TECHNICAL REPORT
HTL-TR-41

FINAL REPORT
CONTRACT NAS8-11850
Vol. IV
TURBULENT MIXING OF THREE CONCENTRIC REACTING JETS

HEAT TECHNOLOGY LABORATORY, INC.
Huntsville, Alabama

September 29, 1967

Prepared For

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Huntsville, Alabama

ACKNOWLEDGEMENT

The efforts of Mrs. E. M. McAllister and Mr. J. V. Hyden as the scientific programmers for this analysis are gratefully acknowledged.

TABLE OF CONTENTS

	Page
SUMMARY	1
INTRODUCTION	2
LIST OF SYMBOLS	2
DISCUSSION	3
Program Input	4
Program Internal Structure	5
Computational Problems	6
Program Output	8
Presentation of Results	8
CONCLUSIONS AND RECOMMENDATIONS	10
APPENDIX A-COMPUTER PROGRAM	12
LISTING OF TWO-STREAM-MIXING COMPUTER PROGRAM	19
LISTING OF THREE-STREAM-MIXING COMPUTER PROGRAM	45
REFERENCES	71
TABLE I	
FIGURES	

PART II

TURBULENT MIXING OF THREE CONCENTRIC REACTING JETS

By Beverly J. Audeh
Heat Technology Laboratory, Inc.

SUMMARY

The results of a computer program based on a modified P - function solution to the multiple-stream-turbulent-mixing problem are presented in graphical form. Though a computational problem persists in the determination of the absolute magnitude of the temperature, the plots of the normalized properties indicate that the trends of the mixing and combustion processes are properly defined. Recommendations to solve the computational problem in the temperature evaluation and to improve the program are outlined. It is concluded with the addition of a recommended subroutine the program would provide a method of describing mixing of equilibrium gases with greater speed than the available programs involving non-equilibrium calculations.

Complete listings of the two-stream-mixing program and the three-stream-mixing program are presented. The input format is illustrated by sample input sheets, for both two and three stream calculations.

INTRODUCTION

A digital computer program has been constructed to evaluate the mixing and combustion characteristics of a rocket exhaust jet with turbine gases exhausting peripherally about the nozzle exit and with a surrounding external air stream. This study is a continuation of work reported in Part I (Ref. 1) of this report. Part I describes the analytical model and the equations used in the program. Computer runs were made for cases approximating the afterburning of H-1 engine flows at low altitudes. The output is in the form of velocity, temperature, and species mass fractions plotted as functions of position. The results of the three-stream-mixing analysis are presented.

LIST OF SYMBOLS

C_i	Mass fraction of the i th species
H	Total enthalpy, cal/g
h	Static enthalpy (intermediate value defined by Eq. (9), Ref. 1), cal/g
h'	Static enthalpy (defined by Eq. (10), Ref. 1) cal/g
r	Radius, cm
T	Temperature, °K
U	Velocity in the x direction, cm/sec
x	Distance in axial direction, cm

Z	ψ/ψ_e
ξ	Transformation variable in axial direction (defined by Eq. (16), Ref. 1)
ρ	Density, g/cm^3
ψ	Dimensionless stream function defined by Eq. (4), Ref. 1
$\bar{\psi}$	Dummy integration variable in the P-function integration

Subscripts:

e	Indicates external conditions
j	Indicates rocket exhaust exit conditions
ref	Reference values
te	Indicates turbine exhaust exit conditions

Superscripts:

*	Dimensionless quantity

DISCUSSION

The solution of the flow equations for the velocity and energy values at a position in a transformed mathematical plane was obtained from the P function representation of the mixing process. With these physical conditions specified, the thermodynamic properties were then determined. All chemical processes were represented by one-step reactions. An atom balance was used with molecular composition determined to comply with the equilibrium restraint. Transformations were used to position these results in the physical plane.

Program Input. - The program constructs and solves the equations defining the mixing and combustion according to the numerical (key) instructions placed in the input. A prior knowledge of the possible reaction products is required. A lower limit upon the mass fractions considered is used as an input variable to allow flexibility in the compromise between accuracy desired and required run time.

The P function solution for the three-stream-mixing cases is modified using a table look-up of the modification factor. The following equation

$$\text{Modification Factor} = \frac{\left(\begin{array}{c}\text{Value at Point}\\ \text{of Consideration}\end{array}\right) - \left(\begin{array}{c}\text{Value in the}\\ \text{External Stream}\end{array}\right)}{\left(\begin{array}{c}\text{Rocket Exhaust}\\ \text{Value}\end{array}\right) - \left(\begin{array}{c}\text{External Stream}\\ \text{Value}\end{array}\right)}$$

was used to determine this factor from the results of the two separate two-stream mixing cases. The modification factor is evaluated for each elemental composition, the enthalpy and the speed. With these modification terms in the calculation procedure, it is possible to represent arbitrary additions of energy and chemical composition such as that of the turbine exhaust. The modified P function solution is still a similar type solution with a Gaussian decay but the initial profiles may be of arbitrary shape.

Because of the equilibrium restriction the program is highly dependent upon the input thermodynamic data. Equilibrium constants were input directly from Ref. 2 in tabular

form. Thermodynamic constants from Ref. 3 were input in array format for use in evaluating the static enthalpy in the mixture of gases present.

Program Internal Structure. - The P-function solution was modified by a factor obtained by normalizing the conditions present at the sub-initial plane. In the three-stream-mixing problem, the normalized conditions at the sub-initial plane replace the initial values used previously in the two-stream mixing case. To provide for the addition of arbitrary chemical species and energy levels, each of the properties and elements are calculated separately using the appropriate modification factor determined by the normalization procedure. A table look-up of the modification factor with provisions for linear interpolation of values is used. This method of storing and retrieving the modification terms was found to be faster and more accurate than a polynomial representation of the modifying factor in the P-function.

The thermodynamic condition of the equilibrium mixture is determined by constructing a matrix of equations involving the unknown mass fractions, density, and temperature. The equations are a combination of algebraic and transcendental equations. A Newton-Raphson iteration technique is applied to evaluate the unknowns. Logarithmic corrections for the gaseous constituents, as shown in Ref. 4, are used to accelerate the convergence of the equilibrium composition calculations.

The atom balance is maintained by use of a numerical

code for each molecule within the system. In the code each atomic species is assigned a particular numerical designation.

Separate subroutines are provided for the transformation of the results from the mathematical plane to the physical plane. The transformation to the axial and the radial position locations are independent subroutines so that the uses may easily alter the form of either of the transformations.

Computational Problems. - Some computational problems persist in the program. Iteration is carried out upon the static enthalpy of the mixture to determine the correct temperature. This involves the calculation of the partial enthalpy for each species from the calculated values of temperature and the mass fractions. To limit computer run time the number of iterations necessary to determine the density and mass fractions must be kept low. This was accomplished by adjusting the tolerance on the mass fraction calculation. Investigation of several cases shows that the mass fractions of some species, though present only in trace amounts, influenced the calculated enthalpy value of the mixture because of their large heat of formation values. The temperature of the mixture is calculated from the enthalpy iteration. Because of the influence of the species in trace amounts with high heats of formation, the computed magnitudes of the temperature are too high. Since the temperature plots appear as normalized curves, the trends and profiles are nevertheless properly described. A possible solution to this problem of

computing the temperature is to establish a test for each species mass fraction and to eliminate those that appear only in trace amounts from the enthalpy calculation. Experimentation would be required to determine the proper limit to place upon the mass fractions. There was insufficient time to attempt any trial of this approach.

Divergence of the solution for the mass fractions and density occurs when the initial estimates are far from the correct values. This is an inherent problem of the Newton-Raphson iteration procedure (Ref. 5). Methods to speed or force convergence exist but it was considered unadvisable to resort to these more sophisticated methods while problems such as that with the enthalpy iteration were present.

A study of expedient methods to eliminate these problems indicated that the addition of the capability of using the Lewis Thermochemical program (Ref. 3) as a subroutine would be the most useful. Inclusion of this capability would not only solve the two previously described problems but in addition would make it unnecessary to provide a priori information about the resultant products of the reactions.

The addition of this subroutine would be facilitated by using the modification of the Lewis program carried out by R. Taylor (Ref. 6) to calculate adiabatic flame temperatures. It presently uses an internal table look-up for variable fuel to oxidant ratios. This modification is restricted at present to a minimum of two fuel to oxidant

ratios. With this restraint removed, the subroutine could be used to calculate the thermodynamic properties given the present input to the Newton-Raphson subroutine.

Program Output. - The output of the program is printed in increments of the transformed stream function for a fixed transformed axial location. Values for the mass fractions, temperature, density, velocity, and enthalpy are printed after each increment. These values are then located in the physical plane with a print-out of the corresponding axial and radial positions.

Presentation of Results. - The results of the analysis of the mixing and combustion are shown as plots in Figs. (1) - (11). The values are plotted as normalized values using initial conditions of the rocket exhaust exit, the turbine exhaust, or the external stream as indicated in the individual coordinate labels. The normalized values are plotted versus the radial location for the fixed axial location which corresponds to the fixed transformed axial location.

Two separate mixing zones were calculated for two-stream-mixing. One case represented the mixing of rocket exhaust and turbine exhaust while the second was for the mixing of turbine exhaust and the external air stream. The intersection of these two mixing zones was calculated to occur at a radial position of 23.3 inches and an axial location of 1.2 inches from the nozzle exit. The sub-initial plane was located by

this intersection.

Examination of the trends shown by the normalized temperature plots indicate that mixing and combustion of the relative cold turbine exhaust and rocket exhaust are properly described. The burning of the exterior portion of the turbine exhaust is shown. The cooling of the rocket exhaust by the cooler turbine exhaust mixing together on the periphery of the rocket exhaust is shown as a dip in the curves in Fig. (2). The onset of combustion of the turbine exhaust is displayed by the rise in temperature following the dip in the temperature curve.

The maximum peak of the temperature curves indicates the secondary combustion of the fuel rich rocket exhaust and turbine exhaust mixture which has mixed with the surrounding air. It should be noted that the temperature caused by the combustion of the turbine exhaust products is less than the initial temperature value of rocket exhaust.

The plot of normalized mass fraction of water (Fig. (6)) displays the action of mixing of the turbine exhaust and rocket exhaust both being fuel-rich with no combustion occuring. This is indicated by the dip in the water profile. As a fuel rich turbine exhaust mixes with the surrounding air, combustion produces a rapid rise in the mass fraction of water present.

The remaining species reflect the mixing and combustion as shown in Figs. (4) - (11).

CONCLUSIONS AND RECOMMENDATIONS

Examination of the plots of normalized properties of the three-mixing-streams indicates that the trends of the process are properly defined. A computational problem of determining the correct magnitude of temperature still persists because of the dominating influence in the enthalpy iteration of species with high heats of formation which are in reality present only in trace amounts. Though the absolute magnitude of the temperature curves is too high, the pattern of the normalized temperature predicts the cooling of the rocket exhaust through mixing with the turbine exhaust and the rise in temperature because of combustion of the mixture of turbine exhaust and air as the radial position moves outward from the centerline. The mixing without combustion of the fuel-rich rocket exhaust and turbine exhaust followed by the combustion of the fuel-rich turbine exhaust when mixed with the surrounding air is also predicted by the plots of the normalized mass fractions of the species.

It is recommended that capability of using the Lewis Thermochemical program as a subroutine be added to the present program. This would eliminate the need for a priori knowledge of reaction products and by-pass the Newton-Raphson subroutine used to calculate the unknown mass fractions.

It is concluded based upon the normalized profiles that the modified P-function adaptation of the similar profile

method is a feasible approach to the multiple-stream-mixing problem. With the addition of the recommended subroutine the program would provide a method of describing mixing of equilibrium gases with greater speed than the available programs involving non-equilibrium calculations.

APPENDIX A

COMPUTER PROGRAM

The following description provides information for those desiring the particular results formulated by the program. This program deals with symbols and is very flexible in the number of equations and species which may be used as known information. Listings of both the two and three-stream-mixing programs are given. A sample is presented for the two-stream-mixing in Figs. (12) - (15). Three-stream-mixing sample input is shown in Figs. (16) - (21).

Limitations. - The program has the following limitations:

1. The maximum number of input equations is 10.
2. The maximum number of species is 19.
3. Each specie table has a maximum of 20 values.
4. The maximum number of equations that can be formed from the input equations is 15.
5. The initial estimate for each unknown species must be positive.
6. The values of the species tables must be input as logarithms of the base ten. This is really not a limitation since most thermochemical tables are given in this form.

List of Input Variables. -

1. KT = number of values in each species table
2. NSMAX = number of species in the equations omitting nitrogen

3. NRO = number of input equations

   a. NRO1 = NRO + 1

   b. NRO2 = NRO + 2

   c. NSMAX1 = NSMAX + 1

   d. NSMAX2 = NSMAX + 2

   e. NRS = NSMAX1 - NRO1

   f. NRS1 = NRS + 1

4. NXG, a zero indicates one set of initial estimates of the unknown species mass fractions; a one indicates multiple sets of initial estimates of the unknown species mass fractions.

5. D1(I) where I = 1,NRS1 are constants used in forming the last NRS1 equations. It is the mass fraction of elements in the outer mixing region.

6. G1(I) where I = 1, NRS1 are constants used in forming the last NRS1 equations. It is the mass fraction of elements in the inner mixing region.

7. SMASS (I) = mass of species I where I = 1, NSMAX

8. NALPHP (I,J) where I = 1, NRO and J = 1, NSMAX (see below)

9. NALPHPP (I,J) where I = 1, NRO and J = 1, NSMAX (see below) NALPHP and NALPHPP will be discussed later.

10. PSI = ψ

11. DPSIBAR = $\Delta\bar{\psi}$ = ΔZ

12. DXI = $\Delta\xi$

13. DPSI = $\Delta\psi$

14. XI = ξ

15. PSIMAX = maximum value of ψ

16. XIMAX = maximum value of ξ

17. TREF = T_{ref}, Reference temperature

18. PREF = ρ_{ref}, Reference density

19. REFV = Reference number of moles

20. HE = H_e

21. HJ = H_j

22. ACCTOL = tolerance of $|h-h'|$ used in the enthalpy iteration to determine temperature

23. PSIJ = ψ_j

24. T = T is the input estimate for the temperature

25. DELTAT = ΔT or change in calculating a new T

26. DT = ΔT used as dT in the integration to determine static enthalpy

27. TOL = tolerance setting for solution of simultaneous equations

28. PSIBAR = $\bar{\psi}$ = Z

29. UJ = U_j

30. UE = U_e

31. USTARJ = U_j^*

32. RHOE = ρ_e

33. RH = r_j

34. CNITROM = mass of nitrogen

35. AACON (I,J,K) where I = 1,6 and J = 1, NSMAX with J = index of specie used for the thermochemical constants and K = 1,2 for higher and lower temperature ranges respectively.

36. Aθ = a_o and PLYC = $a_1, a_2 \ldots, a_{20}$ are coefficients of the polynomial for $F(\bar{\psi})$ used in the P-function integration. [Note: Used only in two-stream-mixing program.]

37. PSI TABLE is used for interpolation base for the modification factor tables. [Note: Used only in three-stream-mixing program.]

38. MODIFICATION FACTOR tables are of the same size as the PSI TABLE. There are complete tables for velocity, first element, second element, ..., thru NRS1 elements, and total enthalpy. [Note: Used only in three-stream-mixing program.]

39. TTAB(K) where K = 1,KT and used for interpolation base for T and species tables.

40. STAB (I,J) where I = 1,KT and J = 1, NSMAX with J = index of species. Therefore, there are NSMAX species tables with KT values in each.

41. AAHOT (I,K) where I = 1,6 and K = 1,2
AAHOT is thermochemical constants for nitrogen.

42. KSPEC (I,J) where I = 1,4 and J = 1, NSMAX will be discussed later.

43. XG(I) are the estimates of the final values of the unknown species mass fraction where I = 1, NSMAX.

44. XG(I) CUT-OFF places a lower limit on the size of the species mass fractions to eliminate computer underflow.

45. SUM provides a means of restarting the calculation and still maintain the accuracy in the radial position integration. It is equal to the value of RSTARS print-out for the previous position.

Data Format. - Data is read by two different formats. The first format reads fixed point integers by the following set up:

a. Each data card has 12 fields of width 5. Each value must be right justified in the field. For example: If the numbers 20,5 and 2 are to be read then 20 would be placed in the first field, 5 in the second field and 2 in the third field. A 2 would be placed in column 4, 0 in column 5, 5 in column 10 and a 2 in column 15.

b. If an array N has a dimension of 15 then the values N(1), N(2),..., N(12) would be punched in the first card with N(13), N(14), N(15) in the second card.

c. If an array, say K, is two-dimensional of maximum dimension of 15 x 20 but only a 3 x 4 matrix is needed then 12 values would be read by columns in the order K(1,1), K(2,1), K(3,1), K(1,2), K(2,2), K(3,2),...,K(1,4), K(2,4), K(3,4), K(4,4).

Note: Most of the input to this program is of the form A(I,J) with I = 1,..., NRO and J = 1,..., NSMAX where NRO and NSMAX are values which are known and change with each new case.

The second format reads real or floating point numbers by the following set up:

a. Each data card has 6 fields of width 12. Each value may be input as a number which contains a demical or as a number which is given with an exponent. For example; -.000560511 and -5.605011E-4.

Formulation of Input. NALPHAP (I,J) and NALPHPP(I,J) are two-dimensional arrays where I = 1,2..., NRO and J = 1,2,..., NSMAX. NALPHP and NALPHPP, the stoichiometric coefficients, are used in the program and must be set up in a particular way. These two arrays are used to input the coefficients of the reaction equations.

Example:

$$2H_2 + O_2 \qquad 2H_2O$$

$$2C + O_2 \qquad 2CO$$

$$C + O_2 \qquad CO_2$$

$$C + 2H_2 \qquad CH_4$$

4 equations hence NRO = 2

7 species hence NSMAX = 7

Therefore, by reading the first equation from the left, there is H_2 (species 1), O_2 (species 2), and H_2O (species 3). Reading through the remaining three equations gives C (species 4), CO (species 5), CO_2 (species 6), and CH_4 (species 7).

The input for NALPHP and NALPHPP is as follows:

NALPHP

H_2	O_2	H_2O	C	CO	CO_2	CH_4
2	1	0	0	0	0	0

NALPHP (Con't)

H_2	O_2	H_2O	C	CO	CO_2	CH_4
0	1	0	2	0	0	0
0	1	0	1	0	0	0
2	0	0	1	0	0	0

NALPHPP

H_2	O_2	H_2O	C	CO	CO_2	CH_4
0	0	2	0	0	0	0
0	0	0	0	2	0	0
0	0	0	0	0	1	0
0	0	0	0	0	0	1

Notice that the column i corresponds to the coefficients of species i and row j corresponds to the equation j. For example, NALPHPP (2,5) represents the second equation and the fifth species (CO). The stoichiometric coefficient of CO in the second equation is NALPHPP (2,5) which is 2. When these arrays are loaded into the machine, all the NALPHP coefficients for the first species are inputed before the coefficients of the remaining species [See Fig. (12)].

KSPEC (I,J) is a two-dimensional array used in the program where I = 1,2,3,4, and J = 1,2,..., NSMAX with J = species J. For the above example,

KSPEC

H_2	O_2	H_2O	C	CO	CO_2	CH_4
1	2	1	0	2	2	1
0	0	2	4	4	0	4
0	0	0	0	0	4	0
0	0	0	0	0	0	0

The species 3, 5, 6, and 7 are products of species 1, 2, and 4 hence H_2O is stored as a 1 in the first row indicating that it contains hydrogen, H_2, and a 2 in the second row indicating that it contains oxygen, O_2. It is important that the numerical key for the species appear in its proper column but it may be placed at random in the column. These values are input so that all of the first species column (even zeros) are input before the remaining species columns are input in order.

The D1(I) and G1(I) are tied to the species numerical key described in the NALPHP and NALPHPP arrays. From this D1(1) is the mass fraction of the element hydrogen, H, in the outer mixing region. D1(2) is the mass fraction of oxygen, O, in the outer mixing region. The last D1(I) value is the mass fraction of nitrogen, N_2. The G1(I) values use the same numerical keys and indicate the mass fraction of the elements in the inner mixing region.

LISTING OF TWO-STREAM-MIXING COMPUTER PROGRAM

```
                                        3200  FORTRAN  (2,1,0)/(RIS)        /  /
      TRANSFER MATRIX
C     A0=CONSTANT COEFFICIENT OF THE POLYNOMIAL
C     PLYC=COEFFICIENT OF POLYNOMIAL(PSIBAR)
C     AACOL(I,J,K)=CONSTANTS USED IN FINDING HOT AND COL.
C     AAHO (I,J,K)=LOWER TEMPERATURE RANGE FOR K=2 AND
C     AACO (I,J,K)=UPPER TEMPERATURE RANGE FOR K=1 AND
C     FOR I=1,...,6 AND J=INDEX FOR SPECIE K.
C     AAHO(I,J) WHERE J=1 OR 2 WITH J=2 IS FOR
C     THE LOWER TEMPERATURE RANGE FOR NITROGEN.
C     TTAB=TABLE OF T
C     NTEMP. OF VALUES IN EACH SPECIE TABLE
C     STAB(I,J) ARE TABLES OF SPECIE VALUES
C     CORRESPONDING TO SPECIE J WHERE I=1,...,KT
C     T IS VALUE OF INPUT FOR TABLE LOOK-UP
C     ALPHR,NLEFRR,NAA,NAA2,NSMAX,NF AND NP ARE USED
C     TO SOLVE INPUT EQUATIONS
C     DT=INCREMENT FOR INTEGRATION
C     DELTAT=INCREMENT FOR T ITERATION
C     D1(I) AND G1(I) FOR EQUATION I+NRO+1 AND CONTROL
C     XBAR(I) IS SOLUTION OF NON-LINEAR SYSTEM OF EQUATIONS
C     SMASS(I) IS CONCENTRATION OR MASS OF SPECIES I
C     CNITRO IS USED IN NRC+1 EQUATION
C     CNITRM=MASS OF NITROGE
C     ANEW = 1 CAUSES NEW CASE TO BE READ.
C     ANEW IS SET IN NEWTON.
C     XG = 1 IMPLIES THAT NO XG GUESSES ARE AFTER CASE.
C     XG = 0 IMPLIES THAT XG GUESSES ARE AFTER CASE.
C     XGMIN IS THE SMALLEST VALUE THAT
C     XG CAN TAKE IN NEWTON.(1.E-5 VALUE)
C
      COMMON XIO,IN,IO,ABX
      DIMENSION TTAB(20),STAB(20,20)
      DIMENSION ASPEC(4,20)
      DIMENSION PLYC(20),KELEM(20),AACO(6,20,2),AAHO(6,2)
      DIMENSION AA(10,20),AA2(10,20),NP(15),NNP(10)
      DIMENSION ALPHR(10,20),NLEFRR(10,20)
      DIMENSION KKCL(10),KKPL(10),TAH(20),ARRAY(21,22)
      DIMENSION XG(20),CHAR(21)
      DIMENSION SMASS(20),D1(15),G1(15)
      DIMENSION FUNION(21),DELX(21),FARRAY(21,22)
      DIMENSION SSTAR(20,20)
      EQUIVALENCE (FDOT,F),(FDOTT,F2)
C
C     READ INPUT
C
C
      IN=60
      IO=61
      ABX=0
 1000 READ (IN,2)XT,NSMAX,NRO,NXG
      I1=1
      I10=10
      NRO1=NRO+1
      NRO2=NRO+2
      NSMAX1=NSMAX+1
      NSMAX2=NSMAX+2
      NRS=NSMAX1-NRO1
      NRS1=NRS+1
      NFA=0
C
```

```
      READ (IN,3)(B1(I),I=1, RS1)
      READ (IN,3)(C1(I),I=1, RS1)
      READ (IN,3)(SMABS(I),I=1,NSMAX)
      READ (IN,2)((NALPHP(I,J),I=1,NRO),J=1,NSMAX)
      READ (IN,2)((NLPHPP(I,J),I=1,NRO),J=1,NSMAX)
      READ (IN,3)PSI,DPSI,LPSBAR,XI,DXI,PSIMAX,XIMAX,IREF,PREF,REFV,
     1PF,HO,ADCOEF,PSIO,T,TEMTAI,DT,TOL,PSHAR,UO,UE,USTARJ,RHOO,HO,
     2CONTIN
      READ (IN,3)((AACON(I,J,1),I=1,6),J=1,NSMAX)
      READ (IN,3)((AACON(I,J,2),I=1,6),J=1,NSMAX)
      READ (IN,3)((AAHOT(I,J),I=1,6),J=1,2)
      READ (IN,3)AL,PLYC
      READ (IN,3)(TIAB(K),K=1,KT)
      READ (IN,3)((STAH(I,J),I=1,KT),J=1,NSMAX)
      READ (IN,2)((NSPEC(I,J),I=1,4),J=1,NSMAX)
 1002 READ (IN,3)(XG(I),I=1,NSMAX)
      READ (IN,3)XGMIN
      READ(1 ,3)SUM
    3 FORMAT ((F12.5)
    2 FORMAT (12I5)
C
C     SET UP NAA AND NAA2 ARRAYS FROM NALPHP AND NALPHPP
C
      DO 470 I=1,NRO
      DO 470 J=1,NSMAX
      NAA(I,J)=0
      NAA2(I,J)=0
  470 CONTINUE
      DO 450 I=1,NRO
      DO 451 J=1,NSMAX
      IF ( NALPHP(I,J))452,453,452
  452 NAA(I,J)=J
  453 IF ( NLPHPP(I,J))454,451,454
  454 NAA2(I,J)=J
  451 CONTINUE
  450 CONTINUE
C
C     SET UP NF AND NNP ARRAYS FROM NALPHP AND NALPHPP
C
      K=1
      J=0
  456 J=J+1
      DO 455 I=1,NRO
      IF ( NALPHP(I,J))455,455,458
  458 NF(K)=J
      K=K+1
      IF (K-RS)456,456,459
  455 CONTINUE
      IF (K-RS)456,456,459
  459 CONTINUE
      K=1
      J=0
  466 J=J+1
      DO 465 I=1,NRO
      IF ( NLPHPP(I,J))465,465,468
  468 NNP(K)=J
      K=K+1
      IF (K-RO)466,466,469
  465 CONTINUE
      IF (K-RO)466,466,469
  469 CONTINUE
C
```

```
C     PRINT INPUT
C
      WRITE (10,91)NI,NSMAX, M
   91 FORMAT (1H1,///,28H SE  MIXING   JOB NO. 560180//,
     151H  NUMBER OF POINTS IN I. TABLES =I3,4X,
     220H NUMBER OF SPECIES =I3,4X,
     327H NUMBER OF INPUT EQUATIONS =I3/)
      WRITE (10,92)(G1(I),I=1,NS1)
   92 FORMAT (6H G1(I)/,(5E22.9))
      WRITE (10,86)(G1(I),I=1,NS1)
   86 FORMAT (6H G1(I)/,(5E22.9))
      WRITE (10,94)(SMASS(I),I=1,NSMAX)
   94 FORMAT (18H MASS OF SPECIES I/,(5E22.9))
      WRITE (10,95)(XG(I),I=1,NSMAX)
   95 FORMAT (13H CHAR GLESS#S/,(5E22.9))
      WRITE (10,352)XGMIN
  352 FORMAT (/,17H XG MIN I.E. M VALUE,5X,E17.9//)
      WRITE (10,96)((VALPHF(I,J),I=1,NMO),J=1,NSMAX)
   96 FORMAT (//,7H VALPHF/,(10I10))
      WRITE (10,97)((VLPHFF(I,J),I=1,NMO),J=1,NSMAX)
   97 FORMAT (8H VALPHF/,(10I10))
      WRITE (10,98)((VAA(I,J),I=1,NMO),J=1,NSMAX)
   98 FORMAT (9H VAA(I,J)/,(10I10))
      WRITE (10,99)((VAA2(I,J),I=1,NMO),J=1,NSMAX)
   99 FORMAT (10H VAA2(I,J)/,(10I10))
      WRITE (10,80)(VF(I),I=1,NMS)
   80 FORMAT (6H VF(I)/,(10I10))
      WRITE (10,81)(VVF(I),I=1,NMO)
   81 FORMAT (7H VVF(I)/,(10I10))
      WRITE (10,82)PSI,DPSI,PSIBAR,XI,DXI,PSIMAX,XIMAX,
     1IMPP,MPEP,MPEV,MF,MO,ACCTOL,PSIU,DELTAT,DT,TOL,
     2PSIAV,HU,DF,DSIABU,RHOU,RU,CPINP
   82 FORMAT (//,6H PSI  E17.9,4X,5H DPSIE17.9,2X,7HPSIBARE17.9,
     14X,5HXI   E17.9,4X,5HDXI  E17.9/,6H PSIMXE17.9,4X,
     25HXI MXE17.9,4X,5HIMPP E17.9,4X,5HMPEP E17.9,4X,
     35HMPEV E17.9/,6H MF   E17.9,4X,5HMO   E17.9,3X,
     46HACCTOLE17.9,4X,5HPSIU E17.9,4X,5HT    E17.9/,
     56H DELTAT17.9,4X,5HDT   E17.9,4X,5H TOL  E17.9,
     13X,6HPSIBARE17.9,4X,5HHU   E17.9/,6H UF   E17.9,
     73X,6HDSIABE17.9,4X,5HRHOU E17.9,4X,5HRU   E17.9,3X,
     6HCPINPE17.9/)
C
      DO 477 K=1,2
      DO 431 J=1,NSMAX
      WRITE (10,83)J,(AACOR(I,J,K),I=1,6)
   83 FORMAT (26X,15HAACOR(I,J,K) J=I2/,(5E22.9))
  431 CONTINUE
      IF (K-2)473,472,472
  473 WRITE (10,476)
  476 FORMAT (//,19X,35H AAC N ARRAY FOR LOWER TEMPERATURES/)
  477 CONTINUE
  472 DO 1 I=1,6
      WRITE (10,489)(AAHCI(I,1),I=1,6)
  489 FORMAT (/,20X,5HAAHCI/,(5E22.9))
      WRITE (10,475)(AAHCI(I,2),I=1,6)
  475 FORMAT (/,20X,28HAAHCI FOR LOWER TEMPERATURES/,(5E22.9))
      WRITE (10,84)AO,PLYC
   84 FORMAT (//,19X,13H AO,A1,...,AN/,(5E22.9))
C
C     PRINT TABLES
C
      WRITE (10,193)(TAB(K),K=1,KT)
```

```
  193 FORMAT (///,20X,7HT TABLE//,(5E22.9))
      WRITE (IO,444)
  444 FORMAT (//,20X,32HSPECIE TABLES WITH X=LOG(NUMBER)/)
      DO 200 I=1,NSMAX
      WRITE (IO,194)I
  194 FORMAT (20X,9HSPECIES =I3,6H TABLE)
      WRITE (IO,195)(STAB(K,I),K=I1,KT)
  195 FORMAT (5E22.9)
  200 CONTINUE
C
C     SETTING KELEM(I)=I
C
      DO 5 K=1,20
      KELEM(K)=K
    5 CONTINUE
C
C     CO VERT S1TAB,...,S20TAB TABLES FROM X=LOG(Y) TO Y VALUES
      IF(NFA)100,350,100
  100 IF(NVG)350,350,351
  350 NFA=1
      DO 255 I=1,KT
      DO 255 J=1,NSMAX
  255 SSTAB(I,J)=STAB(I,J)
C
  351 DO 260 J=1,NSMAX
      DO 260 I=1,KT
      IF (STAB(I,J))493,491,491
  491 STAB(I,J)=10.0**STAB(I,J)
      GO TO 490
  493 ST=STAB(I,J)
      NST=ST
      FNST=NST
      FEST=ABS(ST)-ABS(FNST)
      STAB(I,J)=10.0**FEST*10.0**FNST
  490 CONTINUE
  260 CONTINUE
C
      WRITE (IO,445)
  445 FORMAT (//,20X,27HSPECIE TABLES WITH X=NUMBER/)
C
      DO 401 J=1,NSMAX
      WRITE(IO,402)J,(STAB(I,J),I=1,KT)
  402 FORMAT (20X,13HSPECIE TABLE I2/,(5E22.9))
  401 CONTINUE
      WRITE (IO,219)
  219 FORMAT (//,20X,24HKSPEC(I,J) WHERE J=INDEX//)
      DO 89 J=1,NSMAX
      WRITE (IO,89)(KSPEC(I,J),I=1,4)
   89 CONTINUE
   89 FORMAT (3X,4I5)
C
C
C     PREDECK VALUES
C
      PPSI=0
      TOLD=TOL
      PSBARD=PSBAR
      PSIO=PSI
      XIO=XI
      TO=300.
      R=1.987
C
```

```
C     900 IS THE RETURN POINT FOR PSI AND
C     XI CHANGE. FOR EACH XI, PSI GOES FROM
C     PSI0 TO PSIMAX.  THEN XI=XI+PXI AND THE
C     PROCESS CONTINUES UNTIL XI=XIMAX.
C
C
  900 L=0
      IE =1
      IS =0
      Y=0.
      YP=0.0
      AY=0.0
      AYP=0.0
      SUM1=0.0
      SUM2=0.0
      KC1=0
      IT1=0.
      IT2=0.
      SM1=0.
      SM2=0.
      PSBAR=PSBAR0
      PI =(1.+.5*PSBAR)/PSBAR
      CONTINUE
      IF ( )50,30,31
   30 CALL MAIN1 (PSI,XI,PSI0,FDOT,PSBAR,-PLYC,A0,PCL)
   31 I=1+
      DO 11 KKK=1,NINC
      I. =1
   15 CALL RKKUT(NEQ,NSE,DPSBAR,PSBAR,F,Y,YP,FK,AY,AYP,SUM1,SUM2,I )
      CALL MAIN1 (PSI,XI,PSI ,FDOT,PSBAR,PLYC,A0,PCL)
      INC=INC+1
      IF(ABS(F).LE.1.0E-36)20,21
   20 Y=AY
      FK=0.
   21 IF(I.D-4)15,15,16
   16 CONTINUE
C     ANY DESIRED CODING
   11 CONTINUE
      X1=PSI0/(2.*XI)
      FF=X1*Y
      IF(FF-1.)7,7,6
    6 FF=1.
    7 USTAR=1.-FF*(1.0-U0/UE)
      F=USTAR*UE
      CONTROU=[1(WMS1)*(USTAR-U0/UE)/(1.-U0/UE)+
     1 1(WMS1)*(1.-USTAR)/(1.-U0/UE)
      CONTROU=CONTRO/CONTRO
C
C
      WRITE (IP,432)PSI,XI,FF,USTAR,U,CONTROU,Y
  432 FORMAT (//,6H PSI  E17.9,4X,5HXI   E17.9,3X,
     13HAND VALUES AFTER FIRST INTEGRATION//,
     16H FF    E17.9,4X,6HUSTAR E17.9,4X,
     26HU      E17.9,4X,6HCONTROE17.9,4X,
     66H Y     E17.9/)
C
C
C     POINT OF RETURN FOR F EVALUATION
C
  500 CONTINUE
C
C     SAVE FOR PROC VALUE EACH EQUATION
```

```
C
      DO 710 I=1,20
      TAB(I)=0.0
  710 CONTINUE
      DO 777 I=1,NRO
      Z14=0.0
      Z13=0.0
      NN = NF(I)
      INDEX=NAA2(I,NN)
      DO 711 II=1,KT
  711 TAB(II)=SSTAB(II,INDEX)
      CALL LINT (T,TTAB,TAB,KT,2,YI)
      IF (YI)252,253,253
  253 YI=10.0**YI
      GO TO 254
  252 ST=YI
      NST=ST
      FNST=NST
      FFST=ABS(ST)-ABS(FNST)
      YI=10.**FFST*10.0**FNST
  254 CONTINUE
      FKPL(I)=YI
      DO 706 J=I1,NSMAX
      Z16= ALPHF(I,J)
      Z13=Z13+Z16
      Z15= LPHFF(I,J)
  706 Z14=Z14+Z15
      Z15=Z13-Z14
      FKPL(I)=FKPL(I)*(82.05*T)**Z15
  777 CONTINUE
C
C
      WRITE (IO,421)(FKCL(I),I=1,NRO)
  421 FORMAT (20X,7HFKCL(I)/,(5E22.9))
      WRITE (IO,422)(FKPL(I),I=1,NRO)
  422 FORMAT (20X,7HFKPL(I)/,(5E22.9))
      WRITE (IO,447)
  447 FORMAT (//)
C
C     ZERO ARRAY(I,J)
C
      DO 230 J=1,21
      DO 230 I=1,22
      ARRAY(J,I)=0.0
  230 CONTINUE
C
C     SET-OF EQUATIONS 1,...,NRO IN ARRAY(I,J)
C     FIRST FILL IN ARRAY(I, SMAX2) AND
C     ARRAY(I,NSMAX1)
C
      DO 310 I=1,NRO
      Z1=0.0
      Z2=0.0
      DO 311 J=I1,NSMAX
      Z2=NALPHF(I,J)+Z2
      Z1=NALPHF(I,J)+Z1
  311 CONTINUE
      ARRAY(I,NSMAX2)=Z1-Z2
  310 CONTINUE
      DO 312 I=1,NRO
      Z1=0.0
      DO 313 J=1,NSMAX
```

```
      SMJ=SMASS(J)
      Z1=Z1+(NLPHPP(I,J)-NALPHP(I,J))*ALOG(SMJ)
  313 CONTINUE
      FKCLJ=FKCL(I)
      ARRAY(I,NSMAX1)=-Z1-ALOG(FKCLJ)
  312 CONTINUE
C
C     FILL ARRAY(I,J),I=1,...,NRO AND J=1,...,NSMAX
C
      DO 314 I=1,NRO
      DO 314 J=1,NSMAX
      ARRAY(I,J)=NLPHPP(I,J)-NALPHP(I,J)
  314 CONTINUE
C
C     SET-UP ARRAY(I,J),I=NRO+1,J=1,...,NSMAX
C
      CONP=FREF*TREF*REFV/T
      DO 315 I=1,NSMAX
      ARRAY(NRO1,I)=1./(CONP*SMASS(I))
  315 CONTINUE
      ARRAY(NRO1,NSMAX1)=CNTRO/CONP
      ARRAY(NRO1,NSMAX2)=-1.
C
C     SET-UP S-R EQUATIONS
C
      DO 347 KKN=1,NRS
      KKNP=NF(KKN)
      K14=KKN+NRO1
 347  ARRAY(K14,KKNP)=1.0
C
      DO 1260 KNCT=1,NRS
      KKNP=NF(KNCT)
      DO 250 I=1,NRO
      DO 251 J=1,NSMAX
      IF (NAA2(I,J))1113,1113,1112
 1113 GO TO 251
 1112 IDEX=NAA2(I,J)
      II=1
 1114 IF (KKNP-KSPEC(II,IDEX))1116,1115,1116
 1116 II=II+1
      IF (II-4)1114,1114,251
 1115 Z2=NALPHP(I,KKNP)
      Z1=NLPHPP(I,IDEX)
      K15=KNCT+NRO1
      ARRAY(K15,IDEX)=(Z2*SMASS(KKNP))/(Z1*SMASS(IDEX))
  251 CONTINUE
  250 CONTINUE
 1260 CONTINUE
C
C     STORE K(NRO+1),...,K(SMAX) INTO ARRAY(I,SMAX+1)
C
      DO 271 I=NRO2,NSMAX1
      J=I-NRO1
      ARRAY(I,NSMAX1)=-D1(J)*(USTAR-UJ/UE)/(1.-UJ/UE)-
     1D1(J)*(1.-USTAR)/(1.-UJ/UE)
  271 CONTINUE
C
C
      DO 403 I=1,NSMAX1
      WRITE (IO,404)I,(ARRAY(I,J),J=1,NSMAX2)
  404 FORMAT (20X,4HROW I2,14H OF ARRAY(I,J)/,(5F22.9))
  403 CONTINUE
```

```
      WRITE (IO,447)
      TOL=TOLO
      KNEW = 0
      CALL NEWTOL (CHAR,ARRAY,FUNION,MARRAY,DELX,NSMAX,NRO,TOL,T,
     1KNEW,XOMIN,SMASS,CNITRM,REFV,PREF,TREF,XG)
      IF (KNEW)1001,1001,1004
 1004 IF(NXG)1000,1000,600
  600 DO 93 I=1,NT
      DO 93 J=1,NSMAX
      STAB(I,J)=SSTAB(I,J)
   93 CONTINUE
      GO TO 1002
 1001 CONTINUE
C
      WRITE (IO,446)(CHAR(I),I=1,NSMAX1)
  446 FORMAT (/,20X,7HCBAR(I)/,(5E22.9))
C
C     CHECK THE ITERATION THROUGH THE ENTHALPY
C
      Z1=0.0
      Z2=0.0
      CBH=CBAR(NSMAX1)
      H=(HO*(USTAR-UJ/UE)+HJ*(1.-USTAR))/(1.-UJ/UE)
      HP=H-1./2.*USTAR**2*UE**2*2.3901E-8
      CBAR(NSMAX1)=CNITRC*CNITRM
C
C     FIRST FIND INTEGRATED PART OF SMALL H
C
      NEQ=1
      Z1=0.0
      Z2=0.0
      NSO=0
      DO 150 N=1,NSMAX1
      YP=0.0
      FK=0.0
      AY=0.0
      SUM1=0.0
      SUM2=0.0
      L=0
      Y2=0.
      TT=TO
      IF (L)130,130,131
  130 CALL MAIN2(FDOTT,SMASS,COP,R,AACON,TT,N,AAHUT,NSMAX1,CNITRM)
  131 L=1+L
  650 DO 111 KKK=N,N
      IND=1
  115 CALL RONKUT(NEQ,NSO,DT,TT,F2,Y2,YP,FK,AY,AYP,SUM1,SUM2,IND)
      CALL MAIN2(FDOTT,SMASS,COP,R,AACON,TT,N,AAHUT,NSMAX1,CNITRM)
      IND=IND+1
      IF (IND-4)115,115,116
  116 CONTINUE
  111 CONTINUE
      Y22=Y21
      Y21=Y2
      TT2=TT1
      TT1=TT
      IF (ABS(T-TT)-.0000001)161,161,162
  162 IF (TT-T)650,650,501
C
C     2 POINT INTERPOLATION IN Y2
C
  501 Y2=Y22+((T-TT2)/(TT1-TT2))*(Y21-Y22)
```

```
  161 Z1=Y2*CBAR(N)+Z1
      CALL ACAL2(HOT,N,AAHCT,TO,N,AACON,NSMAX1)
      IF (N-NSMAX1)480,479,479
  480 Z2=CBAR(N)*HOT/SMASS(N)+Z2
      GO TO 478
  479 Z2=CBAR(N)*HOT/CNITRM+Z2
  478 CONTINUE
      TT=T
      WRITE (IO,499)COP,FOT,Y2,TT,FDOTT,N
  499 FORMAT (/,5H CO-PE17.9,4X,5H HO-TE17.9,4X,5H Y2  E17.9,
     14X,5H TT  E17.9/,5H FDOTE17.9,4X,5H N    I5/)
  150 CONTINUE
      SH=Z1+Z2
      WRITE (IO,433)SH,HP
  433 FORMAT (28H VALUES USED IN THE ENTHALPY,4X,
     15HSH    E17.9,4X,5HHP    E17.9///)
C
C     KC1 IS USED TO DETERMINE CUT-OFF APPROACH FORM
C     ABOVE OF BELOW THE H-PRIME VALUE.
C     SH2,SH1,TC2,TC1 AND HP ARE USED IN INTERPOLATION
C     ON T WHICH GIVES SH=HP.
C
      IF(ABS(SH2-SH)-.01)35,35,2197
   35 IF(SH2)2198,2198,2199
 2198 IF(ABS(SH2)-ABS(HP))821,812,812
 2199 IF(ABS(SH2)-ABS(HP))812,812,821
 2197 SH2=SH1
      SH1=SH
      TC2=TC1
      TC1=T
      IF (KC1-3)840,850,840
  840 IF (ABS(HP-SH)-ACCTOL)850,850,1197
 1197 IF(HP)400,400,498
  400 IF(SH)841,841,812
  498 IF(SH)821,841,1199
 1199 IF(HP-SH)822,850,811
  841 IF (ABS(HP)-ABS(SH))801,850,802
  801 KC1=1
      IF (ABS(HP-SH)-ACCTOL)850,850,810
  810 IF (ABS(HP)-ABS(SH))811,850,812
  811 T=T+DELTAT
      GO TO 500
  812 T=TC2+(ABS(HP-SH2))/(ABS(SH1-SH2))*ABS(TC1-TC2)
      KC1=3
      GO TO 500
  802 KC1=2
      IF (ABS(HP-SH)-ACCTOL)850,850,820
  820 IF (ABS(HP)-ABS(SH))821,850,822
  821 T=TC1+(ABS(HP-SH1))/(ABS(SH2-SH1))*ABS(TC2-TC1)
      KC1=3
      GO TO 500
  822 T=T-DELTAT
      GO TO 500
  850 CONTINUE
C     CALCULATIONS FOR PRINT-OUT
      Z12=0.0
      RHSTAR=CPB/RHOJ
      IF (KPSI)610,610,611
  610 KSPI1=0.0
      KSPI2=(2.*KSI)/((CPB/RHOJ)*(P/CO))
      KSPI=1
  611 CONTINUE
```

```
      FSPI1=FSPI2
      FSPI2=(2.*PSI)/((CBB/RHOO)*(U/C0))
      DFY=FSPI2-FSPI1
      SUM=SUM+FPSI*(FSPI1+1./2.*DFY+(1./3.-1./2.)*1./2.*
     1DFY**2+(1./4.)*DFY**3/FACT(3)+(1./5.-3./2.+11./3.-3.)*
     2DFY**4/FACT(4)+(1./6.-2.+35./4.-50./3.+12.)*DFY**5
     3/FACT(5)+(1./7.-15./6.+17.-225./4.+274./3.-60.)*
     4DFY**6/FACT(6))
      RSTARS=SUM
      RSTAR=SQRT(RSTARS)
      CALL CAL (XI,XSTARS)
      XSTAR=SQRT(XSTARS)
      RRR=RSTAR*RO
      X=XSTAR*RO
C
C
C     FINAL PRINT STATEMENT
C
      WRITE (IO,300)T,H,SH,HH,RHSTAR,RSTARS,XSTARS,RRR,X
  300 FORMAT (/,20X,28HFINAL PRINT STATEMENT FOR I=E17.9//,
     17H H      E17.9,4X,6HSH     E17.9,4X,
     26HHH INE E17.9,4X,6HRHSTAR E17.9/,
     37H RSTARSE17.9,4X,6HXSTARSE17.9,4X,
     46HRRR     E17.9,4X,6HX      E17.9//)
C
C
      IF (ABS(PSI-PSIMAX)-.0000001)1195,1195,2196
 2196 IF (PSI-PSIMAX)1196,1195,1195
 1196 PSI=PSI+DPSI
      DO 50 I=I1,NSMAX
      XG(I)=CBAR(I)
   50 CONTINUE
      WRITE(IO,95)(XG(I),I=I1,NSMAX)
      GO TO 900
 1195 IF (ABS(XI-XIMAX)-.0000001)1000,1000,2194
 2194 IF (XI-XIMAX)1194,1000,1000
 1194 PSI=PSIO
      KPSI=0
      SUM=0.0
      XI=XI+DXI
      GO TO 900
      END

      3200 FORTRAN DIAGNOSTIC RESULTS - FOR   JETMIX

NO ERRORS
```

```
      SUBROUTINE NEWTON (CBAR,ARRAY,FUNION,PARRAY,DELX,NSMAX,NRO,TOL,
     1T,KNEW,XGMIN,SMASS,CNTRO,REFV,PREF,TREF,XG)
C     CBAR(21) ARE APPROXIMATE ANSWER WHICH
C     ARE TO BE USED FOR THE NEWTON METHOD
C     ARRAY(21,22) USED IN EVALUATING THE GIVEN FUNCTIONS
C     FUNCTION(21) IS THE VALUE OF FUNCTIONS
C     PARRAY(21,22) IS AN ARRAY OF PARTIALS USED
C     IN NEWTONS METHOD
C     DELX(21) IS DELTA XS USED TO SET-UP
C     THE NEW CBAR VALUES
C     CNTRO IS USED IN NRO+1 EQUATION
C     SMASS(I) IS CONCENTRATION OR MASS OF SPECIES I
C     XG(I) IS THE INITIAL GUESS FOR CBAR(I)
C     XGMIN IS THE SMALLEST VALUE THAT
C     XG CAN TAKE IN NEWTON.(INPUT VALUE)
      COMMON XIO,IN,IO,ABX
      DIMENSION CBAR(21),ARRAY(21,22),FUNION(21),
     1PARRAY(21,22),DELX(21),XX(21),XX1(21)
      DIMENSION EXPAX(21),SMASS(20),XG(20),DELXI(21)
      SUM1=0.0
      NSMAX1=NSMAX+1
      NSMAX2=NSMAX+2
      NRO1=NRO+1
      NRO2=NRO+2
      TOLNEW=TOL*.001
      Z1=CNTRO
      Z2=PREF*TREF*REFV/T
      DO 222 I=1,NSMAX
      Z1=XG(I)/SMASS(I)+Z1
  222 CONTINUE
      RHOVOE=Z2/Z1
      CBAR(NSMAX1)=RHOVOE
      DO 115 I=1,NSMAX
      CBAR(I)=XG(I)
  115 CONTINUE
C     SET XX(I)=CBAR(I)
      DO 1 I=1,NSMAX1
      XX(I)=CBAR(I)
    1 CONTINUE
      DO 116 I=1,NSMAX1
      IF (XX(I)-XGMIN)117,117,119
  119 XX(I)=ALOG(XX(I))
      GO TO 116
  117 XX(I)=XGMIN
      XX(I)=ALOG(XX(I))
  116 CONTINUE
 1000 WRITE(IO,100)(XX(I),I=1,NSMAX1)
  100 FORMAT(10H XX VALUES/,(5E22.9))
      DO 41 I=1,NSMAX1
      EXPXX(I)=EXP(XX(I))
      IF (EXPXX(I)-10.)41,41,902
   41 CONTINUE
C     FIND THE VALUE OF FUNION(I),I=1,...,NRO
      DO 2 I=1,NRO
      Z1=0.0
      DO 3 J=1,NSMAX
      Z1=Z1+ARRAY(I,J)*XX(J)
    3 CONTINUE
      FUNION(I)=Z1+ARRAY(I,NSMAX1)+ARRAY(I,NSMAX2)*XX(NSMAX1)
    2 CONTINUE
```

```
C     FIND THE VALUE OF FUNCTION(I),I=NRO+1
      Z1=0.0
      DO 4 J=1,NSMAX
      Z1=Z1+ARRAY(NRO1,J)*EXPXX(J)
    4 CONTINUE
      FUNCTION(NRO1)=Z1+ARRAY(NRO1,NSMAX1)-EXP(-XX(NSMAX1))
C     FIND THE VALUE OF FUNCTION(I),I=NRO+2,...,NSMAX+1
      DO 5 I=NRO2,NSMAX1
      Z1=0.0
      DO 6 J=1,NSMAX
      Z1=Z1+ARRAY(I,J)*EXPXX(J)
    6 CONTINUE
      FUNCTION(I)=Z1+ARRAY(I,NSMAX1)
    5 CONTINUE
C     STORE FUNCTION VALUES IN PARRAY
      DO 35 I=1,NSMAX1
   35 PARRAY(I,NSMAX2)=-FUNCTION(I)
      IF(ABX)26,25,26
   25 WRITE (IO,27)(FUNCTION(I),I=1,NSMAX1)
   27 FORMAT (11H FUNCTION 1/,(5E22.9))
   26 CONTINUE
C     PARTIALS OF FUNCTIONS
      DO 7 I=1,NRO
      DO 8 J=1,NSMAX
      PARRAY(I,J)=ARRAY(I,J)
    8 CONTINUE
      PARRAY(I,NSMAX1)=ARRAY(I,NSMAX2)
    7 CONTINUE
      DO 9 J=1,NSMAX
      PARRAY(NRO1,J)=ARRAY(NRO1,J)*EXPXX(J)
    9 CONTINUE
      PARRAY(NRO1,NSMAX1)=EXP(-XX(NSMAX1))
      DO 10 I=NRO2,NSMAX1
      DO 11 J=1,NSMAX
      PARRAY(I,J)=ARRAY(I,J)*EXPXX(J)
   11 CONTINUE
      PARRAY(I,NSMAX1)=0.0
   10 CONTINUE
      IF (SOL1)52,52,54
   52 DO 50 I=1,NSMAX1
      Z1=0.
      DO 51 J=1,NSMAX1
      Z1=Z1+ABS(PARRAY(I,J))
   51 CONTINUE
      IF (Z1-1.0)50,50,54
   50 CONTINUE
      WRITE (IO,55)
   55 FORMAT (20X,26H*MATRIX IMPLIES CONVERGENCE//)
      SOL1=1.0
   54 CONTINUE
      IF(ABX)31,30,31
   30 WRITE (IO,32)((PARRAY(I,J),I=1,NSMAX1),J=1,NSMAX2)
   32 FORMAT (14H PARTIAL ARRAY/,(5E22.9))
   31 CONTINUE
      CALL SIMEQ(PARRAY,NSMAX1,DELX)
      DO 12 I=1,NSMAX1
      XX1(I)=XX(I)+DELX(I)
   12 CONTINUE
      IF(ABX)14,13,14
   13 WRITE (IO,15)(FUNCTION(I),I=1,NSMAX1)
      WRITE (IO,16)(DELX(I),I=1,NSMAX1)
      WRITE(IO,21)(XX1(I),I=1,NSMAX1)
```

```
   15 FORMAT (10H XX VALUES/,(5E22.9))
   16 FORMAT (11H DELTA X(I)/,(5E22.9))
   21 FORMAT(11H XX1 VALUES/,(5E22.9))
   14 CONTINUE
      DO 18 I=1, NSMAX1
      DELXI(I)=EXP(DELX(I))
      IF(ABS(DELXI(I)-1.)-TO NEW)18,18,20
   18 CONTINUE
      DO 22 I=1,NSMAX1
      CBAR(I)=EXP(XX1(I))
   22 CONTINUE
      DO 301 I=1,NSMAX
      XG(I)=CBAR(I)
  301 CONTINUE
      RETURN
   20 DO 60 I=1,NSMAX1
      XX(I)=XX1(I)
   60 CONTINUE
      GO TO 1000
  902 WRITE(IC,903)
  903 FORMAT (14H CASE DIVERGES//)
      KNEW = 1
      RETURN
      END

        3200 FORTRAN DIAGNOS IC RESULTS - FOR    EFT

NO ERRORS
```

```
      SUBROUTINE FIC (PXI,PPSIO,PPSI,ANS,SPSBAR)
      FORMAT X10
C     X IS THE POINT OF EVALUATION
C     ANS IS THE ANSWER DESIRED
C     PPSI AND PPSI AND PXI ARE NEEDED VALUES
C     FOR THE CALCULATIONS
      X10=(PPSI*SPSBAR/PPSIO)/(2.0*PXI/PPSIO)
      IF(X10-20.)5,5,2
    5 N=1
      ANS=X10**2/4.0+1.0
      ANS1=ANS
   10 N=N+1
      ANS=ANS+(X10**N/(2.0**N*FACT(N)))**2
      ANS2=ANS1
      ANS1=ANS
      IF(N-200)4,4,2
    4 IF (ABS(ANS1-ANS2)-.00000001)2,2,3
    3 GO TO 10
    2 RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR FIC

NO ERRORS

```
                                    3200  FORTRAN  (2.1.0)/(RTS)
      SUBROUTINE MAIN1(SPSI,SXI,SPSIJ,SFOUT,SPSBAR,SA,SAO,SPOL)
      COMMON XIO
C     SFOUT IS THE DOT EVALUATION TO BE INTEGRATED
      DIMENSION SA(20)
      CALL FIG(SXI,SPSIJ,SPSI,ANSWER,SPSBAR)
      IF(XIO-20.)2,2,4
    2 Z1=(SPSBAR**2/(4.0*SXI))/SPSIJ
      IF(Z1-700.)5,5,3
    3 Z2=0.
      GO TO 7
    5 Z2=EXP(-Z1)
    7 Z3=(SPSI**2/(SPSIJ**2))/((4.*SXI)/SPSIJ)
      IF(Z3-700.)9,9,8
    8 Z4=0.
      GO TO 10
    9 Z4=EXP(-Z3)
   10 CALL POLVAL(SAO,SA,SPOL,SPSBAR)
      SFOUT=Z2*Z4*SPOL*SPSBAR*ANSWER
      GO TO 6
    4 Z1=((-(SPSI**2)-(SPSBAR**2*SPSIJ**2)+2.*SPSI*SPSIJ
     1*SPSBAR)/(4.*SXI*SPSIJ))
      IF(Z1)11,12,12
   11 IF(Z1+700.)13,13,12
   12 Z2=EXP(Z1)
      GO TO 14
   13 Z2=0.
   14 CALL POLVAL(SAO,SA,SPOL,SPSBAR)
      EM=SQRT((2.*3.14159265*SPSI*SPSBAR)/(2.*SXI))
      SFOUT=(Z2*SPOL*SPSBAR)/EM
    6 RETURN
      END

      3200 FORTRAN DIAGNOSTIC RESULTS - FOR   MAIN1

NO ERRORS
```

```
                                    3200  FORTRAN  (2.1.0)/(RTS)
      SUBROUTINE POLVAL(AAC,A,YY,PPSBAR)
C     AAC IS THE CONSTANT VALUE OF THE POLYNOMIAL
C     AA IS THE A(1)...A(20) COEFFICIENTS
C     YY IS THE VALUE OF THE POLYNOMIAL AT PPSIBAR
      DIMENSION A(20)
      YY=AAC+((((((((((((((((((((A(20)*PPSBAR+A(19))*PPSBAR+
     1A(18))*PPSBAR+A(17))*PPSBAR+A(16))*PPSBAR+A(15))*
     2PPSBAR+A(14))*PPSBAR+A(13))*PPSBAR+A(12))*
     3PPSBAR+A(11))*PPSBAR+A(10))*PPSBAR+A(9))*
     4PPSBAR+A(8))*PPSBAR+A(7))*PPSBAR+A(6))*
     5PPSBAR+A(5))*PPSBAR+A(4))*PPSBAR+A(3))*
     6PPSBAR+A(2))*PPSBAR+A(1))*PPSBAR)
      RETURN
      END

        3200 FORTRAN DIAGNOSTIC RESULTS - FOR   POLVAL

NO ERRORS
```

```
      SUBROUTINE ACAL1(CCOP,RR,AAACON,TT,N,AAAHOT,NSMAX1)
      DIMENSION AAACON(6,20,2),AAAHOT(6,2)
      IF (TT-1000.)1,2,2
    2 IF (N-NSMAX1)11,22,22
   11 CCOP=(AAACON(1,N,1)+AAACON(2,N,1)*TT+AAACON(3,N,1)*TT**2+
     1AAACON(4,N,1)*TT**3+AAACON(5,N,1)*TT**4)*RR
      GO TO 3
   22 CCOP=(AAAHOT(1,1)+AAAHOT(2,1)*TT+AAAHOT(3,1)*TT**2+
     1AAAHOT(4,1)*TT**3+AAAHOT(5,1)*TT**4)*RR
    3 CONTINUE
      RETURN
    1 IF (N-NSMAX1)31,32,32
   31 CCOP=(AAACON(1,N,2)+AAACON(2,N,2)*TT+AAACON(3,N,2)*TT**2+
     1AAACON(4,N,2)*TT**3+AAACON(5,N,2)*TT**4)*RR
      GO TO 33
   32 CCOP=(AAAHOT(1,2)+AAAHOT(2,2)*TT+AAAHOT(3,2)*TT**2+
     1AAAHOT(4,2)*TT**3+AAAHOT(5,2)*TT**4)*RR
   33 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR ACAL1

NO ERRORS

```
      SUBROUTINE MAIN2(FDOTT,SMASS,COP,R,AACON,TT,N,AAHOT,NSMAX1,
     1CNITRM)
      DIMENSION AACON(6,20,2),SMASS(20),AAHOT(6,2)
      CALL ACAL1(COP,R,AACON,TT,N,AAHOT,NSMAX1)
      IF (N-NSMAX1)1,2,2
    1 FDOTT=COP/SMASS(N)
      GO TO 3
    2 FDOTT=COP/CNITRM
    3 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR MAIN2

NO ERRORS

```
      SUBROUTINE ACAL2(RHOT,RR,AAAHOT,TT,N,AAACON,NSMAX1)
      DIMENSION AAAHOT(6,2),AAACON(6,20,2)
      IF (TT-1000.)1,2,2
    1 K=2
      GO TO 4
    2 K=1
    4 IF (N-NSMAX1)5,6,6
    5 RHOT=(AAACON(1,N,K)+AAACON(2,N,K)*TT/2.+AAACON(3,N,K)*TT*TT/3.+
     1AAACON(4,N,K)*TT**3/4.+AAACON(5,N,K)*TT**4/5.+
     2AAACON(6,N,K)/TT)*RR*TT
      GO TO 7
    6 RHOT=(AAAHOT(1,K)+AAAHOT(2,K)*TT/2.+AAAHOT(3,K)*TT*TT/3.+
     1AAAHOT(4,K)*TT**3/4.+AAAHOT(5,K)*TT**4/5.+
     2AAAHOT(6,K)/TT)*RR*TT
    7 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR ACAL2

NO ERRORS

3200 FORTRAN (2.1.0)/(RTS) / /.

```
SUBROUTINE CAL(XI,ANSWER)
ANSWER =2.*XI/.00075
RETURN
END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR CAL

NO ERRORS

```
      FUNCTION FACT(I)
C     DETERMINES N-FACTORIAL FOR VN
      IF(I-1)1,1,2
    1 FACT=1.
      RETURN
    2 FACT=1.
      DO 3 J=2,I
      FJ=J
    3 FACT=FACT*FJ
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR FACT

NO ERRORS

```
      SUBROUTINE RUNKUT(NEQ,NSO,H,X,F,Y,YP,FK,AY,AYP,SUM1,SUM2,INDT)
C     RUNKUT**RUNGE KUTTA INTEGRATION
      DIMENSION F(100),Y(100),YP(100),AY(100),AYP(100),SUM1(100)
      DIMENSION SUM2(100),FK(100)
C     THIS INTEGRATES NEQ EQUATIONS,NSO OF WHICH ARE SECOND ORDER.
C     H IS THE STEP SIZE,X THE INDEPENDENT VARIABLE. THE DEPENDENT
C     VARIABLES ARE STORED IN Y AND YP,SECOND ORDER EQUATIONS FIRST.
C     F IS THE VALUE OF THE FUNCTION (FIRST OR SECOND DERIVATIVE.
C     AUXILIARY STORAGE MUST BE FURNISHED, FK(NEQ),AY(NEQ),AYP(NEQ),
C     SUM1(NSO),SUM2(NEQ). SUBROUTINE IS CALLED FOUR TIMES WITH VALUES
C     OF X,Y,YP,F. SUBROUTINE FURNISHES VALUES OF X,Y,YP. INDT IS
C     COUNTER (1,2,3 OR 4). SUBROUTINE STORES NO ESSENTIAL INFORMATION
C     INTERNALLY
      IF(INDT-2) 1,2,5
    1 X=X+H/2.
      AL1=H/2.
      AL2=0.
      AL3=.5
      AL4=1.
      DO 6 I=1,NEQ
      AY(I)=Y(I)
      AYP(I)=YP(I)
      SUM1(I)=0.
    6 SUM2(I)=0.
      GO TO 7
    2 AL1=H/2.
      AL2=H/4.
      AL3=.5
      AL4=2.
      GO TO 7
    5 IF (INDT-3) 3,3,4
    3 X=X+H/2.
      AL1=H
      AL2=H/2.
      AL3=1.
      AL4=2.
    7 DO 8 I=1,NEQ
      IF (I-NSO) 9,9,10
    9 Y(I)=AY(I)+AL1*AYP(I)+AL2*FK(I)
      FK(I)=H*F(I)
      YP(I)=AYP(I)+AL3*FK(I)
      SUM1(I)=SUM1(I)+FK(I)
      GO TO 11
   10 FK(I)=H*F(I)
      Y(I)=AY(I)+AL3*FK(I)
   11 SUM2(I)=SUM2(I)+AL4*FK(I)
    8 CONTINUE
      RETURN
    4 DO 12 I=1,NEQ
      FK(I)=H*F(I)
      SUM2(I)=(SUM2(I)+FK(I))/6.
      IF (I-NSO) 13,13,14
   13 Y(I)=AY(I)+H*AYP(I)+H* SUM1(I)/6.
      YP(I)=AYP(I)+SUM2(I)
      GO TO 12
   14 Y(I)=AY(I)+SUM2(I)
   12 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR RUNKUT

NO ERRORS

```
      SUBROUTINE CLINT(ARG,X,Y,NPT,IP,YI)
C     LAGRANGE INTERPOLATION,2,3,4,5 POINT
C     IP IS DEGREE OF INTERPOLATION
C     NPT IS NUMBER OF POINTS GIVEN IN TABLE
      DIMENSION X(2),Y(2)
      I=ARG
      I=1
 20   IF(I-X(1))40,25,30
 25   YI=Y(1)
      GO TO 400
 30   I=I+1
      IF(I-NPT)20,20,35
 35   I=NPT
 40   GO TO (40,55,60,60,60),IP
C     2 POINT INTERPOLATION
 55   IF(I-1)56,56,57
 56   I=I+1
 57   CONTINUE
      C01=(I-X(I))/(X(I-1)-X(I))
      C02=(I-X(I-1))/(X(I)-X(I-1))
      YI=C01*Y(I-1)+C02*Y(I)
      GO TO 400
 60   IF(I-I+2)95,95,110
 95   I=I+1
      GO TO 60
 110  IF(I+1-NPT)145,145,140
 140  I=I-1
      GO TO 110
 145  GO TO (145,145,100,200,300),IP
C     3 POINT INTERPOLATION
 100  C1=X(I-1)
      C2=X(I)
      C3=X(I+1)
      C01=((I-C2)*(I-C3))/((C1-C2)*(C1-C3))
      C02=((I-C1)*(I-C3))/((C2-C1)*(C2-C3))
      C03=((I-C1)*(I-C2))/((C3-C1)*(C3-C2))
      YI=C01*Y(I-1)+C02*Y(I)+C03*Y(I+1)
      GO TO 400
C     4 POINT INTERPOLATION
 200  C1=X(I-2)
      C2=X(I-1)
      C3=X(I)
      C4=X(I+1)
      T1=C1-C2
      T2=C1-C3
      T3=C1-C4
      T4=C2-C3
      T5=C2-C4
      T6=C3-C4
      T1=I-C2
      T2=I-C3
      T3=I-C4
      T4=I-C1
      C01=(T1*T2*T3)/(D1*D2*D3)
      C02=-(T4*T2*T3)/(D1*D4*D5)
      C03=(T4*T1*T3)/(D2*D4*D6)
      C04=-(T4*T1*T2)/(D3*D5*D6)
      YI=C01*Y(I-2)+C02*Y(I-1)+C03*Y(I)+C04*Y(I+1)
      GO TO 400
C     5 POINT INTERPOLATION
```

```
300   T1=X(I-3)
      T2=X(I-2)
      T3=X(I-1)
      T4=X(I)
      T5=X(I+1)
      D1=C1-C2
      D2=C1-C3
      D3=C1-C4
      D4=C1-C5
      D5=C2-C3
      D6=C2-C4
      D7=C2-C5
      D8=C3-C4
      D9=C3-C5
      D10=C4-C5
      T1=T-C2
      T2=T-C3
      T3=T-C4
      T4=T-C5
      T5=T-C1
      C01=(T1*T2*T3*T4)/(D1*D2*D3*D4)
      C02=-(T5*T2*T3*T4)/(D1*D5*D6*D7)
      C03=(T5*T1*T3*T4)/(D2*D5*D8*D9)
      C04=-(T5*T1*T2*T4)/(D3*D6*D8*D10)
      C05=(T5*T1*T2*T3)/(D4*D7*D9*D10)
      Y1=C01*Y(I-3)+C02*Y(I-2)+C03*Y(I-1)+
     1C04*Y(I)+C05*Y(I+1)
400   RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR LINT

NO ERRORS

```
                                        3200  FORTRAN  (2.1.0)/(RTS)        / /
      SUBROUTINE SIMEQ(A,N,C)
      DIMENSION A(21,22),B(21,22),C(21)
      TOL=5.0E-08
      KA=N+1
      K2=2
      K3=1
      DO 10 I=1,N
      DO 10 J=1,KA
C     MOVE MATRIX TO WORKING AREA
   10 B(I,J)=A(I,J)
   20 IF(ABS(B(K3,K3))-TOL)30,100,100
   30 IF(K2-N)40,40,300
   40 DO 50 I=1,KA
      C=B(K3,I)
      B(K3,I)=B(K2,I)
      B(K2,I)=C
   50 CONTINUE
      K2=K2+1
      GO TO 20
  100 K1=K3
      K3=K3+1
      K2=K3+1
      IF(K3-N)120,120,200
  120 DO 130 I=K3,KA
      B(K1,I)=B(K1,I)/B(K1,K1)
  130 CONTINUE
      DO 140 I=K3,N
      DO 140 J=K3,KA
      B(I,J)=B(I,J)-B(I,K1)*B(K1,J)
  140 CONTINUE
      GO TO 20
  200 C(N)=B(N,N+1)/B(N,N)
      K3=N-1
  210 K1=K3+1
      D=0.
      DO 220 I=K1,N
  220 D=D+B(K3,I)*C(I)
      C(K3)=B(K3,KA)-D
      K3=K3-1
      IF(K3)350,350,210
  300 TOL=TOL/10.
      IF(TOL-1.0E-30)320,310,310
  310 K2=K3+1
      GO TO 20
  320 WRITE(61,321)
  321 FORMAT(//23H NO SOLUTION FROM SIMEQ//)
  350 RETURN
      END

        3200 FORTRAN DIAGNOSTIC RESULTS - FOR  SIMEQ

NO ERRORS
LOAD,56
RUN,6
```

LISTING OF THREE-STREAM-MIXING COMPUTER PROGRAM

```
                                        3200  FORTRAN  (2,1.0)/$RTS)        / /
      PROGRAM JETMIX
      COMMON XIO,IN,IO,ABX
C     AO=CONSTANT COEFFICIENT OF THE POLYNOMIAL
C     PLYC=COEFFICIENT OF POLYNOMIAL(PSIBAR)
C     AACON(I,J,K)=CONSTANTS USED IN FINDING HOT AND COR,
C     AACON(I,J,K)=LOWER TEMPERATURE RANGE FOR K=2 AND
C     AACON(I,J,K)=UPPER TEMPERATURE RANGE FOR K=1 AND
C     FOR I=1,...,6 AND J=INDEX FOR SPECIE K.
C     AAHOT(I,J) WHERE J=1 OR 2 WITH J=2 IS FOR
C     THE LOWER TEMPERATURE RANGE FOR NITROGEN.
C     TTAB=TABLE OF T
C     KT=NO. OF VALUES IN EACH SPECIE TABLE
C     STAB(I,J) ARE TABLES OF SPECIE VALUES
C     CORRESPONDING TO SPECIE J WHERE I=1,...,KT
C     T IS VALUE OF INPUT FOR TABLE LOOK-UP
C     NALPHP,NALPHPP,NAA,NAA2,NSMAX,NP AND NNP ARE USED
C     TO SETUP INPUT EQUATIONS
C     DT=INCREMENT FOR INTEGRATION
C     DELTAT=INCREMENT FOR T ITERATION
C     D1(I) AND G1(I) FOR EQUATION I+NRO+1 AND CNITRO
C     CBAR(I) IS SOLUTION OF NON-LINEAR SYSTEM OF EQUATIONS
C     SMASS(I) IS CONCENTRATION OR MASS OF SPECIES I
C     CNITRO IS USED IN NRO+1 EQUATION
C     CNITROM=MASS OF NITROGEN
C     KNEW = 1 CAUSES NEW CASE TO BE READ.
C     KNEW IS SET IN NEWTON.
C     NXG = 1 IMPLIES THAT NO XG GUESSES ARE AFTER CASE.
C     NXG = 0 IMPLIES THAT XG GUESSES ARE AFTER CASE.
C     XGMIN IS THE SMALLEST VALUE THAT
C     XG CAN TAKE IN NEWTON.(INPUT VALUE)
C
      DIMENSION TTAB(20),STAB(20,20)
      DIMENSION KSPEC(4,20)
      DIMENSION AOC(40),AO(40),KELEM(20),AACON(6,20,2),AAHOT(6,2)
      DIMENSION NAA(10,20),NAA2(10,20),NP(15),NNP(10)
      DIMENSION NALPHP(10,20),NLPHPP(10,20)
      DIMENSION FKCL(10),FKPL(10),TAB(20),ARRAY(21,22)
      DIMENSION XG(20),CBAR(21)
      DIMENSION SMASS(20),D1(15),G1(15),PP(6),PY(6)
      DIMENSION AOV(40),AOE1(40),AOE2(40),AOE3(40),AOE4(40),AOEE(40)
      DIMENSION FUNION(21),DELX(21),PARRAY(21,22)
      DIMENSION SSTAB(20,20)
      EQUIVALENCE (FDOT,F),(FDOTT,F2)
C
C     READ INPUT
C
C
      IN=60
      IO=61
      ABX=0
 1000 READ (IN,2)KT,NSMAX,NRO,NXG
      I1=1
      I10=10
      NRO1=NRO+1
      NRO2=NRO+2
      NSMAX1=NSMAX+1
      NSMAX2=NSMAX+2
      NRS=NSMAX1-NRO1
      NRS1=NRS+1
      NFA=0
```

```
C
      READ (IN,3)(D1(I),I=1,NRS1)
      READ (IN,3)(G1(I),I=1,NRS1)
      READ (IN,3)(SMASS(I),I=1,NSMAX)
      READ (IN,2)((NALPHP(I,J),I=I1,NRO),J=I1,NSMAX)
      READ (IN,2)((NLPHPP(I,J),I=I1,NRO),J=I1,NSMAX)
      READ (IN,3)PSI,DPSI,DPSBAR,XI,DXI,PSIMAX,XIMAX,TREF,PREF,REFV,
     1PE,HJ,ACCTOL,PSIJ,T,DELTAT,DT,TOL,PSBAR,UJ,UE,USTARJ,RHOJ,RJ,
     2CNITRM
      READ (IN,3)((AACON(I,J,1),I=1,6),J=1,NSMAX)
      READ (IN,3)((AACON(I,J,2),I=1,6),J=1,NSMAX)
      READ (IN,3)((AAHOT(I,J),I=1,6),J=1,2)
      READ(IN,3)AUC
      READ(IN,3)AUV
      READ(IN,3)AUE1
      READ(IN,3)AUE2
      READ(IN,3)AUE3
      READ(IN,3)AUE4
      READ(IN,3)AUEE
      READ (IN,3)(TTAB(K),K=I1,KT)
      READ (IN,3)((STAB(I,J),I=1,KT),J=1,NSMAX)
      READ (IN,2)((KSPEC(I,J),I=1,4),J=1,NSMAX)
 1002 READ (IN,3)(XG(I),I=I1,NSMAX)
      READ (IN,3)XGMIN
      READ(IN,3)SUM
    2 FORMAT (12I5)
    3 FORMAT (6F12.5)
    4 FORMAT(6E20.8)
C
C     SET UP NAA AND NAA2 ARRAYS FROM NALPHP AND NALPHPP
C
      DO 470 I=1,NRO
      DO 470 J=1,NSMAX
      NAA(I,J)=0
      NAA2(I,J)=0
  470 CONTINUE
      DO 450 I=1,NRO
      DO 451 J=1,NSMAX
      IF (NALPHP(I,J))452,453,452
  452 NAA(I,J)=J
  453 IF (NLPHPP(I,J))454,451,454
  454 NAA2(I,J)=J
  451 CONTINUE
  450 CONTINUE
C
C     SET UP NP AND NNP ARRAYS FROM NALPHP AND NALPHPP
C
      K=1
      J=0
  456 J=J+1
      DO 455 I=1,NRO
      IF (NALPHP(I,J))455,455,458
  458 NP(K)=J
      K=K+1
      IF (K-NRS)456,456,459
  455 CONTINUE
      IF (K-NRS)456,456,459
  459 CONTINUE
      K=1
      J=0
  466 J=J+1
      DO 465 I=1,NRO
```

```
      IF (NLPHPP(I,J))465,465,468
  468 NNP(K)=J
      K=K+1
      IF (K-NRO)466,466,469
  465 CONTINUE
      IF (K-NRO)466,466,469
  469 CONTINUE
C
C     PRINT INPUT
C
      WRITE (IO,91)KT,NSMAX,NRO
   91 FORMAT (1H1,///,28H JET MIXING    JOB NO. 560180//,
     131H NUMBER OF ELEMENTS IN TABLES =I3,4X,
     220HNUMBER OF UNKNOWNS =I3,4X,
     327HNUMBER OF INPUT EQUATIONS =I3/)
      WRITE (IO,92)(D1(I),I=I1,NRS1)
   92 FORMAT (6H D1(I)/,(5E22.9))
      WRITE (IO,86)(G1(I),I=I1,NRS1)
   86 FORMAT (6H G1(I)/,(5E22.9))
      WRITE (IO,94)(SMASS(I),I=I1,NSMAX)
   94 FORMAT (18H MASS OF SPECIES I/,(5E22.9))
      WRITE (IO,95)(XG(I),I=I1,NSMAX)
   95 FORMAT (13H CBAR GLESSES/,(5E22.9))
      WRITE (IO,352)XGMIN
  352 FORMAT (/,17H XG MINIMUM VALUE,5X,E17.9//)
      WRITE (IO,96)((NALPHF(I,J),I=I1,NRO),J=I1,NSMAX)
   96 FORMAT (//,7H NALPHP/,(10I10))
      WRITE (IO,97)((NLPHPP(I,J),I=I1,NRO),J=I1,NSMAX)
   97 FORMAT (8H NALPHPP/,(10I10))
      WRITE (IO,98)((NAA(I,J),I=I1,NRO),J=I1,NSMAX)
   98 FORMAT (9H NAA(I,J)/,(10I10))
      WRITE (IO,99)((NAA2(I,J),I=I1,NRO),J=I1,NSMAX)
   99 FORMAT (10H NAA2(I,J)/,(10I10))
      WRITE (IO,80)(NP(I),I=I1,NRS)
   80 FORMAT (6H NP(I)/,(10I10))
      WRITE (IO,81)(NNP(I),I=I1,NRO)
   81 FORMAT (7H NNP(I)/,(10I10))
      WRITE (IO,82)PSI,DPSI,DPSBAR,XI,DXI,PSIMAX,XIMAX,
     1TREF,PREF,REFV,HE,FJ,ACCTOL,PSIJ,T,DELTAT,DT,TOL,
     2FSBAR,UJ,UE,USTARJ,RHOJ,RJ,CNITRM
   82 FORMAT (//,6H PSI  E17.9,4X,5H DPSIE17.9,2X,7HDPSIBARE17.9,
     14X,5HXI   E17.9,4X,5HDXI  E17.9/,6H PSIMXE17.9,4X,
     25HXIMAXE17.9,4X,5HTREF E17.9,4X,5HPREF  E17.9,4X,
     35HREFV E17.9/,6H HE   E17.9,4X,5HHJ   E17.9,3X,
     46HACCTOLE17.9,4X,5HPSIJ E17.9,4X,5HT    E17.9/,
     56H DELATE17.9,4X,5HDT   E17.9,4X,5HTOL  E17.9,
     13X,6HPSIBARE17.9,4X,5HUJ   E17.9/,6H UE   E17.9,
     73X,6HUSTARJE17.9,4X,5HRHOJ E17.9,4X,5HRJ   E17.9,3X,
     86HCNITRME17.9/)
C
      DO 477 K=1,2
      DO 431 J=1,NSMAX
      WRITE (IO,83)J,(AACON(I,J,K),I=1,6)
   83 FORMAT (20X,15HAACON(I,J,K) J=I2/,(5E22.9))
  431 CONTINUE
      IF (K-2)473,472,472
  473 WRITE (IO,476)
  476 FORMAT (//,19X,35H AACON ARRAY FOR LOWER TEMPERATURES/)
  477 CONTINUE
  472 CONTINUE
      WRITE (IO,489)(AAMCT(I,17,I=1,6)
  489 FORMAT (/,20X,5HAAMOT/,(5E22.9))
```

```
      WRITE (IO,475)(AAHCT(I,2),I=1,6)
  475 FORMAT (/,20X,28HAAHCT FOR LOWER TEMPERATURES/,(5E22.9))
      WRITE(IO,84)AUC
      WRITE(IO,84)AUV
      WRITE(IO,84)AUE1
      WRITE(IO,84)AUE2
      WRITE(IO,84)AUE3
      WRITE(IO,84)AUE4
      WRITE(IO,84)AUEE
   84 FORMAT (//,19X,13H AC,A1,...,AN/,(5E22.9))
C
C     PRINT TABLES
C
      WRITE (IO,193)(ITAB(K),K=I1,KT)
  193 FORMAT (///,20X,7HT TABLE//,(5E22.9))
      WRITE (IO,444)
  444 FORMAT (//,20X,32HSPECIE TABLES WITH X=LOG(NUMBER)/)
      DO 200 I=1,NSMAX
      WRITE (IO,194)I
  194 FORMAT (20X,9HSPECIES =I3,6H TABLE)
      WRITE (IO,195)(STAB(K,I),K=I1,KT)
  195 FORMAT (5E22.9)
  200 CONTINUE
C
C     SETTING KELEM(I)=I
C
      DO 5 K=1,20
      KELEM(K)=K
    5 CONTINUE
C
C     CONVERT S1TAB,...,S20TAB TABLES FROM X=LOG(Y) TO Y VALUES
      IF(NFA)100,350,100
  100 IF(NXG)350,350,351
  350 NFA=1
      DO 255 I=1,KT
      DO 255 J=1,NSMAX
  255 SSTAB(I,J)=STAB(I,J)
C
  351 DO 260 J=1,NSMAX
      DO 260 I=1,KT
      IF (STAB(I,J))493,491,491
  491 STAB(I,J)=10.0**STAB(I,J)
      GO TO 490
  493 ST=STAB(I,J)
      NST=ST
      FNST=NST
      FFST=ABS(ST)-ABS(FNST)
      STAB(I,J)=10.0**FFST*10.0**FNST
  490 CONTINUE
  260 CONTINUE
C
      WRITE (IO,445)
  445 FORMAT (//,20X,27HSPECIE TABLES WITH X=NUMBER/)
C
      DO 401 J=1,NSMAX
      WRITE(IO,402)J,(STAB(I,J),I=1,KT)
  402 FORMAT (20X,13HSPECIE TABLE I2/,(5E22.9))
  401 CONTINUE
      WRITE (IO,219)
  219 FORMAT (//,20X,24HKSPEC(I,J) WHERE J=INDEX//)
      DO 88 J=1,NSMAX
      WRITE (IO,89)(KSPEC(I,J),I=1,4)
```

```
   88 CONTINUE
 89   FORMAT (3X,4I5)
      DO 34 I=1,40
      AO(I)=0.
   34 CONTINUE
C
C
C     PREDECK VALUES
C
      KPSI=0
      TOLO=TOL
      PSBARO=PSBAR
      PSIO=PSI
      XIO=XI
      TO=300.
      R=1.987
C
C     900 IS THE RETURN POINT FOR PSI AND
C     XI CHANGE. FOR EACH XI, PSI GOES FROM
C     PSIO TO PSIMAX.  THEN XI=XI+DXI AND THE
C     PROCESS CONTINUES UNTIL XI=XIMAX.
C
C
  900 KC1=0
      TT1=0.
      TT2=0.
      SH1=0.
      SH2=0.
      KLL=0
      DO 31 IL=1,6
      IF(KLL-1)6,7,8
    6 DO 9 I=1,40
      AO(I)=AOV(I)
    9 CONTINUE
      KLL=1
      GO TO 30
    7 DO 10 I=1,40
      AO(I)=AOE1(I)
   10 CONTINUE
      KLL=2
      GO TO 30
    8 IF(KLL-3)12,13,14
   12 DO 17 I=1,40
      AO(I)=AOE2(I)
   17 CONTINUE
      KLL=3
      GO TO 30
   13 DO 18 I=1,40
      AO(I)=AOE3(I)
   18 CONTINUE
      KLL=4
      GO TO 30
   14 IF(KLL-5)19,20,20
   19 DO 21 I=1,40
      AO(I)=AOE4(I)
   21 CONTINUE
      KLL=5
      GO TO 30
   20 DO 22 I=1,40
      AO(I)=AOEE(I)
   22 CONTINUE
   30 NEO=1
```

```
      SUM=0
      I=0
      PSBAR=PSBAR0
      [illegible]=(1.+.5*(PSBAR)/(PSBAR
      Y=0.
      YP=0.0
      AY=0.0
      AYP=0.0
      SUM1=0.0
      SUM2=0.0
      GO TO(24,23,24
   23 CALL MAIN1(PSI,XI,PSIU,PSOUT,PSBAR,AUC,AU,PUC)
      I=1
   24 DO 11 KKK=1,[illegible]
      IN=1
   15 CALL MONOUT(WEG,PSI,PSBAR,PSBAR,F,Y,YP,FK,AY,AYP,SUM1,SUM2,I [illegible])
      CALL MAIN1(PSI,XI,PSIU,PSOUT,PSBAR,AUC,AU,PUC)
      INC=INC+1
      IF(ABS(F).LE.1.0E-36)32,33
   32 Y=AY
      FK=0.
   33 IF(I [illegible]-4)15,15,16
   16 CONTINUE
C     ANY ESTIMED COPING
   11 CONTINUE
      Y=AY
      Z1=PSIU/(2.*XI)
      FP(IL)=Z1*Y
      FY(IL)=Y
      WRITE(10,4)FP(IL),FY(IL)
   31 CONTINUE
      USTAR=1.-FP(1)*(1.-[illegible]E)
      E=USTAR*[illegible]E
      CNITRO=E1(PSI)*(1.-FP(5))+G1(PSI)*RP(5)
      [illegible]=CNITRO/CNITRO
C
C
      WRITE (10,432)PSI,XI,FP(1),USTAR,U,CNITRO,FY(1)
  432 FORMAT (///,6H PSI  E17.9,4X,5HXI   E17.9,3X,
     1 34HAND VALUES AFTER FIRST INTEGRATION///,
     1 6H FP(1)E17.9,4X,6HUSTAR E17.9,4X,
     2 6HU      E17.9,4X,6HCNITRO E17.9,4X,
     6 6H FY    E17.9//)
C
C
C     ;
C     ; FORM OF N-DIM. FOR T EVALUATION.
C     ;
  500 CONTINUE
C
C     SOLVE FOR TABLE VALUE EACH EQUATION
C
      DO 710 I=1,26
      TA(I)=0.0
  710 CONTINUE
      DO 711 I=1,NMO
      T14=0.0
      T15=0.0
      N=NN(I)
      INDEX=AA2(I,N,N)
      DO 711 II=1,N
  711 TA(II)=SSTAR(II,INDEX)
      CALL [illegible](N,STAR,TAE,PT,2,[illegible])
```

```
      IF (YI)252,253,253
253   YI=10.0**YI
      GO TO 254
252   ST=YI
      KST=ST
      FNST=NST
      FFST=ABS(ST)-ABS(FNST)
      YI=10.**FFST*10.0**FNST
254   CONTINUE
      FKPL(I)=YI
      DO 706 J=I1,NSMAX
      Z16=NALPHP(I,J)
      Z13=Z13+Z16
      Z15=NLPHPP(I,J)
706   Z14=Z14+Z15
      Z15=Z13-Z14
      FKCL(I)=FKPL(I)*(82.05*T)**Z15
777   CONTINUE
C
C
      WRITE (IO,421)(FKCL(I),I=1,NRO)
421   FORMAT (20X,7HFKCL(I)/,(5E22.9))
      WRITE (IO,422)(FKPL(I),I=1,NRO)
422   FORMAT (20X,7HFKPL(I)/,(5E22.9))
      WRITE (IO,447)
447   FORMAT (//)
C
C     ZERO ARRAY(I,J)
C
      DO 230 J=1,21
      DO 230 I=1,22
      ARRAY(J,I)=0.0
230   CONTINUE
C
C     SET-UP EQUATIONS 1,...,NRO IN ARRAY(I,J)
C     FIRST FILL IN ARRAY(I,NSMAX2) AND
C     ARRAY(I,NSMAX1)
C
      DO 310 I=1,NRO
      Z1=0.0
      Z2=0.0
      DO 311 J=I1,NSMAX
      Z2=NALPHP(I,J)+Z2
      Z1=NLPHPP(I,J)+Z1
311   CONTINUE
      ARRAY(I,NSMAX2)=Z1-Z2
310   CONTINUE
      DO 312 I=1,NRO
      Z1=0.0
      DO 313 J=1,NSMAX
      SMJ=SMASS(J)
      Z1=Z1+(NLPHPP(I,J)-NALPHP(I,J))*ALOG(SMJ)
313   CONTINUE
      FKCLJ=FKCL(I)
      ARRAY(I,NSMAX1)=-Z1-ALOG(FKCLJ)
312   CONTINUE
C
C     FIND ARRAY(I,J),I=1,...,NRO AND J=1,...,NSMAX
C
      DO 314 I=1,NRO
      DO 314 J=1,NSMAX
      ARRAY(I,J)=NLPHPP(I,J)-NALPHP(I,J)
```

```
  314 CONTINUE
C
C     SET-UP ARRAY(I,J),I=NRO+1,J=1,...,NSMAX
C
      CONP=PREF*TREF*REFV/T
      DO 315 I=1,NSMAX
      ARRAY(NRO1,I)=1./(CONP*SMASS(I))
  315 CONTINUE
      ARRAY(NRO1,NSMAX1)=CNTIRO/CONP
      ARRAY(NRO1,NSMAX2)=-1.
C
C     SET-UP S-R EQUATIONS
C
      DO 347 NKN=1,NRS
      NKNN=NP(NKN)
      N14=NKN+NRO1
  347 ARRAY(N14,NKNN)=1.0
C
      DO 1260 KNCT=1,NRS
      KKNP=NP(KNCT)
      DO 250 I=1,NRO
      DO 251 J=1,NSMAX
      IF (NAA2(I,J))1113,1113,1112
 1113 GO TO 251
 1112 IDEX=NAA2(I,J)
      II=1
 1114 IF (KKNP-KSPEC(II,IDEX))1116,1115,1116
 1116 II=II+1
      IF (II-4)1114,1114,251
 1115 Z2=NALPHP(I,KKNP)
      Z1=NLPHPP(I,IDEX)
      N15=KNCT+NRO1
      ARRAY(N15,IDEX)=(Z2*SMASS(KKNP))/(Z1*SMASS(IDEX))
  251 CONTINUE
  250 CONTINUE
 1260 CONTINUE
C
C     STORE K(NRO+1),...,K(SMAX) INTO ARRAY(I,SMAX+1)
C
      DO 271 I=NRO2,NSMAX1
      J=I-NRO1
      WRITE(10,84)PP(J+1)
      ARRAY(I,NSMAX1)=-D1(J)*(1.-PP(J+1))-G1(J)*PP(J+1)
  271 CONTINUE
C
C
      DO 403 I=1,NSMAX1
      WRITE (10,404)I,(ARRAY(I,J),J=1,NSMAX2)
  404 FORMAT (20X,4HROW I2,14H OF ARRAY(I,J)/,(5E22.9))
  403 CONTINUE
      WRITE (10,447)
      TOL=TOLU
      KNEW = 0
      CALL NEWTON (CBAR,ARRAY,FUNION,PARRAY,DELX,NSMAX,NRO,TOL,T,KNEW,
     1XGMIN,SMASS,CNTIRO,REFV,PREF,TREF,XG)
      IF (KNEW)1001,1001,1004
 1004 IF(NXG)1000,1000,600
  600 DO 93 I=1,KT
      DO 93 J=1,NSMAX
      STAB(I,J)=SSTAB(I,J)
   93 CONTINUE
      GO TO 1002
```

```
 1001 CONTINUE
C
      WRITE (IO,446)(CBAR(I),I=1,NSMAX1)
  446 FORMAT (/,20X,7HCBAR(I)/,(5E22.9))
C
C     CHECK THE ITERATION THROUGH THE ENTHALPY
C
      Z1=0.0
      Z2=0.0
      CBB=CBAR(NSMAX1)
      H=HE*(1.-PP(6))+HJ*PP(6)
      HP=H-1./2.*USTAR**2*LE**2*2.3901E-8
      CBAR(NSMAX1)=CNITRC*CNITRM
C
C     FIRST FIND INTEGRATED PART OF SMALL H
C
      NEO=1
      Z1=0.0
      Z2=0.0
      NSO=0
      DO 150 N=1,NSMAX1
      YP=0.0
      FK=0.0
      AY=0.0
      SUM1=0.0
      SUM2=0.0
      L=0
      Y2=0.
      TT=TO
      IF (L)130,130,131
  130 CALL MAIN2(FDOTT,SMASS,COP,R,AACON,TT,N,AAHOT,NSMAX1,CNITRM)
  131 L=1+L
  650 DO 111 KKK=N,N
      IND=1
  115 CALL RUNKUT(NEO,NSO,DT,TT,F2,Y2,YP,FK,AY,AYP,SUM1,SUM2,IND)
      CALL MAIN2(FDOTT,SMASS,COP,R,AACON,TT,N,AAHOT,NSMAX1,CNITRM)
      IND=IND+1
      IF (IND-4)115,115,116
  116 CONTINUE
  111 CONTINUE
      Y22=Y21
      Y21=Y2
      TT2=TT1
      TT1=TT
      IF (ABS(T-TT)-.0000001)161,161,162
  162 IF (TT-T)650,650,501
C
C     2 POINT INTERPOLATION ON Y2
C
  501 Y2=Y22+((T-TT2)/(TT1-TT2))*(Y21-Y22)
  161 Z1=Y2*CBAR(N)+Z1
      CALL ACAL2(HOT,H,AAHOT,TO,N,AACON,NSMAX1)
      IF (N-NSMAX1)480,479,479
  480 Z2=CBAR(N)*HOT/SMASS(N)+Z2
      GO TO 478
  479 Z2=CBAR(N)*HOT/CNITRM+Z2
  478 CONTINUE
      TT=T
      WRITE (IO,499)COP,HOT,Y2,TT,FDOTT,N
  499 FORMAT (/,5H CO-PE17.9,4X,5H HO-TE17.9,4X,5H Y2  E17.9,
     14X,5H TT  E17.97,5H FDTTE17.9,4X,5H N    I5/)
  150 CONTINUE
```

```
      SH=Z1+Z2
      WRITE (10,433)SH,HP
  433 FORMAT (28H VALUES USED IN THE ENTHALPY,4X,
     15HSH    E17.9,4X,5HHP    E17.9///)
C
C     KC1 IS USED TO DETERMINE CUT-OFF APPROACH FORM
C     ABOVE OR BELOW THE H-PRIME VALUE.
C     SH2,SH1,TC2,TC1 AND HP ARE USED IN INTERPOLATION
C     ON T WHICH GIVES SH=HP.
C
      IF(ABS(HP-SH)-ACCTOL)850,850,25
   25 IF(ABS(SH2-SH)-.01)35,35,2197
   35 IF(SH2)2198,2197,2199
 2198 IF(ABS(SH2)-ABS(HP))821,812,812
 2199 IF(ABS(SH2)-ABS(HP))812,812,821
 2197 SH2=SH1
      SH1=SH
      TC2=TC1
      TC1=T
      IF(SH2)830,822,830
  830 IF(KC1-3)840,850,840
  840 IF (ABS(HP-SH)-ACCTOL)850,850,1197
 1197 IF(HP)400,400,498
  400 IF(SH)841,841,812
  498 IF(SH)821,841,1199
 1199 IF(HP-SH)822,850,811
  841 IF (ABS(HP)-ABS(SH))801,850,802
  801 KC1=1
      IF (ABS(HP-SH)-ACCTOL)850,850,810
  810 IF (ABS(HP)-ABS(SH))811,850,812
  811 T=T+DELTAT
      GO TO 500
  812 T=TC2+(ABS(HP-SH2))/(ABS(SH1-SH2))*ABS(TC1-TC2)
      KC1=3
      GO TO 500
  802 KC1=2
      IF (ABS(HP-SH)-ACCTOL)850,850,820
  820 IF (ABS(HP)-ABS(SH))821,850,822
  821 T=TC1+(ABS(HP-SH1))/(ABS(SH2-SH1))*ABS(TC2-TC1)
      KC1=3
      GO TO 500
  822 T=T-DELTAT
      GO TO 500
  850 CONTINUE
C     CALCULATIONS FOR PRINT-OUT
      Z12=0.0
      RHSTAR=CBB/RHOJ
      IF (KPSI)610,610,611
  610 FSPI1=0.0
      FSPI2=(2.*PSI)/((CBB/RHOJ)*(U/UJ))
      KSPI=1
  611 CONTINUE
      FSPI1=FSPI2
      FSPI2=(2.*PSI)/((CBB/RHOJ)*(U/UJ))
      DFY=FSPI2-FSPI1
      SUM=SUM+DPSI*(FSPI1+1./2.*DFY+(1./3.-1./2.)*1./2.*
     1DFY**2+(1./4.)*DFY**3/FACT(3)+(1./5.-3./2.+11./3.-3.)*
     2DFY**4/FACT(4)+(1./6.-2.+35./4.-50./3.+12.)*DFY**5
     3/FACT(5)+(1./7.-15./8.+17.-225./4.+274./3.-60.)*
     4DFY**6/FACT(6))
      RSTARS=SUM
      RSTAR=SQRT(RSTARS)
```

```
      CALL CAL (XI,XSTARS)
      XSTAR=SQRT(XSTARS)
      RRR=RSTAR*RJ
      X=XSTAR*RJ
C
C
C     FINAL PRINT STATMENT
C
      WRITE (IO,300)T,H,SH,HP,RHSTAR,RSTARS,XSTARS,RRR,X
  300 FORMAT (/,20X,28HFINAL PRINT STATEMENT FOR T=E17.9//,
     17H H      E17.9,4X,6HSH     E17.9,4X,
     26HPRIMEE17.9,4X,6HRHSTARE17.9/,
     37H RSTARSE17.9,4X,6HXSTARSE17.9,4X,
     46HRRR    E17.9,4X,6HX      E17.9//)
C
C
      IF (ABS(PSI-PSIMAX)-.0000001)1195,1195,2196
 2196 IF (PSI-PSIMAX)1196,1195,1195
 1196 PSI=PSI+DPSI
      DO 50 I=I1,NSMAX
      XG(I)=CBAR(I)
   50 CONTINUE
      WRITE(IO,95)(XG(I),I=I1,NSMAX)
      GO TO 900
 1195 IF (ABS(XI-XIMAX)-.0000001)1000,1000,2194
 2194 IF (XI-XIMAX)1194,1000,1000
 1194 PSI=PSIO
      KPSI=0
      SUM=0.0
      XI=XI+DXI
      GO TO 900
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR JETMIX

NO ERRORS

```
      SUBROUTINE NEWTON (CBAR,ARRAY,FUNION,PARRAY,DELX,NSMAX,NRO,TOL,
     1T,KNEW,XGMIN,SMASS,CNITRO,REFV,PREF,TREF,XG)
C     CBAR(21) ARE APPROXIMATE ANSWER WHICH
C     ARE TO BE USED FOR THE NEWTON METHOD
C     ARRAY(21,22) USED IN EVALUATING THE GIVEN FUNCTIONS
C     FUNCTION(21) IS THE VALUE OF FUNCTIONS
C     PARRAY(21,22) IS AN ARRAY OF PARTIALS USED
C     IN NEWTONS METHOD
C     DELX(21) IS DELTA XS USED TO SET-UP
C     THE NEW CBAR VALUES
C     CNITRO IS USED IN NRO+1 EQUATION
C     SMASS(I) IS CONCENTRATION OR MASS OF SPECIES I
C     XG(I) IS THE INITIAL GUESS FOR CBAR(I)
C     XGMIN IS THE SMALLEST VALUE THAT
C     XG CAN TAKE IN NEWTON.(INPUT VALUE)
      COMMON XIO,IN,IO,ABX
      DIMENSION CBAR(21),ARRAY(21,22),FUNION(21),
     1PARRAY(21,22),DELX(21),XX(21),XX1(21)
      DIMENSION EXPXX(21),SMASS(20),XG(20),DELXI(21)
      SOL1=0.0
      NSMAX1=NSMAX+1
      NSMAX2=NSMAX+2
      NRO1=NRO+1
      NRO2=NRO+2
      TOLNEW=TOL*.001
      Z1=CNITRO
      Z2=PREF*TREF*REFV/T
      DO 222 I=1,NSMAX
      Z1=XG(I)/SMASS(I)+Z1
  222 CONTINUE
      RHOVUE=Z2/Z1
      CBAR(NSMAX1)=RHOVUE
      DO 115 I=1,NSMAX
      CBAR(I)=XG(I)
  115 CONTINUE
C     SET XX(I)=CBAR(I)
      DO 1 I=1,NSMAX1
      XX(I)=CBAR(I)
    1 CONTINUE
      DO 116 I=1,NSMAX1
      IF (XX(I)-XGMIN)117,117,119
  119 XX(I)=ALOG(XX(I))
      GO TO 116
  117 XX(I)=XGMIN
      XX(I)=ALOG(XX(I))
  116 CONTINUE
 1000 WRITE(IO,100)(XX(I),I=1,NSMAX1)
  100 FORMAT(10H XX VALUES/,(5E22.9))
      DO 41 I=1,NSMAX1
      EXPXX(I)=EXP(XX(I))
      IF (EXPXX(I)-10.)41,41,902
   41 CONTINUE
C     FIND THE VALUE OF FUNION(I),I=1,....NRO
      DO 2 I=1,NRO
      Z1=0.0
      DO 3 J=1,NSMAX
      Z1=Z1+ARRAY(I,J)*XX(J)
    3 CONTINUE
      FUNION(I)=Z1+ARRAY(I,NSMAX1)+ARRAY(I,NSMAX2)*XX(NSMAX1)
    2 CONTINUE
```

```
C     FIND THE VALUE OF FUNION(I),I=NRO+1
      Z1=0.0
      DO 4 J=1,NSMAX
      Z1=Z1+ARRAY(NRO1,J)*EXPXX(J)
    4 CONTINUE
      FUNION(NRO1)=Z1+ARRAY(NRO1,NSMAX1)-EXP(-XX(NSMAX1))
C     FIND THE VALUE OF FUNION(I),I=NRO+2,...,NSMAX+1
      DO 5 I=NRO2,NSMAX1
      Z1=0.0
      DO 6 J=1,NSMAX
      Z1=Z1+ARRAY(I,J)*EXPXX(J)
    6 CONTINUE
      FUNION(I)=Z1+ARRAY(I,NSMAX1)
    5 CONTINUE
C     STORE FUNCTION VALUES IN PARRAY
      DO 35 I=1,NSMAX1
   35 PARRAY(I,NSMAX2)=-FUNION(I)
      IF(ABX)26,25,26
   25 WRITE (10,27)(FUNICN(I),I=1,NSMAX1)
   27 FORMAT (11H FUNCTICN I/,(5E22.9))
   26 CONTINUE
C     PARTIALS OF FUNCTICNS
      DO 7 I=1,NRO
      DO 8 J=1,NSMAX
      PARRAY(I,J)=ARRAY(I,J)
    8 CONTINUE
      PARRAY(I,NSMAX1)=ARRAY(I,NSMAX2)
    7 CONTINUE
      DO 9 J=1,NSMAX
      PARRAY(NRO1,J)=ARRAY(NRO1,J)*EXPXX(J)
    9 CONTINUE
      PARRAY(NRO1,NSMAX1)=EXP(-XX(NSMAX1))
      DO 10 I=NRO2,NSMAX1
      DO 11 J=1,NSMAX
      PARRAY(I,J)=ARRAY(I,J)*EXPXX(J)
   11 CONTINUE
      PARRAY(I,NSMAX1)=0.0
   10 CONTINUE
      IF (SOL1)52,52,54
   52 DO 50 I=1,NSMAX1
      Z1=0.
      DO 51 J=1,NSMAX1
      Z1=Z1+ABS(PARRAY(I,J))
   51 CONTINUE
      IF (Z1-1.0)50,50,54
   50 CONTINUE
      WRITE (10,55)
   55 FORMAT (20X,28HM*MATRIX IMPLIES CONVERGENCE//)
      SOL1=1.0
   54 CONTINUE
      IF(ABX)31,30,31
   30 WRITE (10,32)((PARRAY(I,J),I=1,NSMAX1),J=1,NSMAX2)
   32 FORMAT (14H PARTIAL ARRAY/,(5E22.9))
   31 CONTINUE
      CALL SIMEQ(PARRAY,NSMAX1,DELX)
      DO 12 I=1,NSMAX1
      XX1(I)=XX(I)+DELX(I)
   12 CONTINUE
      IF(ABX)14,13,14
   13 WRITE (10,15)(EXPXX(I),I=1,NSMAX1)
      WRITE (10,16)(DELX(I),I=1,NSMAX1)
      WRITE(10,21)(XX1(I),I=1,NSMAX1)
```

```
   15 FORMAT (10H XX VALUES/,(5E22.9))
   16 FORMAT (11H DELTA X(I)/,(5E22.9))
   21 FORMAT(11H XX1 VALUES/,(5E22.9))
   14 CONTINUE
      DO 18 I=1,NSMAX1
      DELXI(I)=EXP(DELX(I))
      IF(ABS(DELXI(I)-1.)-TOLNEW)18,18,20
   18 CONTINUE
      DO 22 I=1,NSMAX1
      CBAR(I)=EXP(XX1(I))
   22 CONTINUE
      DO 301 I=1,NSMAX
      XG(I)=CBAR(I)
  301 CONTINUE
      RETURN
   20 DO 60 I=1,NSMAX1
      XX(I)=XX1(I)
   60 CONTINUE
      GO TO 1000
  902 WRITE(IO,903)
  903 FORMAT (14H CASE DIVERGES//)
      KNEW = 1
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR NEWTON

NO ERRORS

```
      SUBROUTINE FIO (PXI,PPSIJ,PPSI,ANS,SPSBAR)
      COMMON XIO
C     X IS THE POINT OF EVALUATION
C     ANS IS THE ANSWER DESIRED
C     PPSIJ,PPSI AND PXI ARE NEEDED VALUES
C     FOR THE CALCULATIONS
      XIO=(PPSI*SPSBAR/PPSIJ)/(2.0*PXI/PPSIJ)
      IF(XIO-20.)5,5,2
    5 N=1
      ANS=XIO**2/4.0+1.0
      ANS1=ANS
   10 N=N+1
      ANS=ANS+(XIO**N/(2.0**N*FACT(N)))**2
      ANS2=ANS1
      ANS1=ANS
      IF(N-200)4,4,2
    4 IF (ABS(ANS1-ANS2)-.00000001)2,2,3
    3 GO TO 10
    2 RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR FIO

NO ERRORS

```
      SUBROUTINE MAIN1(SPSI,SXI,SPSIJ,SFDOT,SPSBAR,SAOC,SAO,SPOL)
      DIMENSION SAOC(40),SAO(40)
      COMMON XID
C     SFDOT IS THE DOT EVALUATION TO BE INTEGRATED
      CALL FIO(SXI,SPSIJ,SFSJ,ANSWER,SPSBAR)
      IF(XID-20.)2,2,4
    2 Z1=(SPSBAR**2/(4.0*SXI))/SPSIJ
      IF(Z1-700.)5,5,3
    3 Z2=0.
      GO TO 7
    5 Z2=EXP(-Z1)
    7 Z3=(SPSI**2/(SPSIJ**2))/((4.*SXI)/SPSIJ)
      IF(Z3-700.)9,9,8
    8 Z4=0.
      GO TO 10
    9 Z4=EXP(-Z3)
   10 CALL LINT(SPSBAR,SAOC,SAO,40,2,SPOL)
      SFDOT=Z2*Z4*SPOL*SPSBAR*ANSWER
      GO TO 6
    4 Z1=((-(SPSI**2)-(SPSBAR**2*SPSIJ**2)+2.*SPSI*SPSIJ
     1*SPSBAR)/(4.*SXI*SPSIJ))
      IF(Z1)11,12,12
   11 IF(Z1+700.)13,13,12
   12 Z2=EXP(Z1)
      GO TO 14
   13 Z2=0.
   14 CALL LINT(SPSBAR,SAOC,SAO,40,2,SPOL)
      EM=SQRT((2.*3.14159265*SPSI*SPSBAR)/(2.*SXI))
      SFDOT=(Z2*SPOL*SPSBAR)/EM
    6 RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR MAIN1

NO ERRORS

```
      SUBROUTINE POLVAL(AAO,A,YY,PPSBAR)
C     AAO IS THE CONSTANT VALUE OF THE POLYNOMIAL
C     AA IS THE A(1)...A(20) COEFFICIENTS
C     YY IS THE VALUE OF THE POLYNOMIAL AT PPSIBAR
      DIMENSION A(20)
      YY=AAO+((((((((((((((((((((A(20)*PPSBAR+A(19))*PPSBAR+
     1A(18))*PPSBAR+A(17))*PPSBAR+A(16))*PPSBAR+A(15))*
     2PPSBAR+A(14))*PPSBAR+A(13))*PPSBAR+A(12))*
     3PPSBAR+A(11))*PPSBAR+A(10))*PPSBAR+A(9))*
     4PPSBAR+A(8))*PPSBAR+A(7))*PPSBAR+A(6))*
     5PPSBAR+A(5))*PPSBAR+A(4))*PPSBAR+A(3))*
     6PPSBAR+A(2))*PPSBAR+A(1))*PPSBAR)
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR POLVAL

NO ERRORS

```
      SUBROUTINE ACAL1(CCOP,RR,AAACON,TT,N,AAAHOT,NSMAX1)
      DIMENSION AAACON(6,20,2),AAAHOT(6,2)
      IF (TT-1000.)1,2,2
    2 IF (N-NSMAX1)11,22,22
   11 CCOP=(AAACON(1,N,1)+AAACON(2,N,1)*TT+AAACON(3,N,1)*TT**2+
     1AAACON(4,N,1)*TT**3+AAACON(5,N,1)*TT**4)*RR
      GO TO 3
   22 CCOP=(AAAHOT(1,1)+AAAHOT(2,1)*TT+AAAHOT(3,1)*TT**2+
     1AAAHOT(4,1)*TT**3+AAAHOT(5,1)*TT**4)*RR
    3 CONTINUE
      RETURN
    1 IF (N-NSMAX1)31,32,32
   31 CCOP=(AAACON(1,N,2)+AAACON(2,N,2)*TT+AAACON(3,N,2)*TT**2+
     1AAACON(4,N,2)*TT**3+AAACON(5,N,2)*TT**4)*RR
      GO TO 33
   32 CCOP=(AAAHOT(1,2)+AAAHOT(2,2)*TT+AAAHOT(3,2)*TT**2+
     1AAAHOT(4,2)*TT**3+AAAHOT(5,2)*TT**4)*RR
   33 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR ACAL1

NO ERRORS

```
      SUBROUTINE MAIN2(FDOTT,SMASS,COP,R,AACON,TT,N,AAHOT,NSMAX1,
     1CNITRM)
      DIMENSION AACON(6,20,2),SMASS(20),AAHOT(6,2)
      CALL ACAL1(COP,R,AACON,TT,N,AAHOT,NSMAX1)
      IF (N-NSMAX1)1,2,2
    1 FDOTT=COP/SMASS(N)
      GO TO 3
    2 FDOTT=COP/CNITRM
    3 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR MAIN2

NO ERRORS

```
      SUBROUTINE ACAL2(HHOT,RR,AAAHOT,TT,N,AAACON,NSMAX1)
      DIMENSION AAAHOT(6,2),AAACON(6,20,2)
      IF (TT-1000.)1,2,2
    1 K=2
      GO TO 4
    2 K=1
    4 IF (N-NSMAX1)5,6,6
    5 HHOT=(AAACON(1,N,K)+AAACON(2,N,K)*TT/2.+AAACON(3,N,K)*TT*TT/3.+
     1AAACON(4,N,K)*TT**3/4.+AAACON(5,N,K)*TT**4/5.+
     2AAACON(6,N,K)/TT)*RR*TT
      GO TO 7
    6 HHOT=(AAAHOT(1,K)+AAAHOT(2,K)*TT/2.+AAAHOT(3,K)*TT*TT/3.+
     1AAAHOT(4,K)*TT**3/4.+AAAHOT(5,K)*TT**4/5.+
     2AAAHOT(6,K)/TT)*RR*TT
    7 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR ACAL2

NO ERRORS

```
      SUBROUTINE CAL(XI,ANSWER)
      ANSWER =2.*XI/.00075
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR CAL

NO ERRORS

```
      FUNCTION FACT(I)
C     DETERMINES N-FACTORIAL FOR NN
      IF(I-1)1,1,2
    1 FACT=1.
      RETURN
    2 FACT=1.
      DO 3 J=2,I
      FJ=J
    3 FACT=FACT*FJ
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR FACT

NO ERRORS

```
      SUBROUTINE RUNKUT(NEQ,NSO,H,X,F,Y,YP,FK,AY,AYP,SUM1,SUM2,INDT)
C     RUNKUT**RUNGE KUTTA INTEGRATION
      DIMENSION F(100),Y(100),YP(100),AY(100),AYP(100),SUM1(100)
      DIMENSION SUM2(100),FK(100)
C     THIS INTEGRATES NEQ EQUATIONS,NSO OF WHICH ARE SECOND ORDER.
C     H IS THE STEP SIZE,X THE INDEPENDENT VARIABLE. THE DEPENDENT
C     VARIABLES ARE STORED IN Y AND YP,SECOND ORDER EQUATIONS FIRST.
C     F IS THE VALUE OF THE FUNCTION (FIRST OR SECOND DERIVATIVE.
C     AUXILIARY STORAGE MUST BE FURNISHED, FK(NEQ),AY(NEQ),AYP(NEQ),
C     SUM1(NSO),SUM2(NEQ). SUBROUTINE IS CALLED FOUR TIMES WITH VALUES
C     OF X,Y,YP,F. SUBROUTINE FURNISHES VALUES OF X,Y,YP. INDT IS
C     COUNTER (1,2,3 OR 4). SUBROUTINE STORES NO ESSENTIAL INFORMATION
C     INTERNALLY
      IF(INDT-2) 1,2,5
    1 X=X+H/2.
      AL1=H/2.
      AL2=0.
      AL3=.5
      AL4=1.
      DO 6 I=1,NEQ
      AY(I)=Y(I)
      AYP(I)=YP(I)
      SUM1(I)=0.
    6 SUM2(I)=0.
      GO TO 7
    2 AL1=H/2.
      AL2=H/4.
      AL3=.5
      AL4=2.
      GO TO 7
    5 IF (INDT-3) 3,3,4
    3 X=X+H/2.
      AL1=H
      AL2=H/2.
      AL3=1.
      AL4=2.
    7 DO 8 I=1,NEQ
      IF (I  -NSO) 9,9,10
    9 Y(I)=AY(I)+AL1*AYP(I)+AL2*FK(I)
      FK(I)=H*F(I)
      YP(I)=AYP(I)+AL3*FK(I)
      SUM1(I)=SUM1(I)+FK(I)
      GO TO 11
   10 FK(I)=H*F(I)
      Y(I)=AY(I)+AL3*FK(I)
   11 SUM2(I)=SUM2(I)+AL4*FK(I)
    8 CONTINUE
      RETURN
    4 DO 12 I=1,NEQ
      FK(I)=H*F(I)
      SUM2(I)=(SUM2(I)+FK(I))/6.
      IF (I-NSO) 13,13,14
   13 Y(I)=AY(I)+H*AYP(I)+F* SUM1(I)/6.
      YP(I)=AYP(I)+SUM2(I)
      GO TO 12
   14 Y(I)=AY(I)+SUM2(I)
   12 CONTINUE
      RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR RUNKUT

```
      SUBROUTINE LINT(ARG,X,Y,NPT,IP,YI)
C     LAGRANGE INTERPOLATION,2,3,4,5 POINT
C     IP IS DEGREE OF INTERPOLATION
C     NPT IS NUMBER OF POINTS GIVEN IN TABLE
      DIMENSION X(2),Y(2)
      T=ARG
      I=1
 20   IF(T-X(I))40,25,30
 25   YI=Y(I)
      GO TO 400
 30   I=I+1
      IF(I-NPT)20,20,35
 35   I=NPT
 40   GO TO (40,55,60,60,60),IP
C     2 POINT INTERPOLATION
 55   IF(I-1)56,56,57
 56   I=I+1
 57   CONTINUE
      CO1=(T-X(I))/(X(I-1)-X(I))
      CO2=(T-X(I-1))/(X(I)-X(I-1))
      YI=CO1*Y(I-1)+CO2*Y(I)
      GO TO 400
 60   IF(I-IP+2)95,95,110
 95   I=I+1
      GO TO 60
 110  IF(I+1-NPT)145,145,140
 140  I=I-1
      GO TO 110
 145  GO TO (145,145,100,200,300),IP
C     3 POINT INTERPOLATION
 100  C1=X(I-1)
      C2=X(I)
      C3=X(I+1)
      CO1=((T-C2)*(T-C3))/((C1-C2)*(C1-C3))
      CO2=((T-C1)*(T-C3))/((C2-C1)*(C2-C3))
      CO3=((T-C1)*(T-C2))/((C3-C1)*(C3-C2))
      YI=CO1*Y(I-1)+CO2*Y(I)+CO3*Y(I+1)
      GO TO 400
C     4 POINT INTERPOLATION
 200  C1=X(I-2)
      C2=X(I-1)
      C3=X(I)
      C4=X(I+1)
      D1=C1-C2
      D2=C1-C3
      D3=C1-C4
      D4=C2-C3
      D5=C2-C4
      D6=C3-C4
      T1=T-C2
      T2=T-C3
      T3=T-C4
      T4=T-C1
      CO1=(T1*T2*T3)/(D1*D2*D3)
      CO2=-(T4*T2*T3)/(D1*D4*D5)
      CO3=(T4*T1*T3)/(D2*D4*D6)
      CO4=-(T4*T1*T2)/(D3*D5*D6)
      YI=CO1*Y(I-2)+CO2*Y(I-1)+CO3*Y(I)+CO4*Y(I+1)
      GO TO 400
C     5 POINT INTERPOLATION
```

```
300   C1=X(I-3)
      C2=X(I-2)
      C3=X(I-1)
      C4=X(I)
      C5=X(I+1)
      D1=C1-C2
      D2=C1-C3
      D3=C1-C4
      D4=C1-C5
      D5=C2-C3
      D6=C2-C4
      D7=C2-C5
      D8=C3-C4
      D9=C3-C5
      D10=C4-C5
      T1=T-C2
      T2=T-C3
      T3=T-C4
      T4=T-C5
      T5=T-C1
      CO1=(T1*T2*T3*T4)/(D1*D2*D3*D4)
      CO2=-(T5*T2*T3*T4)/(D1*D5*D6*D7)
      CO3=(T5*T1*T3*T4)/(D2*D5*D8*D9)
      CO4=-(T5*T1*T2*T4)/(D3*D6*D8*D10)
      CO5=(T5*T1*T2*T3)/(D4*D7*D9*D10)
      YI=CO1*Y(I-3)+CO2*Y(I-2)+CO3*Y(I-1)+
     1CO4*Y(I)+CO5*Y(I+1)
400   RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR LINT

NO ERRORS

```
      SUBROUTINE SIMEQ(A,N,C)
      DIMENSION A(21,22),B(21,22),C(21)
      TOL=5.0E-08
      KA=N+1
      K2=2
      K3=1
      DO 10 I=1,N
      DO 10 J=1,KA
C     MOVE MATRIX TO WORKING AREA
   10 B(I,J)=A(I,J)
   20 IF(ABS(B(K3,K3))-TOL)30,100,100
   30 IF(K2-N)40,40,300
   40 DO 50 I=1,KA
      D=B(K3,I)
      B(K3,I)=B(K2,I)
      B(K2,I)=D
   50 CONTINUE
      K2=K2+1
      GO TO 20
  100 K1=K3
      K3=K3+1
      K2=K3+1
      IF(K3-N)120,120,200
  120 DO 130 I=K3,KA
      B(K1,I)=B(K1,I)/B(K1,K1)
  130 CONTINUE
      DO 140 I=K3,N
      DO 140 J=K3,KA
      B(I,J)=B(I,J)-B(I,K1)*B(K1,J)
  140 CONTINUE
      GO TO 20
  200 C(N)=B(N,N+1)/B(N,N)
      K3=N-1
  210 K1=K3+1
      D=0.
      DO 220 I=K1,N
  220 D=D+B(K3,I)*C(I)
      C(K3)=B(K3,KA)-D
      K3=K3-1
      IF(K3)350,350,210
  300 TOL=TOL/10.
      IF(TOL-1.0E-30)320,310,310
  310 K2=K3+1
      GO TO 20
  320 WRITE(61,321)
  321 FORMAT(//23H NO SOLUTION FROM SIMEQ//)
  350 RETURN
      END
```

3200 FORTRAN DIAGNOSTIC RESULTS - FOR SIMEQ

NO ERRORS

REFERENCES

1. Audeh, B. J., "Turbulent Mixing of Three Concentric Reacting Jets," Heat Technology Laboratory, Inc. Technical Report, HTL-TR-29, November 19, 1965.

2. Stull, D. R., "JANAF Thermochemical Tables", Dow Chemical Co., Midland, Michigan, August 1965.

3. Zeleznik, F. J. and Gordon, S., "A General IBM 704 or 7090 Computer Program for Computation of Chemical Equilibrium Compositions, Rocket Performance, and Chapman-Jouguet Detonations", NASA TN D-1454, October 1962.

4. Zeleznik, F. J. and Gordon, S., "An Analytical Investigation of Three General Methods of Calculating Chemical-Equilibrium Compositions", NASA TN D-473, September 1960.

5. Lapidus, L., Digital Computation for Chemical Engineers, McGraw-Hill Book Company, Inc., New York, New York, 1962.

6. Taylor, R., "CCDS Chemical Equilibrium Subroutine", Chrysler Corp., New Orleans, Louisanna, (To be published).

TABLE I. - INITIAL CONDITIONS FOR THREE-STREAM MIXING

Physical Property	LOCATION		
	Jet Exit	Turbine Exhaust Exit	Free Stream
Temperature, °K	2751.5	830.0	281.1
Pressure, dyne/cm^2	8.9149×10^5	8.9149×10^5	8.9149×10^5
Density, gm/cm^3	9.10712×10^{-5}	1.17099×10^{-4}	1.1005×10^{-3}
Speed, cm/sec	293,126.1	36,576.0	9,144.0
Enthalpy, cal/gm	73.38	-1,049.25	125.34
Mass Fractions			
C_{H_2}	0.04377	0.16558	0.0
C_{O_2}	0.0	0.0	0.233
C_{H_2O}	0.02246	0.0	0.0
C_C	0.0	0.10093	0.0
C_{CO}	0.00428	0.0	0.0
C_{CO_2}	0.93267	0.67101	0.0
C_{CH_4}	0.0	0.06249	0.0
C_{N_2}	0.0	0.0	0.767

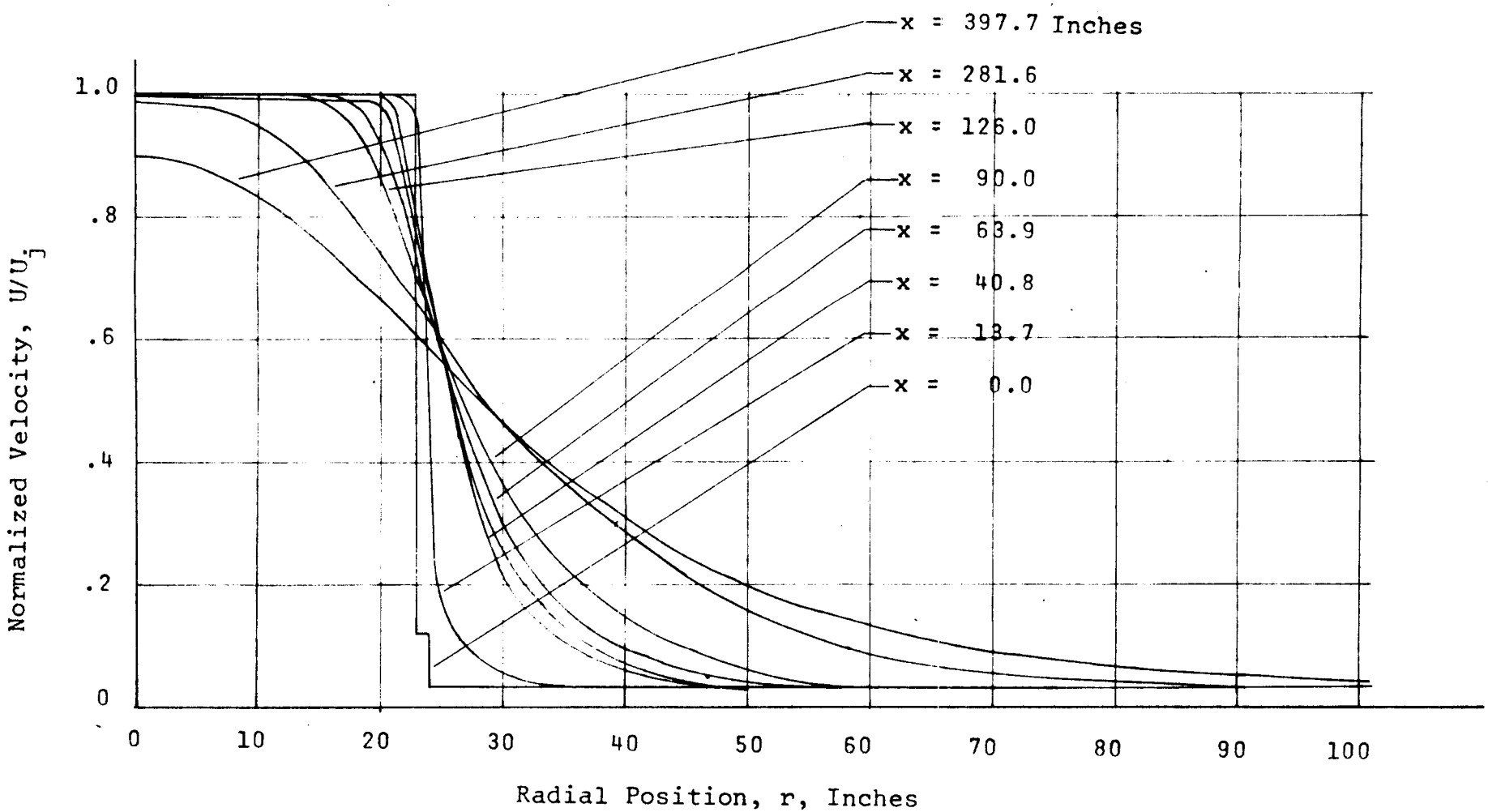

FIG. 1 VELOCITY PROFILE FOR THREE-STREAM-MIXING

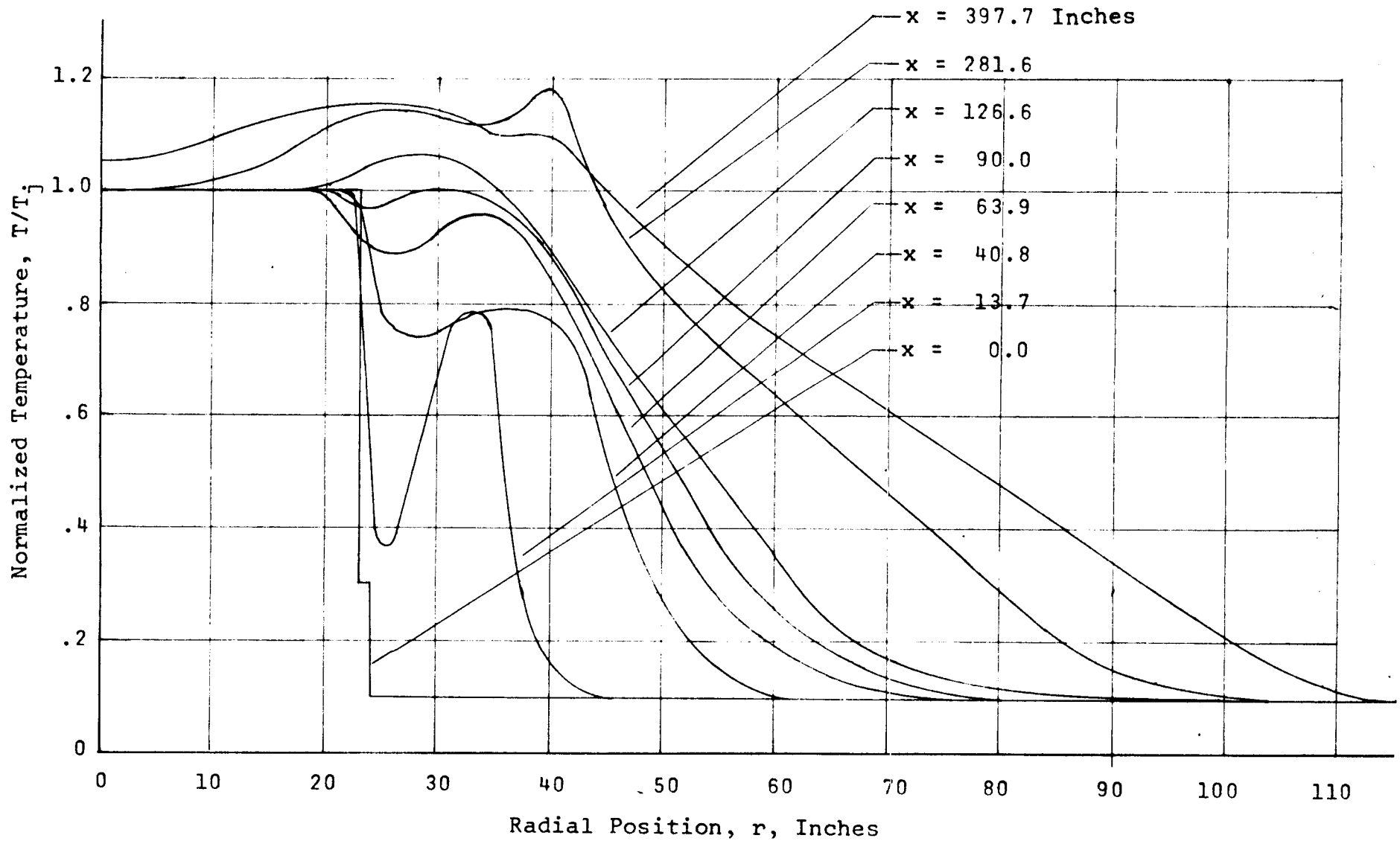

FIG. 2 TEMPERATURE PROFILE FOR THREE-STREAM-MIXING

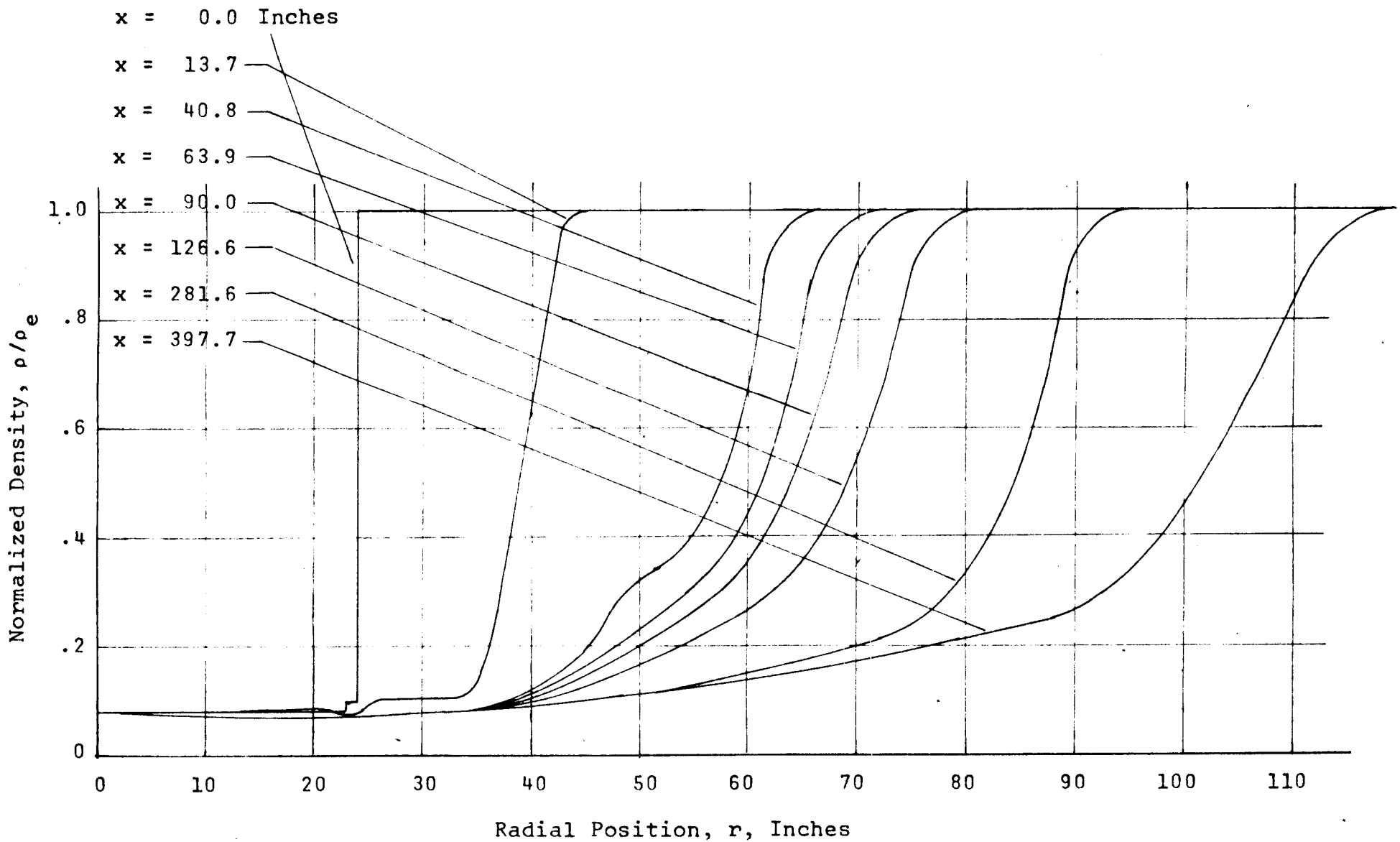

FIG. 3 DENSITY PROFILE FOR THREE-STREAM-MIXING

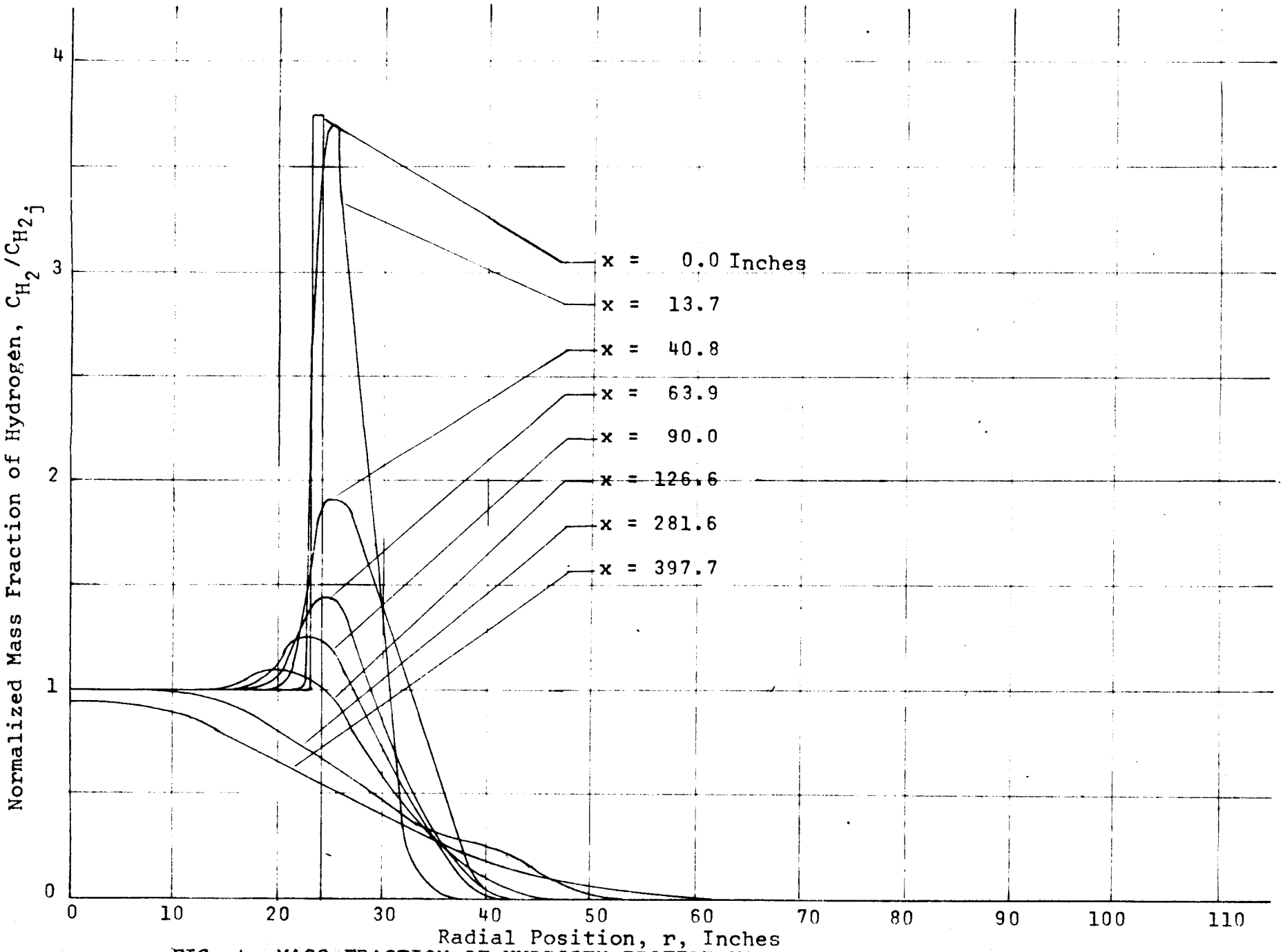

FIG. 4 MASS FRACTION OF HYDROGEN PROFILE FOR THREE-STREAM-MIXING

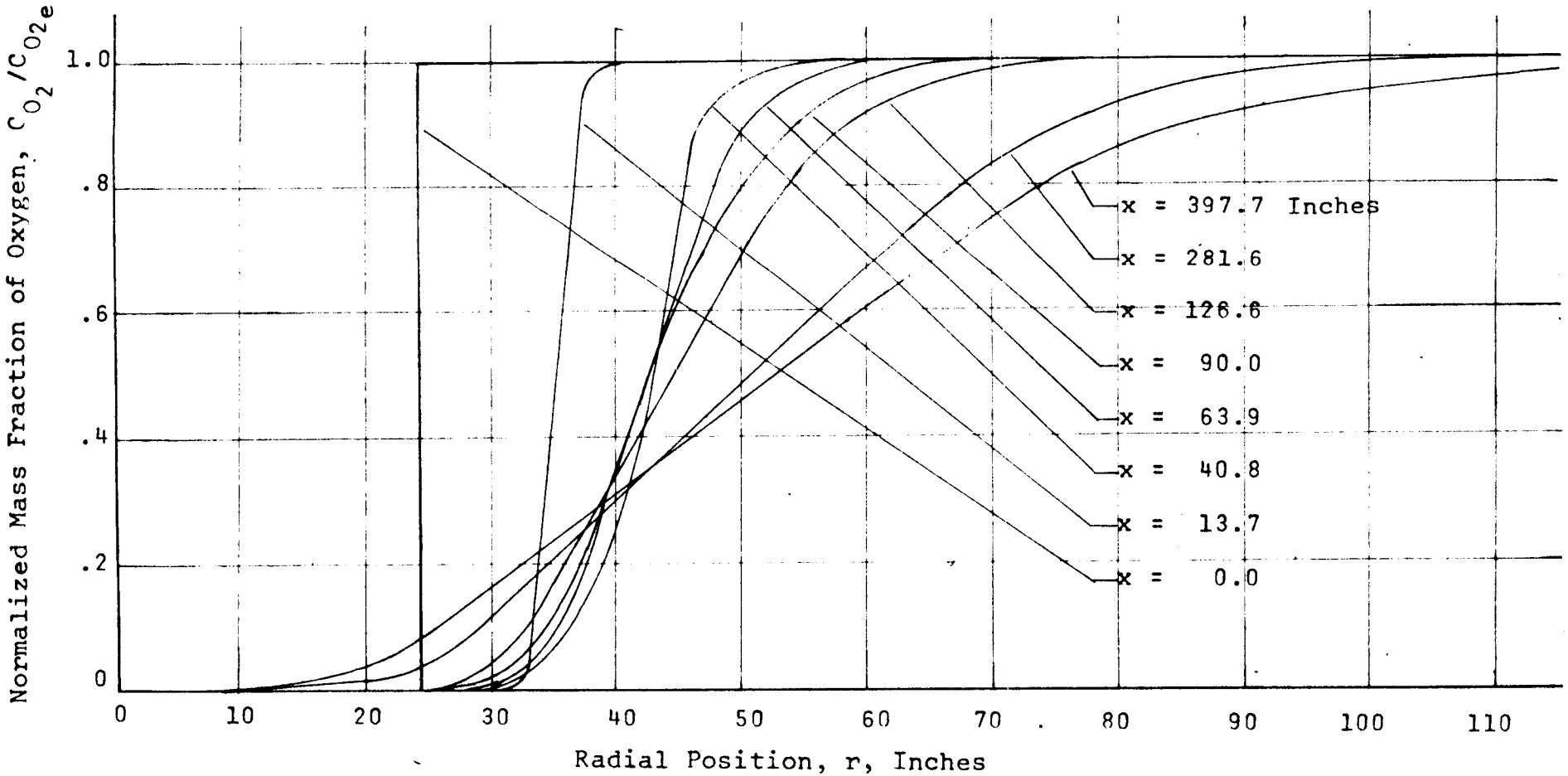

FIG. 5 MASS FRACTION OF OXYGEN PROFILE FOR THREE-STREAM-MIXING

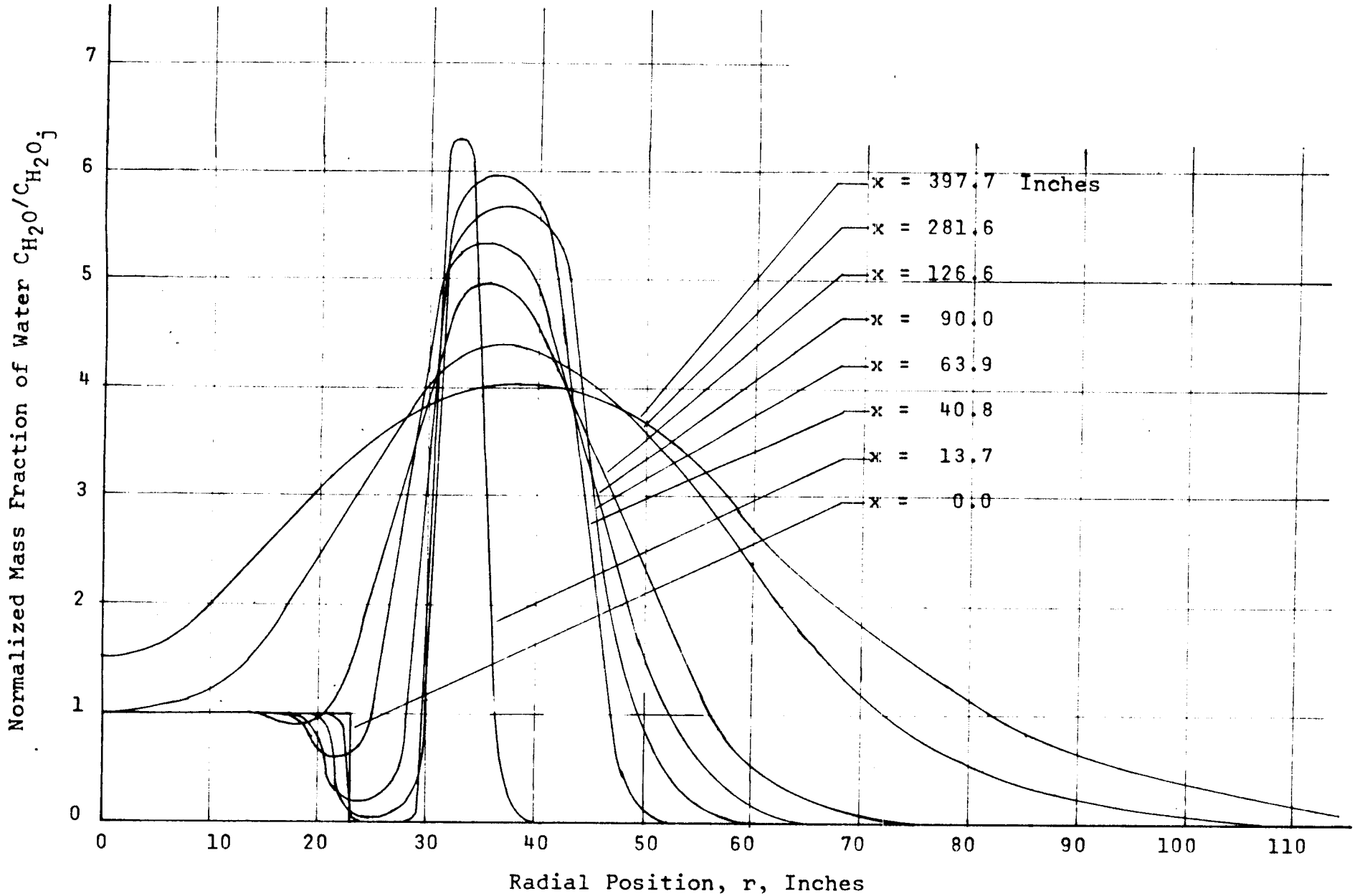

FIG. 6 MASS FRACTION OF WATER PROFILE FOR THREE-STREAM-MIXING

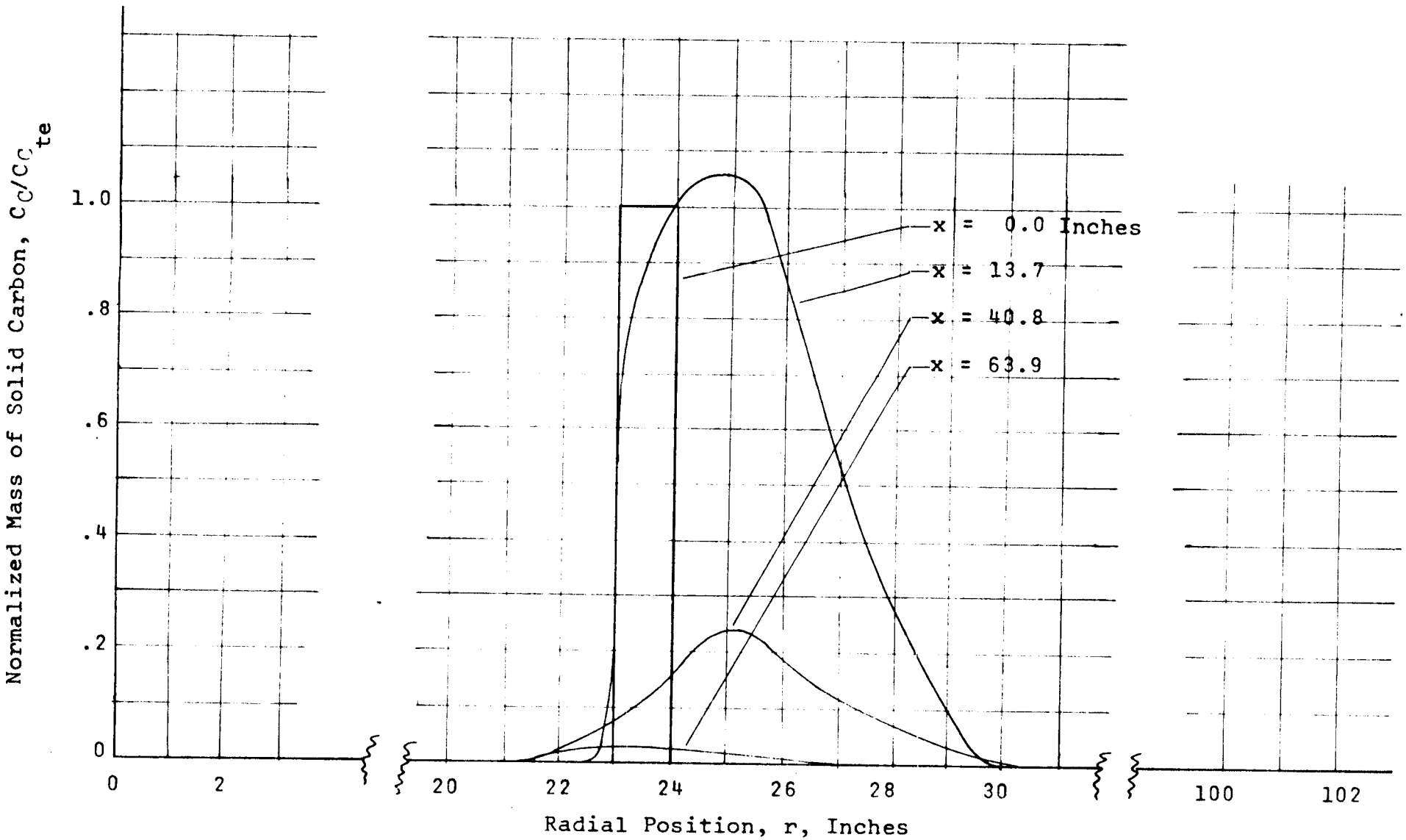

FIG. 7 MASS FRACTION OF SOLID CARBON PROFILE FOR THREE-STREAM-MIXING

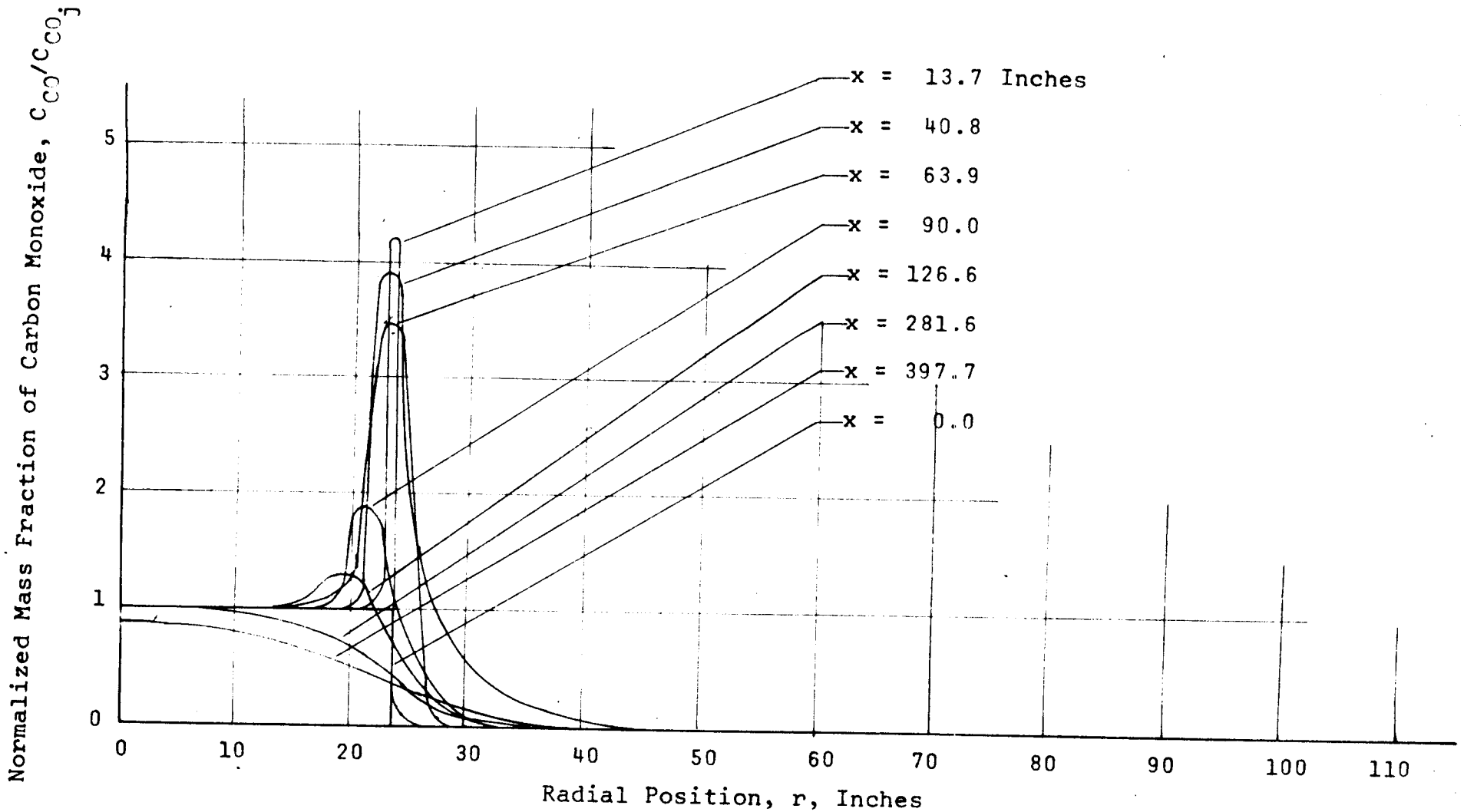

FIG. 8 MASS FRACTION OF CARBON MONOXIDE PROFILE FOR THREE-STREAM-MIXING

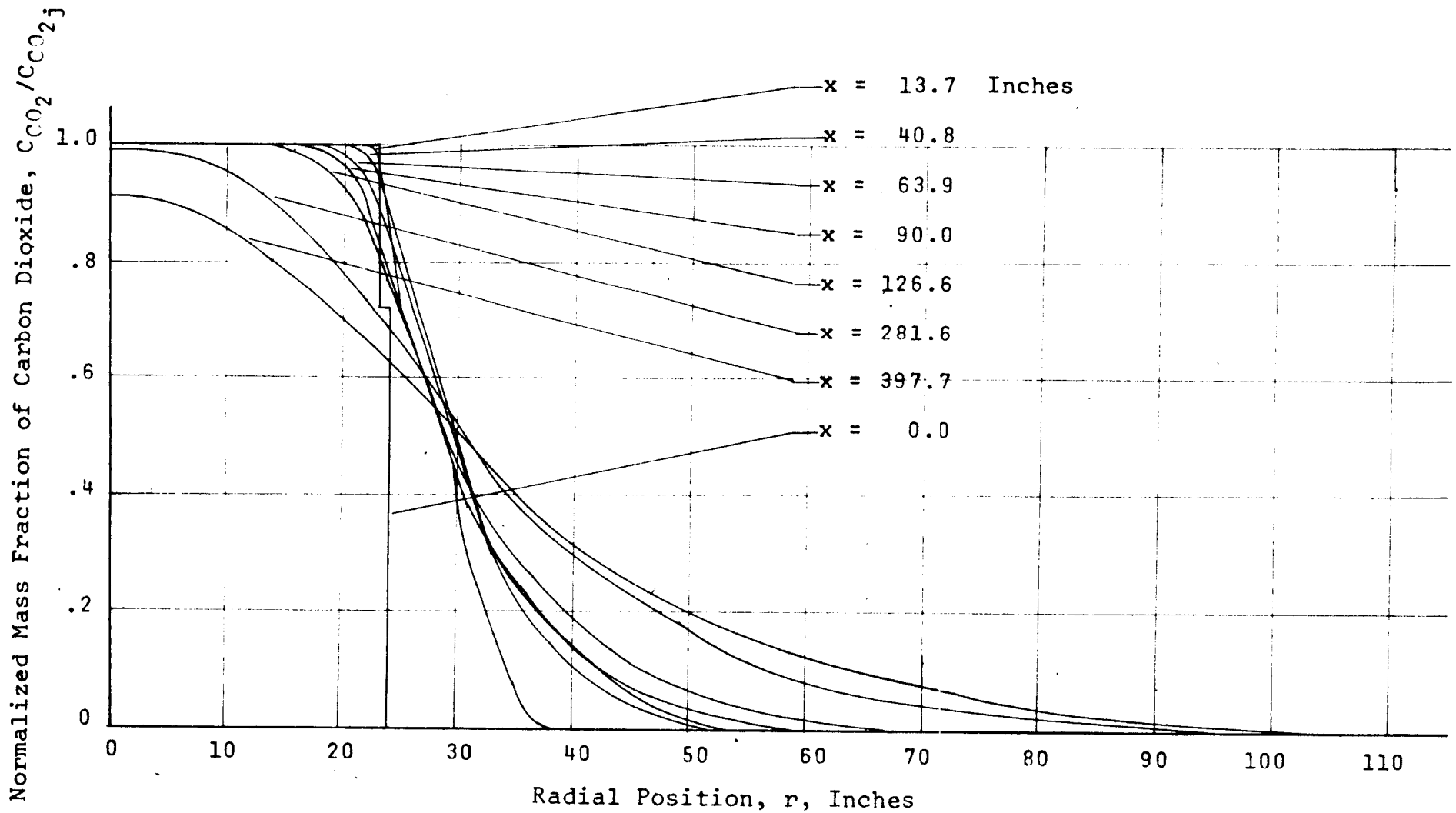

FIG. 9 MASS FRACTION OF CARBON DIOXIDE PROFILE FOR THREE-STREAM-MIXING

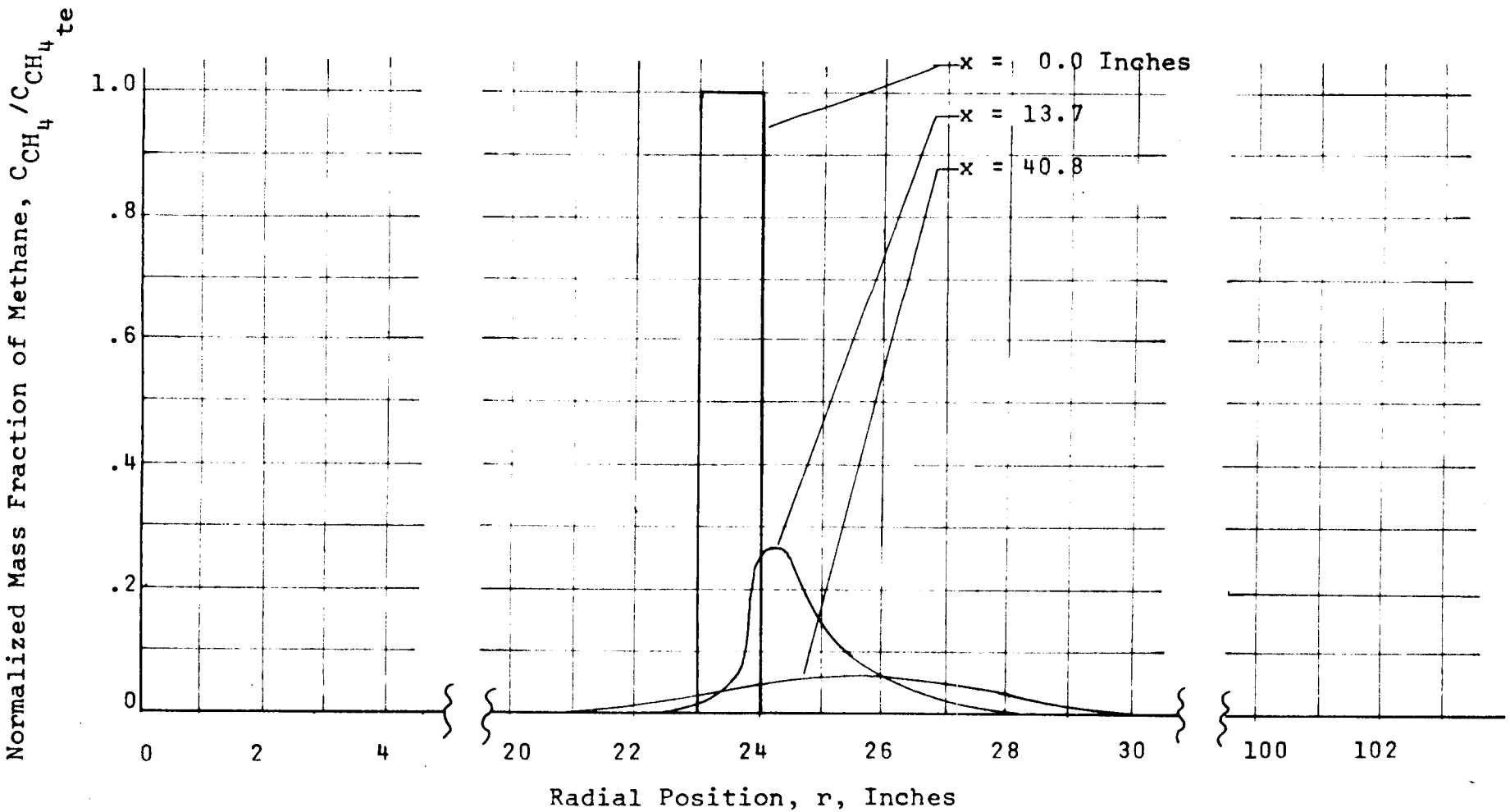

FIG. 10 MASS FRACTION OF METHANE PROFILE FOR THREE-STREAM-MIXING

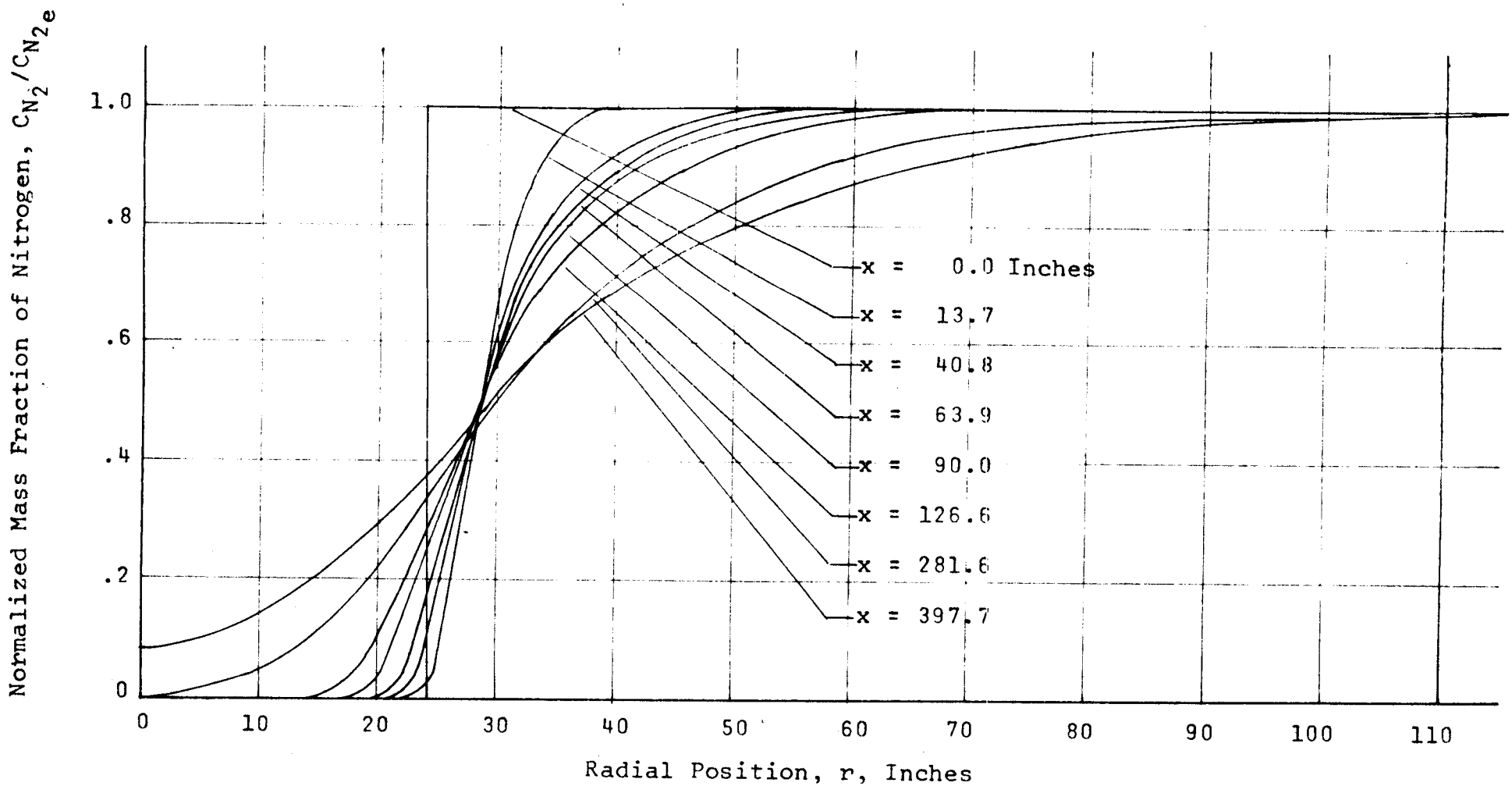

FIG. 11 MASS FRACTION OF NITROGEN PROFILE FOR THREE-STREAM-MIXING

NRO=4 NSMAX=7 NRS=3
NRO1=5 NSMAX1=8 NRS1=4
NRS=NRS=NSMAX1—NRO1

GENERAL CARD DESIGN

| | | |

GENERAL CARD DESIGN

PSI, DPSI, DPSIBAR, XI, DXI, PSIMAX	0.0	0.05	0.01	.0025	0.005	1.4
XIMAX, TREF, PREF, REFV, HE, HJ	0.0001	281.11	1.100505E-3	0.034670	125.3393436	73.382
ACCTOL, PSIJ, T, DELTAT, DT, TOL	10.0	1.0	2600.0	100.0	100.0	10.0
PSIBAR, UJ, UE, USTARJ, RHOE, RJ	0.0	293126.1	9144.0	0.0	9.1071203E-5	24.28401
CNITROM	28.0					
AACON (I,J,1)	3.0798700	5.421663E-4	3.848762E-8	-3.46338E-11	3.67191E-15	-8.669205E+2
I, =1,6 J=1, NSMAX	3.5980995	7.805671E-4	-2.233751E-7	4.23966E-11	-3.34191E-15	-1.192975E+3
(Higher Temperature	2.6707532	3.031711E-3	-8.535157E-7	1.17908E-10	-6.19735E-15	-2.988899E+4
Range)	1.3522089	1.878796E-3	-7.822261E-7	1.553577E-10	-1.18144E-14	-645.83539
	2.9537179	1.547681E-3	-6.160384E-7	1.12728E-10	-7.72089E-15	-1.423281E+4
	4.4008955	3.216385E-3	-1.313466E-6	2.45482E-10	-1.70946E-14	-4.894009E+4
	1.1795744	1.095059E-2	-4.062213E-6	7.13703E-10	-4.74904E-14	-9.855663E+3
AACON (I,J,2)	2.8612126	4.063282E-3	-9.192602E-6	8.94899E-09	-3.05132E-12	-9.686986E+2
I=1,6 J=1, NSMAX	3.7189946	-2.516728E-3	8.583935E-6	-8.29987E-09	2.70822E-12	-1.057671E+3
(Lower Temperature	4.1565016	-1.724433E-3	5.698231E-6	-4.59300E-09	1.42336E-12	-3.028877E+4
Range)	-.64250569	6.826247E-3	-4.225572E-6	1.522231E-10	4.81929E-13	-75.029564
	3.7929971	-2.215637E-3	5.193912E-6	-3.603526E-9	8.22622E-13	-1.436408E+4
	2.1974487	1.016306E-2	-1.016489E-5	5.724671E-9	-1.38808E-12	-4.835513E+4
	4.2497878	-6.912656E-3	3.160213E-5	-2.971543E-8	9.51036E-12	-1.018663E+4
AAHOT (I,J) I=1,6	2.8609546	1.585176E-3	-6.177990E-7	1.11194E-10	-7.52106E-15	-8.924897E+2
J=1,2	3.6927389	-1.341199E-3	2.669338E-6	-9.95596E-10	-9.37266E-14	-1.062950E+3
Aθ, PLYC(I) I=1,20	1.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0			

FIG. 13 TWO-STREAM INPUT-B

GENERAL CARD DESIGN

TTAB(K) K=1, KT	600.0	800.0	1000.0	1200.0	1400.0	1600.0
	1800.0	2000.0	2200.0	2400.0	2600.0	2800.0
	3000.0	3200.0	3400.0	3600.0	3800.0	4000.0
	4200.0	4400.0				
STAB(I,J) I=1, KT	0.0	0.0	0.0	0.0	0.0	0.0
J=1, NSMAX	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	18.633	13.289
	10.062	7.899	6.347	5.180	4.270	3.540
	2.942	2.443	2.021	1.658	1.343	1.067
	0.842	0.607	0.413	0.238	0.079	0.065
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	14.318	11.914	10.459	9.479
	8.771	8.234	7.811	7.469	7.185	6.946
	6.741	6.563	6.407	6.269	6.145	6.034
	5.933	5.841	5.756	5.679	34.405	25.830
	20.680	17.243	14.785	12.940	11.504	10.353
	9.411	8.625	7.960	7.388	6.892	6.458
	6.074	5.732	5.425	5.149	4.898	4.670
	2.001	0.146	-1.011	-1.801	-2.372	-2.803

MSFC - Form 353 (July 1960)

FIG. 14 TWO-STREAM INPUT - C

GENERAL CARD DESIGN

	-3.139	-3.408	-3.627	-3.809	-3.962	-4.093
	-4.206	-4.304	-4.391	-4.467	-4.536	-4.598
	-4.653	-4.704				
KSPEC (I,J) I=1,4	1 0	0 0	0 2	0 0	1 2	0 0
J=1, NSMAX	4 0	0 0	4 2	0 0	0 4	2 0
	0 4	1 0				
XG(I) I=1,..., NSMAX	0.011943	0.0000548	0.27855	1.0 E-05	0.40266	0.30681
	1.0 E-05					
XG(I) CUT-OFF	1.0 E-40					
SUM	0.0					

MSFC - Form 193 (July 1960)

FIG. 15 TWO-STREAM INPUT-D

NRO=4 NSMAX=7 NRS=3
NRO1=5 NSMAX1=8 NRS1=4
NRS=NRS=NSMAX1—NRO1

GENERAL CARD DESIGN

| | | |

GENERAL CARD DESIGN

PSI, DPSI, DPSIBAR, XI, DXI, PSIMAX	0.0	0.05	0.01	.0025	0.005	1.4
XIMAX, TREF, PREF, REFV, HE, HJ	0.0001	281.11	1.100505E-3	0.034670	125.3393436	73.382
ACCTOL, PSIJ, T, DELTAT, DT, TOL	10.0	1.0	2600.0	100.0	100.0	10.0
PSIBAR, UJ, UE, USTARJ, RHOE, RJ	0.0	293126.1	9144.0	0.0	9.1071203E-5	24.28401
CNITROM	28.0					
AACON (I,J,1)	3.0798700	5.421663E-4	3.848762E-8	-3.46338E-11	3.67191E-15	-8.689205E+2
I, =1,6 J=1, NSMAX	3.5980995	7.805671E-4	-2.233751E-7	4.23966E-11	-3.34191E-15	-1.192975E+3
(Higher Temperature	2.6707532	3.031711E-3	-8.535157E-7	1.17908E-10	-6.19735E-15	-2.988899E+4
Range)	1.3522089	1.878796E-3	-7.822261E-7	1.553577E-10	-1.18144E-14	-645.83539
	2.9537179	1.547681E-3	-6.160384E-7	1.12728E-10	-7.72089E-15	-1.423281E+4
	4.4008955	3.216385E-3	-1.313466E-6	2.45482E-10	-1.70946E-14	-4.894009E+4
	1.1795744	1.095059E-2	-4.062213E-6	7.13703E-10	-4.74904E-14	-9.855663E+3
AACON (I,J,2)	2.8612126	4.063282E-3	-9.192602E-6	8.94899E-09	-3.05132E-12	-9.686986E+2
I=1,6 J=1, NSMAX	3.7189946	-2.516728E-3	8.583935E-6	-8.29987E-09	2.70822E-12	-1.057671E+3
(Lower Temperature	4.1565016	-1.724433E-3	5.698231E-6	-4.59300E-09	1.42336E-12	-3.028877E+4
Range)	-.64250569	6.826247E-3	-4.225572E-6	1.522231E-10	4.81929E-13	-75.029564
	3.7929971	-2.215637E-3	5.193912E-6	-3.603526E-9	8.22622E-13	-1.436408E+4
	2.1974487	1.016306E-2	-1.016489E-5	5.724671E-9	-1.38808E-12	-4.835513E+4
	4.2497878	-6.912656E-3	3.160213E-5	-2.971543E-8	9.51036E-12	-1.018663E+4
AAHOT (I,J) I=1,6	2.8609546	1.585176E-3	-6.177990E-7	1.11194E-10	-7.52106E-15	-8.924897E+2
J=1,2	3.6927389	-1.341199E-3	2.669338E-6	-4.95596E-10	-9.37266E-14	-1.062950E+3

FIG. 17 THREE-STREAM INPUT-B

GENERAL CARD DESIGN

PSI TABLE	0.0	0.938129	0.942866	0.943818	0.944790	0.945856
	0.947213	0.949147	0.951969	0.955824	0.961346	0.961796
	0.963856	0.965916	0.967975	0.970035	0.972095	0.974154
	0.976214	0.978273	0.980333	0.982392	0.984452	0.986743
	0.987445	0.988169	0.988891	0.989742	0.990862	0.992197
	0.993657	0.994900	0.995807	0.996533	0.997130	0.997703
	0.998277	0.998851	0.999429	1.0		
MODIFICATION FACTOR	1.0	1.0	0.99981	0.99788	0.984651	0.92887
FOR VELOCITY	0.78326	0.54804	0.312946	0.16754	0.111893	0.096492
	0.096517	0.096534	0.096546	0.096555	0.096562	0.096574
	0.096571 6	0.096575	0.096718	0.096580	0.096474	0.090383
	0.084229	0.074757	0.062155	0.047666	0.03326	0.020894
	0.011709	0.005815	0.002546	0.000979	0.000329	0.000097
	0.000025	0.000006	0.000001	0.0		
MODIFICATION FACTOR	1.0	1.0	1.000653	1.007557	1.054814	1.254035
FOR FIRST ELEMENT	1.774118	2.614177	3.453863	3.973176	4.171930	4.221940
	4.223036	4.223759	4.224296	4.224692	4.224972	4.225205
	4.225392	4.225532	4.225672	4.225742	4.221123	3.954632
	3.685366	3.270923	2.719561	2.085557	1.455449	0.914196
	0.512320	0.256058	0.111385	0.042813	0.014404	0.004231
	0.001082	0.000240	0.000047	0.0		
MODIFICATION FACTOR	1.0	1.0	0.999906	0.998936	0.992273	0.964196
FOR SECOND ELEMENT	0.890860	0.772421	0.654038	0.580820	0.552798	0.544504
	0.544643	0.543978	0.544808	0.544857	0.544896	0.544925
	0.544951	0.544968	0.544983	0.544996	0.544400	0.510030

MSFC - Form 383 (July 1960)

FIG. 18 THREE-STREAM INPUT-C

GENERAL CARD DESIGN

	0.475302	0.421850	0.350740	0.268975	0.187711	0.117904
	0.066075	0.032814	0.014364	0.005521	0.001858	0.000545
	0.000139	0.000032	0.000006	0.0		
MODIFICATION FACTOR	1.0	1.0	1.000078	1.000663	1.004939	1.022886
FOR THIRD ELEMENT	1.069736	1.145414	1.221057	1.267840	1.285744	1.289259
	1.289591	1.289817	1.28998	1.290098	1.290188	1.290258
	1.290316	1.290359	1.290398	1.290426	1.289013	1.207635
	1.125407	0.99885	0.830475	0.636869	0.444458	0.279170
	0.156420	0.077695	0.0340139	0.013074	0.004399	0.001292
	0.000331	0.000074	0.000014	0.0		
MODIFICATION FACTOR	1.0	1.0	1.0	1.0	1.0	1.0
FOR FOURTH ELEMENT	1.0	1.0	1.0	1.0	1.0	1.0
	1.0	1.0	1.0	1.0	1.0	1.0
	1.0	1.0	0.99998	0.99982	0.99872	0.93568
	0.87196	0.77390	0.64345	0.49345	0.34436	0.21630
	0.121218	0.06020	0.02635	0.010130	0.003408	0.001002
	0.000258	0.000057	0.0	0.0		
MODIFICATION FACTOR	1.0	1.0	1.004446	1.050600	1.367034	2.701137
FOR TOTAL ENTHALPY	6.183883	11.809516	17.432454	20.910176	22.241122	22.582366
	22.588179	22.592143	22.594973	22.597071	22.598649	22.599881
	22.600862	22.601651	22.602306	22.602787	22.578036	20.459765
	19.712397	17.495564	14.546394	11.155282	7.784995	4.889861
	2.740304	1.360875	0.595781	0.228997	0.077044	0.022634
	0.005793	0.001290	0.000250	0.0		

MSFC - Form 383 (July 1966)

FIG. 19 THREE-STREAM INPUT-D

GENERAL CARD DESIGN

	-3.139	-3.408	-3.627	-3.809	-3.962	-4.093
	-4.206	-4.304	-4.391	-4.467	-4.536	-4.598
	-4.653	-4.704				
KSPEC (I,J) I=1,4	1 0	0 0	0 2	0 0	1 2	0 0
J=1, NSMAX	4 0	0 0	4 2	0 0	0 4	2 0
	0 4	1 0				
XG(I) I=1,..., NSMAX	0.011943	0.0000548	0.27855	1.0 E-05	0.40266	0.30681
	1.0 E-05					
XG(I) CUT-OFF	1.0 E-40					
SUM	0.0					

MSFC - Form 153 (July 1966)

FIG. 21 THREE-STREAM INPUT-F

GENERAL CARD DESIGN

TTAB(K) K=1, KT	600.0	800.0	1000.0	1200.0	1400.0	1600.0
	1800.0	2000.0	2200.0	2400.0	2600.0	2800.0
	3000.0	3200.0	3400.0	3600.0	3800.0	4000.0
	4200.0	4400.0				
STAB(I,J) I=1, KT	0.0	0.0	0.0	0.0	0.0	0.0
J=1, NSMAX	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	18.633	13.289
	10.062	7.899	6.347	5.180	4.270	3.540
	2.942	2.443	2.021	1.658	1.343	1.067
	0.842	0.607	0.43	0.238	0.079	0.065
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	14.318	11.914	10.459	9.479
	8.771	8.234	7.81	7.469	7.185	6.946
	6.741	6.563	6.407	6.269	6.145	6.034
	5.933	5.841	5.756	5.679	34.405	25.830
	20.680	17.243	14.785	12.940	11.504	10.353
	9.411	8.625	7.960	7.388	6.892	6.458
	6.074	5.732	5.425	5.149	4.898	4.670
	2.001	0.146	-1.011	-1.801	-2.372	-2.803

FIG. 20 THREE-STREAM INPUT-E