(I) * 2.54
      Z(I)= Z(I) * 2.54
  356 CONTINUE
C
C     READ A TABLE FOR KL, CPL, MUL, TMPL    *    *    *    *    *    **
C
      READ (5,1610) IRAD
      READ (5,1620) (TMPL(I), I= 1, IRAD)
      DO 3035 I= 1, IRAD
      TMPL(I)= TMPL(I) * .5555555555
 3035 CONTINUE
      READ (5, 1967) (ZMUL1(I), I= 1, IRAD)
      READ (5, 1967) (ZMUL10(I), I=1, IRAD)
      READ (5, 1967) (ZMUL20(I), I= 1, IRAD)
      READ (5,1620) (ZMUL30(I), I= 1, IRAD)
      READ (5,1620) (ZMUL60(I), I= 1, IRAD)
      READ (5,1620) (ZMUL99(I), I= 1, IRAD)
      READ (5, 1967) (ZKL1(I), I= 1, IRAD)
```

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
READ (5, 1967) (ZKL10(I), I= 1, IRAD)
READ (5, 1967) (ZKL20(I), I= 1, IRAD)
READ (5,1620) (ZKL30(I), I= 1, IRAD)
READ (5,1620) (ZKL60(I), I= 1, IRAD)
READ (5,1620) (ZKL99(I), I= 1, IRAD)
READ (5, 1967) (CPL1(I), I= 1, IRAD)
READ (5, 1967)  (CPL10(I), I= 1, IRAD)
READ (5, 1967)  (CPL20(I), I= 1, IRAD)
READ (5,1620) (CPL30(I), I= 1, IRAD)
READ (5,1620) (CPL60(I), I= 1, IRAD)
READ (5,1620) (CPL99(I), I= 1, IRAD)
READ (5, 1967) (GAM1(I), I= 1, IRAD)
READ (5, 1967)  (GAM10(I), I= 1, IRAD)
READ (5, 1967)  (GAM20(I), I= 1, IRAD)
READ (5,1620) (GAM30(I), I= 1, IRAD)
READ (5,1620) (GAM60(I), I= 1, IRAD)
READ (5,1620)          (GAM99(I), I= 1, IRAD)
DO 5901 I= 1, IRAD
ZKL1(I)= ZKL1(I) * .004133
ZKL10(I)= ZKL10(I) * .004133
ZKL20(I)= ZKL20(I) * .004133
ZKL30(I)= ZKL30(I) * .004133
ZKL60(I)= ZKL60(I) * .004133
ZKL99(I)= ZKL99(I) * .004133
GAM30(I)= GAM30(I) * .01601
```

```
      GAM60(I)= GAM60(I) * .01601
      GAM99(I)= GAM99(I) * .01601
      ZMUL1(I)= ZMUL1(I) * 1.E-03
      ZMUL10(I)= ZMUL10(I) * 1.E-03
      ZMUL20(I)= ZMUL20(I) * 1.E-03
 5901 CONTINUE
 3181 WRITE (6,9161) DT
 9161 FORMAT (1H-, 10X, 9HDT (CM) =E15.8)
      WRITE (6, 9009)            EM
 9009 FORMAT (1H-
     1 10X,  9HEM (CM) =E15.8)
      WRITE (6,1636)      ZKM, HNFACT
 1636 FORMAT (1H-, 10X, 24HZKM (CAL/CM.SEC.DEG K) =E15.8, 10X,
     18HHNFACT =E15.8)
      WRITE (6,1639) NOST, NOZ    , NOTZ, NOFZ
 1639 FORMAT (1H-, 10X, 6HNOST =I5,    10X, 5HNOZ =I5,
     1 10X, 6HNOTZ =I5, 10X, 6HNOFZ =I5)
      WRITE (6,5673)
 5673 FORMAT (1H-, 10X, 27HTEMP VALUES (TEMP., DEG. K))
      WRITE (6,2122) TMP
      WRITE (6,3175)
 3175 FORMAT (1H-, 10X, 26HH VALUES (VISC., G/CM.SEC))
      WRITE (6,2122) H
      WRITE (6,2767)
 2767 FORMAT (1H-, 10X, 27HOH VALUES (VISC., G/CM.SEC))
```

```
      WRITE (6,2122) OH
      WRITE (6,2737)
 2737 FORMAT (1H-, 10X, 27HH2 VALUES (VISC., G/CM.SEC))
      WRITE (6,2122) H2
      WRITE (6,2131)
 2131 FORMAT (1H-, 10X, 28HH20 VALUES (VISC., G/CM.SEC))
      WRITE (6,2122) H20
      WRITE (6,2673)
 2673 FORMAT (1H-, 10X, 27HO1 VALUES (VISC., G/CM.SEC))
      WRITE (6,2122) O1
      WRITE (6,2637)
 2637 FORMAT (1H-, 10X, 27HO2 VALUES (VISC., G/CM.SEC))
      WRITE (6,2122) O2
      WRITE (6,1787)
 1787 FORMAT (1H-, 10X, 36HZMII VALUES (MOL. WT. OF COMPONENTS))
      WRITE (6,2122) ZMII
      WRITE (6,1135)
 1135 FORMAT (1H-, 10X, 38HCPH VALUES (SPEC. HEAT, CAL/MOLE DEG K))
      WRITE (6,2122) CPH
      WRITE (6,1137)
 1137 FORMAT (1H-, 10X, 39HCPOH VALUES (SPEC. HEAT, CAL/MOLE DEG K))
      WRITE (6,2122) CPOH
      WRITE (6,1139)
 1139 FORMAT (1H-, 10X, 39HCPH2 VALUES (SPEC. HEAT, CAL/MOLE DEG K))
      WRITE (6,2122) CPH2
```

```
      WRITE (6, 1141)
1141 FORMAT (1H-, 10X, 40HCPH2O VALUES (SPEC. HEAT, CAL/MOLE DEG K))
      WRITE (6,2122) CPH2O
      WRITE (6, 1143)
1143 FORMAT (1H-, 10X, 39HCPO1 VALUES (SPEC. HEAT, CAL/MOLE DEG K))
      WRITE (6,2122) CPO1
      WRITE (6, 1145)
1145 FORMAT (1H-, 10X, 39HCPO2 VALUES (SPEC. HEAT, CAL/MOLE DEG K))
      WRITE (6,2122) CPO2
      WRITE (6,5761)
5761 FORMAT (1H-, 10X, 25HZH VALUES (TUBE PAR., CM))
      WRITE (6,2122) ZH
      WRITE (6,5762)
5762 FORMAT (1H-, 10X, 25HZN VALUES (TUBE PAR., CM))
      WRITE (6,2122) ZN
      WRITE (6,5763)
5763 FORMAT (1H-, 10X, 28HX VALUES (NOZZLE COORD., CM))
      WRITE (6,2122) X
      WRITE (6,5764)
5764 FORMAT (1H-, 10X, 28HZ VALUES (NOZZLE COORD., CM))
      WRITE (6,2122) Z
      WRITE (6,5501)
5501 FORMAT (1H-, 10X, 25HTMPL VALUES (TMP. DEG. K))
      WRITE (6,2122) TMPL
      WRITE (6, 1076)
```

```
1076 FORMAT (1H-, 10X, 39HMU VALUES (VISC. OF H, G/CM.S) AT 1 ATM)
     WRITE (6, 2122) ZMUL1
     WRITE (6, 1077)
1077 FORMAT (1H-, 10X, 39HMU VALUES (VISC. OF H, G/CM.S) AT 10ATM)
     WRITE (6, 2122) ZMUL10
     WRITE (6, 1078)
1078 FORMAT (1H-, 10X, 40HMU VALUES (VISC. OF H, G/CM.S) AT 20 ATM)
     WRITE (6, 2122) ZMUL20
     WRITE (6,5502)
5502 FORMAT (1H-, 10X, 42HMU VALUES AT 30 ATM (VISC. OF H, G/CM.SEC))
     WRITE (6,2122) ZMUL30
     WRITE (6,5503)
5503 FORMAT (1H-,10X, 42HMU VALUES AT 60 ATM (VISC. OF H, G/CM.SEC))
     WRITE (6,2122) ZMUL60
     WRITE (6,5504)
5504 FORMAT (1H-, 10X, 43HMU VALUES AT 100 ATM (VISC. OF H, G/CM.SEC))
     WRITE (6,2122) ZMUL99
     WRITE (6, 1081)
1081 FORMAT (1H-, 10X,45HK VALUES (COND. OF H, CAL/CM.S.DEG K)AT 1 ATM)
     WRITE (6, 2122)    ZKL1
     WRITE (6, 1082)
1082 FORMAT(1H-,10X,46HK VALUES (COND. OF H, CAL/CM.S.DEG K)AT 10 ATM)
     WRITE (6, 2122) ZKL10
     WRITE (6, 1083)
1083 FORMAT(1H-,10X,47HK VALUES (COND. OF H, CAL/CM.S.DEG K) AT 20 ATM)
```

```
      WRITE (6, 2122) ZKL20
      WRITE (6,5505)
 5505 FORMAT (1H-, 10X, 47HK VALUES AT 30 ATM (COND. OF H, CAL/CM.S.DEG
     1K))
      WRITE (6,2122) ZKL30
      WRITE (6,5506)
 5506 FORMAT (1H-,10X,47HK VALUES AT 60 ATM (COND. OF H, CAL/CM.S.DEG K)
     1)
      WRITE (6,2122) ZKL60
      WRITE (6,5507)
 5507 FORMAT(1H-,10X,47HK VALUES AT 100 ATM (COND OF H, CAL/CM.S.DEG K))
      WRITE (6,2122) ZKL99
      WRITE (6, 1091)
 1091 FORMAT(1H-,10X,49HCP VALUES (SPEC. HEAT OF H, CAL/G.DEG K) AT 1 AT
     1M)
      WRITE (6, 2122) CPL1
      WRITE (6, 1092)
 1092 FORMAT(1H-,10X,50HCP VALUES (SPEC. HEAT OF H, CAL/G.DEG K) AT 10 A
     1TM)
      WRITE (6, 2122) CPL10
      WRITE (6, 1093)
 1093 FORMAT(1H-,10X,50HCP VALUES (SPEC. HEAT OF H, CAL/G.DEG K) AT 20 A
     1TM)
      WRITE (6, 2122) CPL20
      WRITE (6,5508)
```

```
5508 FORMAT (1H-,10X,50HCP VALUES AT 30 ATM (SPEC HEAT OF H, CAL/G. DEG
    1 K))
     WRITE (6,2122) CPL30
     WRITE (6,5509)
5509 FORMAT (1H-, 10X, 49HCP VALUES AT 60 ATM (SPEC HEAT OF H, CAL/G.DE
    1G K))
     WRITE (6,2122) CPL60
     WRITE (6,5510)
5510 FORMAT (1H-, 10X, 50HCP VALUES AT 100 ATM (SPEC HEAT OF H, CAL/G.D
    1EG K))
     WRITE (6,2122) CPL99
     WRITE (6, 2021)
2021 FORMAT(1H-,10X,45HGAMMA VALUES (SPEC. WT. OF H, G/CM3) AT 1 ATM)
     WRITE (6, 2122) GAM1
     WRITE (6, 2022)
2022 FORMAT(1H-,10X,45HGAMMA VALUES (SPEC. WT. OF H, G/CM3) AT 10ATM)
     WRITE (6, 2122) GAM10
     WRITE (6, 2023)
2023 FORMAT(1H-,10X,46HGAMMA VALUES (SPEC. WT. OF H, G/CM3) AT 20 ATM)
     WRITE (6, 2122)GAM20
      WRITE (6,5511)
5511 FORMAT(1H-10X,45HGAMMA VALUES AT 30 ATM (SPEC.WT. OF H, G/CM3))
     WRITE (6, 2122) GAM30
     WRITE (6, 5512)
2122 FORMAT (1H , 10X, 6E15.8)
```

```
 5512 FORMAT(1H-,10X,46HGAMMA VALUES AT 60 ATM (SPEC. WT. OF H, G/CM3))
      WRITE (6, 2122) GAM60
      WRITE (6, 5514)
 5514 FORMAT (1H-,10X,44HGAMMA VALUES AT 100 ATM(SPEC WT OF H, G/CM3))
      WRITE (6, 2122) GAM99
C
C     READ  TABLES  FOR INTERP.    *    *    *    *    *    *    *    **
      READ (5,101) NOVAL
  101 FORMAT ( I3)
      READ (5,102)  (TEMP(I), I= 1, NOVAL)
      READ (5, 102) (H298H(I), I=1, NOVAL)
      READ (5, 102) ( H298OH(I), I=1, NOVAL)
      READ (5, 102) ( H298H2(I), I=1, NOVAL)
      READ (5, 102) ( H29H2O(I), I=1, NOVAL)
      READ (5, 102) ( H298O(I), I=1, NOVAL)
      READ (5,102) ( H29802(I), I=1, NOVAL)
      READ (5, 102) (CPTH(I), I=1, NOVAL)
      READ (5, 102) (CPTOH(I), I=1, NOVAL)
      READ (5, 102) (CPTH2(I), I=1, NOVAL)
      READ (5, 102) (CPTH2O(I), I=1, NOVAL)
      READ (5, 102) (CPTO(I), I=1, NOVAL)
      READ (5, 102) (CPTO2(I), I=1, NOVAL)
      READ (5, 102) (SH(I), I=1, NOVAL)
      READ (5, 102) (SOH(I), I=1, NOVAL)
      READ (5, 102) ( SH2(I), I=1, NOVAL)
```

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
      READ (5, 102) (SH2O(I), I=1, NOVAL)
      READ (5, 102) ( SO(I), I=1, NOVAL)
      READ (5, 102) ( SO2(I), I=1, NOVAL)
      READ (5, 102) ( ZKPH(I), I=1, NOVAL)
      READ (5, 102) ( ZKPOH(I), I=1, NOVAL)
      READ (5, 102) ( ZKPH2O(I), I=1, NOVAL)
      READ (5, 102) (ZKPO(I), I=1, NOVAL)
  102 FORMAT (8F10.0)
      WRITE (6, 123)
  123 FORMAT (1H-)
      WRITE (6,201)
  201 FORMAT (1H-,10X, 20HTEMP VALUES (DEG. K))
      WRITE (6,202) TEMP
  202 FORMAT (1H , 10X, 6E15.8)
      WRITE (6, 203)
  203 FORMAT (1H-, 10X, 29HH298H VALUES (CAL/MOLE/DEG K))
      WRITE (6,202) H298H
      WRITE (6, 204)
  204 FORMAT (1H-, 10X, 30HH298OH VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) H298OH
      WRITE (6, 205)
  205 FORMAT (1H-, 10X, 30HH298H2 VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) H298H2
      WRITE (6, 206)
  206 FORMAT (1H-, 10X, 31HH298H2O VALUES (CAL/MOLE/DEG K))
```

```
      WRITE (6, 202) H29H2O
      WRITE (6, 207)
  207 FORMAT (1H-, 10X, 29HH298O VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) H298O
      WRITE (6, 208)
  208 FORMAT (1H-,10X, 30HH298O2 VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) H298O2
      WRITE (6, 209)
  209 FORMAT (1H-,10X, 28HCPTH VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) CPTH
      WRITE (6, 210)
  210 FORMAT (1H-, 10X, 29HCPTOH VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) CPTOH
      WRITE (6, 211)
  211 FORMAT (1H-, 10X, 29HCPTH2 VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) CPTH2
      WRITE (6, 212)
  212 FORMAT (1H-, 10X, 30HCPTH2O VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) CPTH2O
      WRITE (6, 213)
  213 FORMAT (1H-, 10X, 28HCPTO VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) CPTO
      WRITE (6, 214)
  214 FORMAT (1H-, 10X  29HCPTO2 VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) CPTO2
```

```
      WRITE (6, 215)
  215 FORMAT (1H-, 10X, 26HSH VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) SH
      WRITE (6, 216)
  216 FORMAT (1H-, 10X, 27HSOH VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) SOH
      WRITE (6, 217)
  217 FORMAT (1H-, 10X, 27HSH2 VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) SH2
      WRITE (6, 218)
  218 FORMAT (1H-, 10X,28HSH2O VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) SH2O
      WRITE (6, 219)
  219 FORMAT (1H-, 10X, 26HSO VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) SO
      WRITE (6, 2026)
 2026 FORMAT (1H-, 10X, 27HSO2 VALUES (CAL/MOLE/DEG K))
      WRITE (6, 202) SO2
      WRITE (6, 221)
  221 FORMAT (1H-, 10X,11HZKPH VALUES)
      WRITE (6, 202) ZKPH
      WRITE (6,2027)
 2027 FORMAT (1H-, 10X, 12HZKPOH VALUES)
      WRITE (6,202) ZKPOH
      WRITE (6,223)
```

```
  223 FORMAT (1H-, 10X, 13HZKPH2O VALUES)
      WRITE (6,202) ZKPH2O
      WRITE (6, 224)
  224 FORMAT (1H-, 10X,11HZKPO VALUES)
      WRITE (6,202) ZKPO
      READ (5, 935) NOAEAT, IAREA
  935 FORMAT (2I3)
      READ (5, 102) (AEATI(I),I=1,NOST)
C
C     SET UP CONSTANTS  .  .  .  *  *  *  .  .  *  **
C
      DELHFO=  -3.08
      DELHFH=  -1.887
      ZMWO=  32.0
      ZMWH=  2.016
      DENO=  1.14
      DENH=  .071
C
C     READ  MIXTURE RATIOS   *  *  *  *  *  *  .  *  **
C
      READ (5, 70001) IMR
70001 FORMAT (I3)
      DO 70002 IKM= 1, IMR
      READ (5, 70003) WDOTO, WDOTF
70003 FORMAT (2E15.8)
```

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
      ZMR= WDOTO * .0717536 / (WDOTF * .0044466)
      WRITE (6, 70004) WDOTO, WDOTF
70004 FORMAT (1H1, 10X, 25HOXIDIZER FLOW (GAL/MIN) =E15.8, 10X,
     1 21HFUEL FLOW (GAL/MIN) =E15.8)
      WRITE (6, 70005)      ZMR
70005 FORMAT (1H-,
     1 10X, 15HMIXTURE RATIO =E15.8)
      WDOTO= .0717536 * WDOTO
      WDOTF= .0044466 * WDOTF
      C1=  (200./2.016)  *   (WDOTF/ (WDOTO  + WDOTF))
      C2=  (200./ 32.)  *  (WDOTO / (WDOTO  + WDOTF))
C
C     A CONSTANT VAR. FOR  WO/(WO + WF) AND  WF/(WO + WF)        *     **
C
      WOOWF=   WDOTO / (WDOTO  +  WDOTF)
      WFOWO=   WDOTF/ ( WDOTO  +  WDOTF)
C
C     BEGIN CALCULATIONS     *    *    *    *    *    *    *    *     **
C
  501 WTO=  100. *  WOOWF
  502 WTH=  100. *  WFOWO
  503 ZMOLEQ=  (100./ 32.) *   WOOWF
  504 ZMOLEH= (100. /  2.016)  *  WFOWO
  505 DENP=  1.14 *  WOOWF +  .071 * WFOWO
  506 DELHF= -3.08 * (100./ 32.) *  WOOWF - 1.887 * (100./ 2.016) *
```

```
    1  WFOWO
 507 HYG=200./ 2.016  *  WFOWO
 508 OXY=  200./ 32. *   WOOWF
C
C      INITIAL GUESS FOR TC, M, GAM, PC
C
 509 TC=  298.0
 510 ZM=  12.0
     DECTC= 1000.
 511 GAMMA= 1.15
 512 PC=  (WDOTF *   (WDOTO /  WDOTF  + 1.) * ((847.9*  TC) / (ZM *
    1  9.81 ))  ** .5) / (1094.947   *  GAMMA ** .5  *  (2./(GAMMA+1.)
    2 )  **  ((  GAMMA + 1.) / ( 2. * (GAMMA - 1.))))
 513 CALL CONV(TC,PC,C1, C2, ZNH2O, ZNO, ZNO2, ZNH, ZNOH, ZNH2, ZNT)
C
C      INTERPOLATE FOR H298 FOR ALL SPECIES AT TC  *   *   *   *   **
C
 901 HCH= TABLE1 (TEMP(1), H298H(1), TC, NOVAL)
 902 HCOH= TABLE1 ( TEMP(1), H298OH(1), TC, NOVAL)
 903 HCH2= TABLE1 (TEMP(1), H298H2(1), TC, NOVAL)
 904 HCH2O= TABLE1 (TEMP(1), H29 H2O(1), TC, NOVAL)
 905 HCO= TABLE1 (TEMP(1), H298O(1), TC, NOVAL)
 906 HCO2= TABLE1 (TEMP(1), H298O2(1), TC, NOVAL)
 907 TERM1=      -(ZNH2O *  HCH2O +   ZNO *  HCO  +  ZNO2 *  HCO2  +
    1   ZNH *  HCH  +  ZNOH * HCOH  +   ZNH2 *  HCH2)
```

```
908 TERM2=  (-ZNH2O *  57.798 +  ZNH *  52.089 +  ZNOH * 9.33 +  ZNO
   1 *  59.55)  +  3.08 *  (100./ 32.) *    WOOWF  +  1.887  *
   2  (100. / 2.016)  *  WFOWO
    IF   (ABS (TERM1 - TERM2) .LE..01  )  GO  TO  915
    IF  (ABS (TERM2)-   ABS (TERM1))  909, 915, 911
909 TC= TC  - DECTC
    DECTC=  DECTC  * .1
911 TC=  TC +  DECTC
    IF (DECTC .LE. .0001) GO TO 915
    GO TO  512
915 ZM=  100./ ZNT
916 CPCH= TABLE1 (TEMP(1), CPTH(1), TC, NOVAL)
917 CPCOH= TABLE1 (TEMP(1), CPTOH(1), TC, NOVAL)
918 CPCH2= TABLE1 (TEMP(1), CPTH2(1), TC, NOVAL)
919 CPCH2O= TABLE1 (TEMP(1), CPTH2O(1), TC, NOVAL)
920 CPCO= TABLE1 (TEMP(1), CPTO(1), TC, NOVAL)
921 CPCO2= TABLE1 (TEMP(1), CPTO2(1), TC, NOVAL)
922 GAMMA= ( ZNH2O * CPCH2O  + ZNH2 *  CPCH2 +  ZNO * CPCO +  ZNO2 *
   1  CPCO2 +   ZNH *  CPCH  + ZNOH *  CPCOH ) /  (ZNH2O *  CPCH2O  +
   2  ZNH2 *  CPCH2 +  ZNO *  CPCO +  ZNO2 * CPCO2  + ZNH  *  CPCH  +
   3  ZNOH *  CPCOH  -  ZNT  *  1.987)
923 PCPRES=  (WDOTF  *   (WDOTO / WDOTF  + 1. ) *  ((847.9 * TC) /
   1    (ZM * 9.81)) * * .5) /   ( 1094.947 *  GAMMA ** .5  *(2. /
   2  (GAMMA + 1.)) **  ((GAMMA + 1.)/ (2. *  (GAMMA - 1.))))
    IF  (ABS (PCPRES - PC) .LE..01  ) GO TO 924
```

```
      PC=  PCPRES
      GO TO   513
C
C     INTERPOLATE FOR (S) FOR ALL SPECIES    *    *    *    *    *)    **
C
  924 SCH= TABLE1 (TEMP(1), SH(1), TC, NOVAL)
  925 SCOH= TABLE1 (TEMP(1), SOH(1), TC, NOVAL)
  926 SCH2= TABLE1 (TEMP(1), SH2(1), TC, NOVAL)
  927 SCH2O= TABLE1 (TEMP(1), SH2O(1), TC, NOVAL)
  928 SCO= TABLE1 (TEMP(1), SO(1), TC, NOVAL)
  929 SCO2= TABLE1 (TEMP(1), SO2(1), TC, NOVAL)
      IF (ZNH .EQ. 0.) GO TO 931
      ZLOGH= ALOG(ZNH/ZNT)
      GO TO 932
  931 ZLOGH= 0.
  932 IF (ZNOH .EQ. 0.) GO TO 933
      ZLOGOH=ALOG (ZNOH/ZNT)
      GO TO 934
  933 ZLOGOH= 0.
  934 IF (ZNH2O .EQ. 0.) GO TO 949
      ZLOGHO= ALOG (ZNH2O/ ZNT)
      GO TO 942
  949 ZLOGHO= 0.
  942 IF(ZNH2.EQ. 0.) GO TO 943
      ZLOGH2= ALOG (ZNH2 / ZNT)
```

```
      GO TO 944
  943 ZLOGH2= 0.
  944 IF (ZNO .EQ. 0.) GO TO 945
      ZLOGO= ALOG (ZNO/ ZNT)
      GO TO 946
  945 ZLOGO= 0.
  946 IF (ZNO2 .EQ. 0.) GO TO 947
      ZLOGO2= ALOG( ZNO2 / ZNT)
      GO TO 930
  947 ZLOGO2= 0.
  930 SC=  (ZNH2O *  SCH2O  +  ZNH2 *  SCH2  +  ZNH * SCH +  ZNOH * SCOH
     1   +  ZNO *  SCO +  ZNO2 *  SCO2) -  ( 1.987 * ZNT * (  ZNH2O /
     2  ZNT *  ZLOGHO                 + ZNH2 / ZNT *   ZLOGH2
     3 +    ZNH  / ZNT  * ZLOGH              +   ZNOH / ZNT  *
     4   ZLOGOH          +   ZNO / ZNT  * ZLOGO              +  ZNO2 / ZNT
     5   *  ZLOGO2                +  ALOG (PC)))
      ZNH2OC= ZNH2O
      ZNOC= ZNO
      ZNO2C= ZNO2
      ZNHC= ZNH
      ZNOHC= ZNOH
      ZNH2C= ZNH2
C
C     CALCULATION OF EXIT CONDITIONS  *         *    *    *    *     ***
C
```

```
      DO 936 I= 1, NOAEAT
      GAMMAE  = GAMMA
  961 DECPCP= 100.
      IF  (I -IAREA)1061, 1061, 1062
 1061 PCP   = 2.
      GO TO 941
 1062 PCP= 1.01
      DECPCP=.01
  941 TERM1= (2. * GAMMAE / (GAMMAE - 1.) * (1. - (1. / PCP) **
     1 (( GAMMAE - 1.) / GAMMAE))) ** .5
      TERM2= GAMMAE ** .5 * (2. / (GAMMAE + 1.)) ** (( GAMMAE + 1.)
     1 / (2. * (GAMMAE - 1.))) / AEATI(I) * PCP ** (1. / GAMMAE)
      IF (ABS (TERM1 - TERM2) .LE..0001)GO TO 940
      IF(I.LE.IAREA)GOTO 6163
      IF(ABS(TERM2)-ABS(TERM1))938,940,939
 6163 IF (ABS(TERM1) - ABS (TERM2)) 938, 940, 939
  938 PCP= PCP - DECPCP
      DECPCP= DECPCP * .1
  939 PCP= PCP + DECPCP
      GO TO 941
  940 PI(I)= 1. / PCP * PC
      TGI(I)= 100.
      DECTE= 1000.
 1058 CALLCONV (TGI(I), PI(I), C1, C2, ZNH2O, ZNO, ZNO2, ZNH, ZNOH,
     1 ZNH2, ZNT)
```

```
      SCH= TABLE1( TEMP(1), SH(1), TGI(I), NOVAL)
      SCOH= TABLE1 (TEMP(1), SOH(1), TGI(I), NOVAL)
      SCH2= TABLE1 (TEMP(1), SH2(1), TGI(I), NOVAL)
      SCH2O= TABLE1 (TEMP(1), SH2O(1), TGI(I), NOVAL)
      SCO= TABLE1 (TEMP(1), SO(1), TGI(I), NOVAL)
      SCO2= TABLE1 (TEMP(1), SO2(1), TGI(I), NOVAL)
      IF (ZNH .EQ. 0.) GO TO 1041
      ZLOGH=  ALOG (ZNH / ZNT)
      GO TO 1042
 1041 ZLOGH=  0.
 1042 IF  (ZNOH .EQ.0.) GO TO 2028
      ZLOGOH=  ALOG (ZNOH / ZNT)
      GO TO 2039
 2028 ZLOGOH=  0.
 2039 IF  (ZNH2O  .EQ. 0.) GO TO 2029
      ZLOGHO=  ALOG (ZNH2O / ZNT)
      GO TO 1046
 2029 ZLOGHO=  0.
 1046 IF (ZNH2 .EQ. 0.) GO TO 1047
      ZLOGH2= ALOG (ZNH2 / ZNT)
      GO TO 1048
 1047 ZLOGH2=  0.
 1048 IF  (ZNO .EQ.  0.) GO TO 1049
      ZLOGO=   ALOG (ZNO/ ZNT)
      GO TO 2024
```

```
1049 ZLOGO=  0.
2024 IF (ZNO2 .EQ. 0.) GO TO 1051
     ZLOGO2= ALOG (ZNO2 / ZNT)
     GO TO 1052
1051 ZLOGO2= 0.
1052 SE=      ZNH2O *  SCH2O + ZNH2 *  SCH2  +   ZNH *  SCH  +  ZNOH *
    1  SCOH  +  ZNO *  SCO  +  ZNO2 * SCO2  -  (1.987 * ZNT * ( ZNH2O
    2  /  ZNT  *  ZLOGHO  + ZNH2 / ZNT * ZLOGH2  +  ZNH / ZNT * ZLOGH
    3  +  ZNOH /  ZNT * ZLOGOH  + ZNO / ZNT *  ZLOGO  +  ZNO2 / ZNT *
    4  ZLOGO2  +  ALOG (PI(I))))
     IF (ABS(SC-  SE)  .LE..0001)  GO TO  1055
     IF  (SC - SE)  1056, 1055, 1057
1056 TGI(I)=  TGI(I) -  DECTE
     DECTE=  DECTE *  .1
1057 TGI(I)=  TGI(I) +  DECTE
     GO TO 1058
1055 CONTINUE
     CPHE= TABLE1 (TEMP(1), CPTH(1), TGI(I), NOVAL)
     CPOHE= TABLE1 (TEMP(1), CPTOH(1), TGI(I), NOVAL)
     CPH2E= TABLE1 (TEMP(1), CPTH2(1), TGI(I), NOVAL)
     CPH2OE=TABLE1 (TEMP(1), CPTH2O(1), TGI(I), NOVAL)
     CPOE=  TABLE1 (TEMP(1), CPTO(1), TGI(I), NOVAL)
     CPO2E= TABLE1 (TEMP(1), CPTO2(1), TGI(I), NOVAL)
     GAMPST= GAMMAE
     GAMMAE=  (ZNH2O  *  CPH2OE  + ZNH2 * CPH2E  + ZNO * CPOE  + ZNO2 *
```

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
1  CPO2E + ZNH * CPHE  + ZNOH * CPOHE) / (ZNH2O *  CPH2OE +
2  ZNH2 *  CPH2E + ZNO * CPOE + ZNO2 *  CPO2E + ZNH * CPHE  +
3   ZNOH * CPOHE -  ZNT  *  1.987)
 IF (ABS(GAMPST-GAMMAE).GT..0001)GO TO 961
 ZMI(I)= 100. / ZNT
 HCHE= TABLE1 (TEMP(1), H298H(1), TGI(I), NOVAL)
 HCOHE= TABLE1 (TEMP(1), H298OH(1), TGI(I), NOVAL)
 HCH2E= TABLE1 (TEMP(1), H298H2(1), TGI(I), NOVAL)
 HCH2OE=TABLE1 (TEMP(1), H29H2O(1), TGI(I), NOVAL)
 HCOE= TABLE1 (TEMP(1), H298O(1), TGI(I), NOVAL)
 HCO2E= TABLE1 (TEMP(1), H298O2(1), TGI(I), NOVAL)
 DELH= ZNH2OC * HCH2O + ZNOC * HCO + ZNO2C * HCO2 + ZNHC
1 * HCH + ZNOHC * HCOH + ZNH2C * HCH2 - ZNH2OC * 57.798
2 + ZNHC * 52.089 + ZNOHC * 9.33 + ZNO * 59.55 - (ZNH2O
3 * HCH2OE + ZNH2 * HCH2E + ZNH * HCHE + ZNOH *
4 HCOHE + ZNO * HCOE + ZNO2 * HCO2E) + ZNH2O * 57.798
5 - ZNH * 52.089 - ZNOH * 9.33 - ZNO         * 59.55
 ZISPI(I)= 9.33 * (10. * DELH) ** .5
 CPI(I)= (ZNH2O * CPH2OE + ZNH2 * CPH2E + ZNH * CPHE
1 + ZNOH * CPOHE + ZNO * CPOE + ZNO2 * CPO2E) /
2 (ZNT * ZMI(I))
 HT(I)= ZNH / ZNT
 OHT(I)= ZNOH / ZNT
 H2T(I)= ZNH2 / ZNT
 H2OT(I)= ZNH2O / ZNT
```

```
      O1T(I)= ZNO / ZNT
      O2T(I)= ZNO2 / ZNT
  936 CONTINUE
      WDOTF= WDOTF * 1000.0
      DO 1725 I= 1, NOST
      PI(I)= PI(I) * 10.E+02
 1725 CONTINUE
      IPRES= 0
      PINI= WDOTF ** 2 / (6300.0   ** 2      ) + PC / 2. +
     1 (PC ** 2 / 4. + WDOTF ** 2 / (19.41181 ** 2 * 1.E+04)) ** .5
      PIN= PINI
C
C     SET A CHECK VALUE FOR THE K EQUATION   *   *   *   *   *   **
C
    5 IHOLD= 0
      IZHNX= 0
      ITSTF= 0
      DELP= 0.
      ISAVE = 0
      ZLN= 18C.0
      TL= 27.4
      DO 10066 NOP= 1, NOST
      CPIS(NOP)= CPI(NOP)
      TGIS(NOP)= TGI(NOP)
1C066 CONTINUE
```

```
C
C     DO NESTING TO SOLVE MU VALUES     .     .     .     *     *     .     **
C
      DO 9999  I= 1 , NOST
      TGISVE=  TGI(I)
      CPSVE= CPI(I)
      TRYFS= 0.
      TSVL= 0.
  765 DO  1000  K= 1, 6
  700 DO  7710  J=1, ITAB
      IF (TMP(J) - TGI(I)) 7710, 800, 900
  800 GO TO (100, 200, 300, 400, 500, 600), K
  100 ZMUH= H(J)
      GO TO 1000
  200 ZMUOH= OH(J)
      GO TO 1000
  300 ZMUH2=  H2(J)
      GO TO 1000
  400 ZMUH2O=  H2O(J)
      GO TO 1000
  500 ZMUO1= O1(J)
      GO TO 1000
  600 ZMUO2= O2(J)
      GO TO 1000
  900 GO TO (110, 220, 330, 440, 550, 660), K
```

```
 110 ZMUH= (H(J) - H(J-1))/ (TMP(J) - TMP(J-1)) * (TGI(I) - TMP(J-1)) +
    1  H(J-1)
     GO TO 1000
 220 ZMUOH= (OH(J) - OH(J-1))/ (TMP(J) - TMP(J-1)) *(TGI(I) -
    1  TMP(J-1)) + OH(J-1)
     GO TO 1000
 330 ZMUH2= (H2(J) - H2(J-1))/ (TMP(J) - TMP (J-1)) * (TGI(I) -
    1 TMP (J-1)) + H2(J-1)
     GO TO 1000
 440 ZMUH2O= (H2O(J) - H2O(J-1))/ (TMP(J) - TMP (J-1))*(TGI(I) -
    1 TMP (J-1)) +  H2O(J-1)
     GO TO 1000
 550 ZMUO1=  (O1(J)- O1(J-1))/ (TMP(J) - TMP(J-1)) *(TGI(I) -TMP(J-1))
    1  + O1(J-1)
     GO TO 1000
 660 ZMUO2= (O2(J) - O2(J-1))/ (TMP(J) - TMP(J-1)) * (TGI(I)- TMP (J-1)
    1 )  +  O2(J-1)
     GO TO  1000
7710 CONTINUE
     WRITE (6,5588)
5588 FORMAT (1H-, 10X, 27HTGI WAS TOO LARGE FOR TABLE)
     CALL EXIT
1000 CONTINUE
C
C     THE EQUATION TO CALCULATE THE MU VALUES     *    *    *    *    **
```

```
C
 1010 ZMU(I)= (ZMI(I) / (HT(I) * ZMII(1)/  ZMUH + OHT(I) * ZMII (2) /
     1 ZMNOH + H2T(I) * ZMII(3)/  ZMUH2 + H2OT(I) * ZMII(4)/ ZMUH2O
     2 +  O1T(I) * ZMII (5) /  ZMUO1 + O2T(I) * ZMII(6)/ ZMUO2))
 1020 ZK(I)=  ZMU(I) *(CPI(I) + 1.25 *  1.987 / ZMI(I))
      IF (ITSTF - I) 1030, 1040,1040
 1030 TF(I)= TGI(I) +ZISPI(I) ** 2 * 9.81 /854.    *(ZMU(I) /
     1 (ZK(I) * CPI(I))) ** .5
      ITSTF=  I
C
C     TEST TO DETERMINE IF THIS GROUP OF EQUATIONS HAVE BEEN
C     CALCULATED IF SO GO TO THE EQUATION TO DETERMINE H
C
 1040 IF (TSVL - 1.) 1043, 1045, 1043
C
C     INTERPOLATION FOR CP IN THE H EQUATION     *        *         **
C
 1043 TWGI(I)=100.
      TWL(1)=  100.
 1044 DO 8875 KK=1, 6
      DO 8810 JJ=1, ITAB
      IF (TMP(JJ) -TWGI(I))8810,880, 990
  880 GO TO (111, 222, 333, 444, 555, 666), KK
  111 CPHN=  CPH( JJ)
      GO TO  8875
```

```
222 CPOHN=  CPOH(JJ)
    GO TO  8875
333 CPH2N=   CPH2(JJ)
    GO TO 8875
444 CPH2ON=  CPH2O(JJ)
    GO TO 8875
555 CPO1N=  CPO1(JJ)
    GO TO  8875
666 CPO2N=  CPO2(JJ)
    GO TO 8875
990 GO TO (120, 230,  340, 490, 560, 670), KK
120 CPHN=  (CPH(JJ) - CPH(JJ-1)) / (TMP(JJ)-TMP(JJ-1)) *(TWGI(I)-TMP
   1 (JJ-1)) +  CPH(JJ-1)
    GO TO 8875
230 CPOHN= (CPOH(JJ)- CPOH(JJ-1)) / (TMP(JJ) - TMP(JJ-1)) * (TWGI(I) -
   1  TMP(JJ-1)) +  CPOH(JJ-1)
    GO TO 8875
340 CPH2N=  (CPH2(JJ) - CPH2(JJ-1))/  (TMP(JJ) - TMP(JJ-1))*(TWGI(I)-
   1  TMP (JJ-1)) +  CPH2(JJ-1)
    GO TO 8875
490 CPH2ON=  (CPH2O(JJ) - CPH2O(JJ-1)) / (TMP(JJ)  -TMP(JJ-1))* (TWGI
   1(I) -TMP(JJ-1)) + CPH2O(JJ-1)
    GO TO 8875
560 CPO1N=  (CPO1(JJ) - CPO1(JJ-1)) / (TMP(JJ) - TMP(JJ-1)) * (TWGI(I)
   1  -  TMP(JJ-1)) +  CPO1(JJ-1)
```

```
      GO TO 8875
  670 CPO2N=  (CPO2(JJ) - CPO2(JJ-1)) / (TMP(JJ) - TMP(JJ-1)) * (TWGI(I)
     1 - TMP(JJ-1)) + CPO2(JJ-1)
      GO TO 8875
 8810 CONTINUE
      WRITE (6, 7769)
 7769 FORMAT (1H-, 10X, 28HTWGI WAS TOO LARGE FOR TABLE)
      CALL EXIT
 8875 CONTINUE
C
C     CALCULATION FOR CPII VALUES  *    *    *    *    *    *    *    **
C
      CPI(I) =  (HT(I) * CPHN  + OHT(I) * CPOHN +H2T(I)*CPH2N + H2OT(I)
     1  *  CPH2ON +  O1T(I) * CPO1N + O2T(I) * CPO2N) /  ZMI(I)
      TGI(I)= TWGI(I)
      TSVL= 1.
C
C     GO BACK TO INTERPOLATE WITH NEW VALUE OF TEMPERATURE THIS
C     VALUE OF CP IS USED IN THE K EQUATION FOR THIS STATION
C
      GO TO  765
 1045 HCAL(I)= .0162 * CPI(I)**.82*ZK (I)**.18 *  (PI(I) * ZISPI(I)*9.81
     1 *   ZMI(I) / ( 847.9   *  TGISVE))  ** .82  *    1. /
     2   (DT  *  AEATI(I) ** .5) * * .18  *   (TF(I) /TWGI(I))** .35
C
```

```
C     ROUTINE TO INTERPOLATE FOR H, N, AND X VALUES
C
      IF (IZHNX - I) 9711, 1050, 1050
 9711 IF (I - NOFZ)      9713, 9713, 9767
 9713 NI= NOFZ
      ZE= DT / 2. * (AEATI(I) ** .5 - 1.)
      ZI=ZE
      DO 9717 MM= 1, 3
      DO 9718 M= 1, NI
      IF (Z(M) - ZE) 9718, 4481, 9721
 4481 ZN I = ZN(M)
      ZH I = ZH(M)
      X I = X(M)
      IZHNX= 1
      GO TO 1050
 9721 GO TO (9722, 9723, 9724) , MM
 9722 ZN I = (ZN(M) - ZN(M-1)) / (Z(M) - Z(M-1)) * (ZE - Z(M-1)) +
     1 ZN(M-1)
      GO TO 9717
 9723 ZH I = (ZH(M) - ZH(M-1)) / (Z(M) - Z(M-1)) * (ZE - Z(M-1)) +
     1 ZH(M-1)
      GO TO 9717
 9724 X I = (X(M) - X(M-1)) / (Z(M) - Z(M-1) ) * (ZE - Z(M-1)) +
     1 X(M-1)
      GO TO 9717
```

```
 9718 CONTINUE
      WRITE (6,9727)
 9727 FORMAT (1H-, 10X, 31HZE WAS TOO LARGE FOR TABLE OF Z)
      CALL EXIT
 9717 CONTINUE
      IZHNX = I
      GO TO 1050
C
C     INTERPOLATION FOR VALUES FROM HEAD TO THROAT
C
 9767 IF (I - (NOZ + NOFZ + 1)) 8776, 8776, 9714
 8776 NI= NOZ + NOFZ + 1
      MMM = NOFZ
      ZE= DT / 2. * (AEATI(I) ** .5 - 1.)
      ZI=ZE
      DO 3761 MM= 1, 3
      DO 3762 M= MMM, NI
      IF (ZE - Z(M)) 3762, 3673, 4673
 3673 ZN I = ZN(M)
      ZH I = ZH(M)
      X I = X(M)
      IZHNX= I
      GO TO 1050
 4673 GO TO (4121, 3763, 3764), MM
 4121 ZN I =-(ZN(M-1) - ZN(M)) / (Z(M-1) - Z(M)) * (Z(M-1) - ZE) +
```

```
     1 ZN(M-1)
      GO TO 3761
 3763 ZH I =-(ZH(M-1) - ZH(M)) / (Z(M-1) - Z(M)) * (Z(M-1)  - ZE) +
     1 ZH(M-1)
      GO TO 3761
 3764 X I =-(X(M-1) - X(M)) / (Z(M-1) - Z(M)) * (Z(M-1) - ZE) +
     1 X(M-1)
      GO TO 3761
 3762 CONTINUE
      WRITE (6,4467)
 4467 FORMAT (1H-, 10X,16HZE WAS TOO SMALL)
      CALL EXIT
 3761 CONTINUE
      IZHNX= I
      GO TO 1050
C
C     INTERPOLATION FOR ZN, ZH, X VALUES BEFORE THROAT     *          **
C
 9714 IF (I - (NOZ + NOFZ + NOTZ)) 9715, 9715 ,6596
 9715 NI= NOZ + NOFZ + NOTZ
      MMM = NOZ + NOFZ + 1
      ZE= DT / 2. * (AEATI(I) ** .5 - 1.)
      ZI=ZE
      DO 9730 MM= 1, 3
      DO 9731 M= MMM, NI
```

```
      IF (Z(M) - ZE) 9731, 4457, 9735
 4457 ZN I = ZN(M)
      ZH I = ZH(M)
      X I = X(M)
      IZHNX= I
      GO TO 1050
 9735 GO TO (9736, 9737, 9738) , MM
 9736 ZN I = (ZN(M) - ZN(M-1)) / (Z(M) - Z(M-1)) * (ZE - Z(M-1)) +
     1 ZN(M-1)
      GO TO 9730
 9737 ZH I = (ZH(M) - ZH(M-1)) / (Z(M) - Z(M-1)) * (ZE - Z(M-1)) +
     1 ZH(M-1)
      GO TO 9730
 9738 X I = (X(M) - X(M-1)) / (Z(M) - Z(M-1) ) * (ZE - Z(M-1)) +
     1 X(M-1)
      GO TO 9730
 9731 CONTINUE
      WRITE (6,9739)
 9739 FORMAT (1H-, 10X, 47HZE WAS TOO LARGE FOR TABLE OF Z,S BEFORE THRO
     1AT)
      CALL EXIT
 9730 CONTINUE
      IZHNX= I
      GO TO 1050
C
```

```
C     CONTINUE WITH PROGRAM  •         *         *         •          **
C
 1050 AL(I)=  3.1416 •  ZN I  ** 2 + 2. • ZN I  • (ZH I - 2.*  ZN I )
 1060 DL(I)= (2. • 3.1416 • ZN I ** 2 + 4. • ZH I  • ZN I  -8.* ZN I
     1 ** 2) /  (3.1416 * ZN I  + ZH I  - 2. * ZN I )
C
C     TEST TOO SEE WHAT PRESSURES TOO INTERPOLATE BETWEEN  •    •    **
C     INTERPOLATE AT 30 ATM
 2307 CONTINUE
      IF (PIN - 1.) 3001, 3002, 3003
 3001 WRITE (6, 3004)
 3004 FORMAT (1H-, 10X, 20HPRESSURE LESS THAN 1)
      CALL EXIT
 3002 DO 3005 IZ= 1, 2
      TWGO= TWL(IZ)
      ZKLC(IZ)= TABLE1( TMPL(1),ZKL1(1),TWGO,IRAD)
      CPLC(IZ)= TABLE1 (TMPL(1),CPL1(1),TWGO,IRAD)
      ZMULC(IZ)= TABLE1( TMPL(1),ZMUL1(1),TWGO,IRAD)
 3005 TWL(2)= TL
      GO TO 1070
 3003 IF (PIN - 10.) 5001, 5002, 5003
 5001 DO 5004 IZ= 1, 2
      PERCNT= (PIN - 1.) / 9.
      TWGO= TWL(IZ)
      ZKLC1= TABLE1( TMPL(1),ZKL1(1),TWGO,IRAD)
```

```
     ZKLC2= TABLE1 (TMPL(1),ZKL10(1),TWGO,IRAD)
     DIFF= ABS(ZKLC1 - ZKLC2) • PERCNT
     ZKLC(IZ)= ZKLC 1  + DIFF
     CPLC1= TABLE1(TMPL(1),CPL1(1),TWGO,IRAD)
     CPLC2= TABLE1(TMPL(1),CPL10(1),TWGO,IRAD)
     DIFF= ABS(CPLC1 - CPLC2) • PERCNT
     IF (CPLC1 .LT. CPLC2) GO TO 5005
     CPLC(IZ)= CPLC1 - DIFF
     GO TO 5006
5005 CPLC(IZ)=  CPLC1 +  DIFF
5006 ZMULC1=  TABLE1 (TMPL(1),ZMUL1(1),TWGO,IRAD)
     ZMULC2=  TABLE1(TMPL(1),ZMUL10(1),TWGO,IRAD)
     DIFF= ABS (ZMULC1  - ZMULC2) • PERCNT
     ZMULC(IZ)= ZMULC1  + DIFF
5004 TWL(2)=  TL
     GO TO 1070
5002 DO  4001 IZ= 1, 2
     TWGO=  TWL (IZ)
     ZKLC(IZ)= TABLE1 (TMPL(1),ZKL10(1),TWGO,IRAD)
     CPLC(IZ)= TABLE1(TMPL(1),CPL10(1),TWGO,IRAD)
     ZMULC(IZ)= TABLE1  (TMPL(1),ZMUL10(1),TWGO,IRAD)
4001 TWL(2)=  TL
     GO TO 1070
5003 IF  (PIN - 20.) 6001, 6002, 6003
6001 DO  6004 IZ=  1,2
```

```
      PERCNT=  (PIN - 10.) /10.
      TWGO=  TWL(IZ)
      ZKLC1= TABLE1 (TMPL(1),ZKL10(1),TWGO,IRAD)
      ZKLC2=  TABLE1(TMPL(1),ZKL20(1),TWGO,IRAD)
      DIFF= ABS (ZKLC1   - ZKLC2)  * PERCNT
      ZKLC(IZ)= ZKLC1  + DIFF
      CPLC1= TABLE1 (TMPL(1),CPL10(1),TWGO,IRAD)
      CPLC2=  TABLE1(TMPL(1),CPL20(1),TWGO,IRAD)
      DIFF=   ABS (CPLC1 - CPLC2) *  PERCNT
      IF  (CPLC1 .LT. CPLC2) GO TO 6005
      CPLC(IZ)= CPLC1  - DIFF
      GO  TO 6006
 6005 CPLC(IZ)= CPLC1  + DIFF
 6006 ZMULC1=  TABLE1 (TMPL(1),ZMUL10(1),TWGO,IRAD)
      ZMULC2= TABLE1(TMPL(1),ZMUL20(1),TWGO,IRAD)
      DIFF=  ABS(ZMULC1  - ZMULC2) *  PERCNT
      ZMULC(IZ)=  ZMULC1 + DIFF
 6004 TWL(2)=  TL
      GO TO  1070
 6002 DO  6010  IZ= 1, 2
      TWGO=  TWL(IZ)
      ZKLC(IZ)=  TABLE1 (TMPL(1),ZKL20(1),TWGO,IRAD)
      CPLC(IZ)= TABLE1 (TMPL(1),CPL20(1),TWGO,IRAD)
      ZMULC(IZ)= TABLE1 (TMPL(1),ZMUL20(1),TWGO,IRAD)
 6010 TWL(2)= TL
```

```
      GO TO 1070
5210  DO 7004  IZ= 1, 2
      PERCNT= (PIN - 20.) / 10.
      TWGO=  TWL(IZ)
      ZKLC1=  TABLE1(TMPL(1),ZKL20(1),TWGO,IRAD)
      ZKLC2=  TABLE1(TMPL(1),ZKL30(1),TWGO,IRAD)
      DIFF=  ABS (ZKLC1 - ZKLC2)* PERCNT
      ZKLC(IZ)=  ZKLC1 + DIFF
      CPLC1=  TABLE1 (TMPL(1),CPL20(1),TWGO,IRAD)
      CPLC2=  TABLE1 (TMPL(1),CPL30(1),TWGO,IRAD)
      DIFF=  ABS (CPLC1 - CPLC2) * PERCNT
      IF (CPLC1  .LT. CPLC2) GO TO 7005
      CPLC(IZ)=  CPLC1 - DIFF
      GO TO 7006
7005  CPLC(IZ)= CPLC1 + DIFF
7006  ZMULC1=  TABLE1 (TMPL(1),ZMUL20(1),TWGO,IRAD)
      ZMULC2=  TABLE1 (TMPL(1),ZMUL30(1),TWGO,IRAD)
      DIFF=   ABS (ZMULC1-ZMULC2) *  PERCNT
      ZMULC(IZ)=  ZMULC1 + DIFF
7004  TWL(2)= TL
      GO TO 1070
6003  IF (PIN - 30.) 5210, 5211, 5212
5211  DO 7617 IZ= 1, 2
      TWGO = TWL(IZ)
      ZKLC(IZ)=  TABLE1( TMPL(1),ZKL30(1),TWGO,IRAD)
```

```
      CPLC(IZ)=TABLE1 (TMPL(1),CPL30(1),TWGO,IRAD)
      ZMULC(IZ)= TABLE1 (TMPL(1),ZMUL30(1),TWGO,IRAD)
 7617 TWL(2) = TL
      GO TO 1070
C
C     INTERPOLATE FOR PRESSURE BETWEEN 30 AND 60  *    *    *    *    **
C
 5212 IF (PIN - 60.) 5216, 5217, 5218
 5216 DO 5219  IZ =1, 2
      PERCNT= (PIN - 30.) / 30.
      TWGO =  TWL(IZ)
      ZKLC1=  TABLE1 (TMPL(1),ZKL30(1),TWGO,IRAD)
      ZKLC2=  TABLE1 (TMPL(1),ZKL60(1),TWGO,IRAD)
      DIFF=    ABS ( ZKLC1 - ZKLC2) * PERCNT
      ZKLC(IZ)=  ZKLC1 + DIFF
      CPLC1=   TABLE1 (TMPL(1),CPL30(1),TWGO,IRAD)
      CPLC2=   TABLE1 (TMPL(1),CPL60(1),TWGO,IRAD)
      DIFF=  ABS ( CPLC1 - CPLC2) * PERCNT
      IF (CPLC1 .LT. CPLC2) GO TO 4860
      CPLC(IZ)= CPLC1 - DIFF
      GO TO 6020
 4860 CPLC(IZ) =   CPLC1 +  DIFF
 6020 ZMULC1=   TABLE1 ( TMPL(1),ZMUL30(1),TWGO,IRAD)
      ZMULC2=   TABLE1 ( TMPL(1),ZMUL60(1),TWGO,IRAD)
      DIFF=   ABS ( ZMULC1- ZMULC2) * PERCNT
```

```
      ZMULC(IZ)=  ZMULC1 + DIFF
 5219 TWL(2)=  TL
      GO TO 1070
C
C     INTERPOLATE WHEN PRESSURE = 60    *    *    *    *    •    •    **
C
 5217 DO  5231 IZ= 1, 2
      TWGO=  TWL(IZ)
      ZKLC(IZ)=TABLE1(TMPL(1),ZKL60(1),TWGO,IRAD)
      CPLC(IZ)= TABLE1 (TMPL(1),CPL60(1),TWGO,IRAD)
      ZMULC(IZ)= TABLE1 (TMPL(1),ZMUL60(1),TWGO,IRAD)
 5231 TWL(2)=  TL
      GO TO 1070
C
C     INTERPOLATE FOR BETWEEN PRESSURES OF 60 AND 100  •    •    •    **
C
 5218 DO  5281 IZ= 1, 2
      PERCNT= (PIN - 60.) / 40.
      TWGO= TWL(IZ)
      ZKLC1= TABLE1( TMPL(1),ZKL60(1),TWGO,IRAD)
      ZKLC2= TABLE1( TMPL(1),ZKL99(1),TWGO,IRAD)
      DIFF=  ABS (ZKLC1 - ZKLC2)* PERCNT
      ZKLC(IZ)= ZKLC1 + DIFF
      CPLC1=  TABLE1 (TMPL(1),CPL60(1),TWGO,IRAD)
      CPLC2=  TABLE1 (TMPL(1),CPL99(1),TWGO,IRAD)
```

```
      DIFF= ABS (CPLC1 - CPLC2) * PERCNT
      IF (CPLC1 .LT. CPLC2) GO TO 7080
      CPLC(IZ)= CPLC1 - DIFF
      GO TO 7510
 7080 CPLC(IZ)= CPLC1 + DIFF
 7510 ZMULC1=  TABLE1 (TMPL(1),ZMUL60(1),TWGO,IRAD)
      ZMULC2=  TABLE1 (TMPL(1),ZMUL99(1),TWGO,IRAD)
      DIFF=    ABS ( ZMULC1 - ZMULC2) * PERCNT
      ZMULC(IZ)= ZMULC1 + DIFF
 5281 TWL(2)=  TL
 1070 CONTINUE
      IF (I- NOFZ)  5318, 5318, 5319
 5319 ZLN= 360.
 5318 HLCAL(I)= .021 *  ZKLC(2) ** .67 *  (WDOTF /(ZLN * AL(I))) ** .8 *
     1   (1./   DL(I) **.2 )  *   (1. /  ZMULC(2)  **.47) *  CPLC(2) **
     2   .33 *  (  (ZMULC(2) *  CPLC(2) / ZKLC(2)) /  (ZMULC(1)*
     3    CPLC(1) /  ZKLC(1))) ** .25
      TWLS= TL + (TF(I) - TL) /  (1. + HLCAL(I) * ((EM / ZKM)  +
     1  (1. /   HCAL(I))))
      TSTTWL =   (TWLS -  TWL(1))
      TWGS=   TF(I)  -   (TF(I) - TL) / (1. +   HCAL(I) * (EM/ ZKM  +
     1    1./ HLCAL(I)))
      TSTTWG=   TWGS - TWGI(I)
 3163 IF  ( ABS(TSTTWG) - 2. )  3169, 3169, 3167
 3167 TWGI(I)= (TWGS  + TWGI(I)) / 2.
```

```
      GO TO 1044
 3169 CONTINUE
      IF (ABS(TSTTWL) -2. )   3161, 3161 , 3162
 3162 TWL(1)= (TWLS  + TWL(1)) / 2.
      GO TO  1044
 3161 TWLCDE =  0.
C
C     INTERPOLATION FOR H AND N AT AEATI - 1 *    *    •    •    •    **
C
      IF (I .EQ. 1) GO TO 2301
      IF (IHOLD-I)  5301, 5302, 5302
 5301 IF (I-(NOFZ+1))5303, 5303, 5304
 5303 ZE=  DT/ 2. *  (AEATI(I-1) ** .5  - 1.)
      ZP1= ZE
      ZNP1= TABLE1 (Z(1),ZN(1),ZE,NOFZ)
      ZHP1= TABLE1  (Z(1),ZH(1),ZE,NOFZ)
      XP1=  TABLE1  (Z(1),X(1),ZE,NOFZ)
      IHOLD= I
      GO TO 5302
 5304 IF (I - (NOZ + NOFZ + 2))5305, 5305, 5306
 5305 NI  = NOZ + NOFZ + 2
      MMM  = NOFZ + 1
      ZE= DT / 2. * (AEATI(I-1) ** .5 - 1.)
      DO 5308  M= MMM, NI
      IF  (ZE - Z(M)) 5308, 5309,  5310
```

```
5309 ZP1=  ZE
     ZNP1= ZN(M)
     ZHP1= ZH(M)
     XP1= X(M)
     IHOLD= I
     GO TO 5302
5310 ZP1= ZE
     ZNP1= (ZN(M-1) - ZN(M)) / (Z(M-1)-  Z(M)) * (Z(M-1) -ZE)  +
    1 ZN(M)
     ZHP1= (ZH(M-1) - ZH(M)) / (Z(M-1) - Z(M))*  (Z(M-1) - ZE)  +
    1 ZH(M)
     XP1= (X(M-1) -X(M))/ (Z(M-1) - Z(M)) * (Z(M-1)-ZE) +
    1 X(M)
     IHOLD=I
     GO TO 5302
5308 CONTINUE
     WRITE (6, 5311)
5311 FORMAT (1H-, 10X,  34HZE FOR VALUES PLUS 1 WAS TOO SMALL)
     CALL EXIT
5306 NI  = NOST + 1
     MMM  = NOZ + NOFZ + 2
     ZE=  DT/ 2. * (AEATI(I- 1) ** .5 - 1 .)
     DO 5313 M= MMM, NI
     IF  ( Z(M)-ZE) 5313, 5314, 5315
5314 ZP1= Z(M)
```

```
      ZNP1= ZN(M)
      ZHP1= ZH(M)
      XP1= X(M)
      IHOLD= I
      GO TO 5302
 5315 ZP1= ZE
      ZNP1= (ZN(M) -ZN(M-1))/ (Z(M)-Z(M-1)) * (ZE-Z(M-1)) +
     1 ZN (M-1)
      ZHP1= ( ZH(M) -ZH(M-1)) / (Z(M) - Z(M-1)) * (ZE - Z(M-1)) +
     1 ZH(M-1)
      XP1= (X(M) - X(M-1)) / (Z(M) - Z(M-1)) * ( ZE - Z(M-1)) +
     1 X(M-1)
      IHOLD=I
      GO TO 5302
 5313 CONTINUE
      WRITE (6,5316)
 5316 FORMAT (1H-, 10X, 23HZE PLUS 1 WAS TOO SMALL)
      CALL EXIT
 5302 CONTINUE
      ZLP= ((ABS (XP1    -X I )) ** 2 + (ABS(ZP1   -Z I ))**2) **.5
      AC= (ZN I *ZH I +ZNP1     * ZHP1    )-(.4292* (ZN I **2+ZNP1   **2))
      DLC=  4. * (ZN I  * ZH I  + ZNP1 * ZHP1  -   .4292 *
     1  (ZN I   ** 2  + ZNP1 ** 2 )) / (3.1416 *      (ZN I   + ZNP1) +
     2 ZH I  + ZHP1 -   2. * (ZN I  + ZNP1))
C
```

```
C     INTERPOLATE FOR GAMMA AND MN      *    *    *    *    •    *    **
C
      A(I)= 2.2000 / 2. * (ZN I  + ZNP1   ) • ZLP
      TRYS= (((ZLN *A(I  ))/ (2. * WDOTF • CPLC(2)) )  *
     1  (HCAL(I-1) * (TF(I-1) - TWGI(I-1)) + HCAL(I) • (TF(I) -
     2  TWGI(I)))  )
      TL= TRYS + TL - TRYFS
      TRY= TL
      IF (PIN - 1.) 2701, 2702, 2703
 2701 WRITE (6, 2704)
 2704 FORMAT (1H-, 10X, 28HTHE PRESSURE WAS LESS THAN 1)
      CALL EXIT
 2702 GAMC= TABLE1 (TMPL(1),GAM1(1),TRY,IRAD)
      ZMUC= TABLE1 (TMPL(1),ZMUL1(1),TRY,IRAD)
      GO TO 5335
 2703 IF (PIN - 10.) 2501, 2502, 2503
 2501 GAMC1= TABLE1 (TMPL(1),GAM1(1),TRY,IRAD)
      GAMC2= TABLE1 (TMPL(1),GAM10(1),TRY,IRAD)
      PERCNT= (PIN -1.) / 9.
      DIFF= ABS( GAMC1 - GAMC2) * PERCNT
      GAMC= GAMC + DIFF
      ZMUC1= TABLE1 (TMPL(1),ZMUL1(1),TRY,IRAD)
      ZMUC2= TABLE1 (TMPL(1),ZMUL10(1),TRY,IRAD)
      DIFF= ABS( ZMUC1 - ZMUC2) • PERCNT
      ZMUC= ZMUC 1   + DIFF
```

```
     GO TO 5335
2502 GAMC= TABLE1 (TMPL(1),GAM10(1),TRY,IRAD)
     ZMUC= TABLE1 (TMPL(1),ZMUL10(1),TRY,IRAD)
     GO TO 5335
2503 IF (PIN - 20.) 2001, 2002, 2310
2001 GAMC1= TABLE1 (TMPL(1),GAM10(1),TRY,IRAD)
     GAMC2= TABLE1 (TMPL(1),GAM20(1),TRY,IRAD)
     PERCNT=  (PIN - 10.) / 10.
     DIFF=  ABS (GAMC1 - GAMC2) * PERCNT
     GAMC= GAMC1 + DIFF
     ZMUC1= TABLE1 (TMPL(1),ZMUL10(1),TRY,IRAD)
     ZMUC2= TABLE1 ( TMPL(1),ZMUL20(1),TRY,IRAD)
     DIFF= ABS (ZMUC1 - ZMUC2) * PERCNT
     ZMUC= ZMUC1 + DIFF
     GO TO 5335
2002 GAMC= TABLE1 (TMPL(1),GAM20(1),TRY,IRAD)
     ZMUC= TABLE1 (TMPL(1),ZMUL20(1),TRY,IRAD)
     GO TO 5335
2310 IF (PIN - 30.)  5321, 5322, 5323
5321 GAMC1= TABLE1 (TMPL(1),GAM20(1),TRY,IRAD)
     GAMC2= TABLE1 (TMPL(1),GAM30(1),TRY,IRAD)
     PERCNT= (PIN - 20.) / 10.
     DIFF= ABS(GAMC1 - GAMC2) * PERCNT
     GAMC= GAMC1 + DIFF
     ZMUC1= TABLE1 (TMPL(1),ZMUL20(1),TRY,IRAD)
```

```
      ZMUC2= TABLE1 (TMPL(1),ZMUL30(1),TRY,IRAD)
      DIFF= ABS(ZMUC1 - ZMUC2) * PERCNT
      ZMUC= ZMUC1 + DIFF
      GO TO 5335
 5322 GAMC=  TABLE1 (TMPL(1),GAM30(1),TRY,IRAD)
      ZMUC=  TABLE1 (TMPL(1),ZMUL30(1),TRY,IRAD)
      GO TO   5335
 5323 IF (PIN-60.)  5325, 5326, 5327
 5325 GAMC1=TABLE1 (TMPL(1),GAM30(1),TRY,IRAD)
      GAMC2= TABLE1 (TMPL(1),GAM60(1), TRY, IRAD)
      PERCNT= (PIN - 30.) / 30.
      DIFF=    ABS (GAMC1- GAMC2) * PERCNT
      GAMC=  GAMC1 + DIFF
      ZMUC1=  TABLE1 (TMPL(1),ZMUL30(1),TRY,IRAD)
      ZMUC2=  TABLE1 ( TMPL(1),ZMUL60(1),TRY,IRAD)
      DIFF=    ABS (ZMUC1 - ZMUC2) * PERCNT
      ZMUC=    ZMUC1 + DIFF
      GO TO 5335
 5326 GAMC=  TABLE1 (TMPL(1),GAM60(1),TRY,IRAD)
      ZMUC=  TABLE1 (TMPL(1),ZMUL60(1),TRY,IRAD)
      GO TO 5335
 5327 GAMC1=  TABLE1 (TMPL(1),GAM60(1),TRY,IRAD)
      GAMC2=  TABLE1 (TMPL(1),GAM99(1),TRY,IRAD)
      PERCNT= (PIN - 60.) / 40.
      DIFF=   ABS ( GAMC1 - GAMC2) * PERCNT
```

```
      GAMC=  GAMC1 + DIFF
      ZMUC1= TABLE1 (TMPL(1),ZMUL60(1),TRY,IRAD)
      ZMUC2=  TABLE1 (TMPL(1),ZMUL99(1),TRY,IRAD)
      DIFF=   ABS ( ZMUC1- ZMUC2) * PERCNT
      ZMUC=   ZMUC1 + DIFF
 5335 DEL= .1
      DELPF= .00001
 5336 TERM1= 1. / ((9.81E5 * GAMC* ZLN ** 2 * AC** 2*(DLC/ ZLP)*
     1 DELPF) / (2. * WDOTF ** 2)) ** .5
      TERM2= 2.28 + 4. * ALOG10(((9.81E5 * GAMC* ZLN ** 2 *
     1 AC** 2 * (DLC/ ZLP)* DELPF) / (2. * WDOTF ** 2)) ** .5*((
     2 WDOTF * DLC) / (ZLN * AC* ZMUC)))- 4. * ALOG10(   .01
     3 *((9.81E5*GAMC* ZLN ** 2 * AC** 2 * (DLC/ ZLP)* DELPF) /
     4 (2. * WDOTF ** 2)) ** .5 * (( WDOTF * DLC)/ (ZLN * AC*
     5 ZMUC))+ 4.67)
      TEST=  TERM1 - TERM2
      IF  (ABS(TEST) - .0005) 5341, 5341, 5342
 5342 IF (TEST) 5343, 5341, 5344
 5343 DELPF= DELPF - DEL
      DEL= DEL  * .1
 5344 DELPF= DELPF + DEL
      GO TO 5336
 5341 CONTINUE
 2308 CONTINUE
      IF (ISAVE .EQ. I) GO TO 2372
```

```
      ISAVE= I
      PIN= PIN - DELPF
      DELPFS= DELPF
      TRYFS= TRYS
       TSVL= 0.
      ITSTF= I-1
      IZHNX= I-1
      IHOLD= I-1
      TGI(I)= TGISVE
      CPI(I)= CPSVE
      GO TO 765
 2372 PIN= PIN - DELPF + DELPFS
 2301 CONTINUE
 8888 CONTINUE
 9999 CONTINUE
       IF (IPRES .EQ. 2) GO TO   90075
      IPRES= 1   + IPRES
      PINJ= PC / 2.   + (PC ** 2 / 4. + WDOTF ** 2 / (19.41181
     1 ** 2 * 1.E+06) * TL ) ** .5
      PIN= -(PIN - PINJ) + PINI
      PINI= PIN
      DO 10067 NOP= 1, NOST
      CPI(NOP)= CPIS(NOP)
      TGI(NOP)= TGIS(NOP)
10067 CONTINUE
```

FORTRAN IV PROGRAM LISTING OF MAIN PROGRAM

```
      GO TO 5
90075 WRITE (6, 8081)
 8081 FORMAT (1H-)
      PC= PC * 14.7 - 14.7
      TC= TC * 9. / 5. - 459.67
      PIN= PIN * 14.7    - 14.7
      TL= TL * (9./ 5.)      - 459. 67
      XAXIS (IKM) = WDOTF / 4.4466
      YAXIST(IKM) = TL
      YAXISP (IKM) = PIN
      WRITE (6, 70009) PC, TC
70009 FORMAT (1H-, 10X, 25HCHAMBER PRESSURE (PSIG) =E15.8, 10X,
     1 30HCHAMBER TEMPERATURE (DEG. F) =E15.8)
      WRITE (6, 1619) PIN, TL
 1619 FORMAT (1H-, 10X, 23HFINAL PRESSURE (PSIG) =E15.8, 10X,
     1 28HFINAL TEMPERATURE (DEG. F) =E15.8)
      WRITE (6, 8081)
      WRITE (6, 8081)
70002 CONTINUE
              WRITE (6, 30036) XAXIS
      WRITE (6, 30036) YAXIST
      WRITE (6, 30036) YAXISP
30036 FORMAT (1H1, 10X, 6E15.8)
 6596 CALL EXIT
      END
```

FORTRAN IV PROGRAM LISTING OF SUBROUTINE CONV

```
      SUBROUTINE CONV(T,PT,C1,C2,XNH2O,XNO,XNO2,XNH,XNOH,XNH2,XNT)
      COMMON /TABLE/ TT(100),CC3(100),CC4(100),CC5(100),CC6(100)
      C3 = 10.**TABLE1(TT(1),CC3(1),T,51)
      C4 =10.**(-TABLE1(TT(1),CC4(1),T,51))
      C5 = 10.**TABLE1(TT(1),CC5(1),T,51)
      C6 = 10.**TABLE1(TT(1),CC6(1),T,51)
      GAMMA = .5
      XNO2 = 1.
    3 CONTINUE
      XNO = C6 * SQRT(XNO2/GAMMA)
      D1 = GAMMA/C3
      D5 = SQRT(XNO2)
      D3 = SQRT(XNO2) * D1
      D4 = SQRT(GAMMA)
      D6 = D4*D5
      D7 = D1/C3
      A = 2.*D7*(1.+D6/C4)
      B = C5*D6/C3+1.
      XNH = (-B+SQRT(B**2-4.*A*(-C1)))/(2.*A)
      D2 = XNH**2
      XNH2 = D7*D2
      XNH2O = D6*D7*D2/C4
      XNOH = D6*XNH*C5/C3
      XNNO = XNH2O + XNOH + XNO + 2. * XNO2
      XNT = XNH2O + XNH2 + XNH + XNOH + XNO2 + XNO
```

FORTRAN IV PROGRAM LISTING OF SUBROUTINE CONV

```
      TEST = C2-XNNO
      IF(ABS(TEST).LT..00005) GO TO 1
      AA = (2.*XNH2O + XNOH)/(C1 + 2.*(XNH2O + XNH2))
      DNO2 = 2.*XNO2*TEST/(XNH2O * (1.-2.*AA) + XNOH * (1.-AA) +
     14.*XNO2 + XNO)
      XNO2 = XNO2 + DNO2
      IF(XNO2.GT.0.) GO TO 3
      XNO2 = (XNO2-DNO2) * .1
      GO TO 3
    1 CONTINUE
      P = GAMMA*XNT
      TEST1 = P/PT-1.
      IF(ABS(TEST1).LT..00005) GO TO 5
      GAMMA = PT/XNT
      GO TO 3
    5 CONTINUE
      IF(XNOH.LT..5E-6 ) XNOH = 0.
      IF(XNH2O .LT..5E-6 ) XNH2O = 0.
      IF(XNH2.LT..5E-6 ) XNH2 = 0.
      IF(XNH.LT..5E-6 ) XNH = 0.
      IF(XNO .LT..5E-6 ) XNO = 0.
      IF(XNO2.LT..5E-6 ) XNO2 = 0.
      IF(XNT.LT..5E-6) XNT =0.
      RETURN
      END
```

FORTRAN IV PROGRAM LISTING OF SUBROUTINE TABLE1

```
      FUNCTION TABLE1 (X,Y, XK, NPTS)
      DIMENSION  X(100), Y(100)
      DO 75 I=1, NPTS
      IF (X(I) - XK) 75, 5201, 5202
 5201 TABLE1=  Y(I)
      RETURN
 5202 TABLE1=  (Y(I)-  Y(I-1))/  (X(I) - X(I-1)) * (XK - X(I-1))
     1 +   Y(I-1)
      RETURN
   75 CONTINUE
      WRITE (6, 5207)
 5207 FORMAT (1H-, 10X,  23HFUNCTION TABLE EXCEEDED)
      CALL EXIT
      END
```

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Brown Engineering Company, Inc.	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

"A FORTRAN Program for Heat Transfer Analysis of the J-2 Engine"

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Note, March 1965

5. AUTHOR(S) (Last name, first name, initial)

Floyd, J. G., Smith, B. G., and Seidel, H. H.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March, 1965	113	2

8a. CONTRACT OR GRANT NO. NAS8-5289	9a. ORIGINATOR'S REPORT NUMBER(S) R-127
b. PROJECT NO.	
c. Not applicable	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) None
d.	

10. AVAILABILITY/LIMITATION NOTICES

None

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
None	NASA

13. ABSTRACT

This report describes an IBM 7040 FORTRAN computer program for determining the heat transfer and pressure drop across the jacket of the regeneratively cooled J-2 engine. Thermochemical and heat transfer analyses were performed and are provided in two separate decks, "Thermochemical Analysis Program of the J-2 Engine", Program No. SP-172, and "Heat Transfer Computation Program of the J-2 Engine", Program No. SP-173. The entire program is incorporated in "A FORTRAN Program for Heat Transfer Analysis of the J-2 Engine", Program No. SP-148.

14. KEY WORDS

heat transfer
regenerative cooling
J-2 engine
LOX - LH2
equilibrium expansion
pressure drop
temperature rise
thermochemistry
FORTRAN program
rocket