POLLUTANT PRODUCTION IN A SIMULATED TURBOJET AFTERBURNER

Part II. COMPUTER PROGRAM FOR CALCULATION OF POLLUTANT HISTORY IN AFTERBURNING TURBOJET ENGINES

The University of Tennessee Space Institute

Technical Report AFAPL-TR-71-66; PART II.
February 1972

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED

Air Force Aero Propulsion Laboratory
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio

Reproduced by NATIONAL TECHNICAL INFORMATION SERVICE
Springfield, Va 22151
POLLUTANT PRODUCTION IN A SIMULATED TURBOJET AFTERBURNER
PART II: COMPUTER PROGRAM FOR CALCULATION OF
POLLUTANT HISTORY IN AFTERBURNING TURBOJET ENGINES


APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED
FOREWORD

This research program was initiated 1 November 1970 under USAF Contract F33615-71-C-1125. The work was performed between 1 November 1970 and 31 July 1971 by The University of Tennessee Space Institute, Tullahoma, Tennessee, and administered by the Air Force Aero Propulsion Laboratory, Wright-Patterson Air Force Base, Ohio, Mr. Kenneth N. Hopkins (AFAPL/TBC), Project Engineer. The contract was initiated under Project 3066, "Gas Turbine Technology," Task 306605 "Combustion Systems Performance and Stability."

Scientific and engineering personnel of the Space Institute, identified as "et al." on the cover, who engaged in the experimental work and preparation of this report, are:

Dr. B. H. Goethert, Dean, The University of Tennessee Space Institute

Mr. D. McClure, Research Assistant

Mr. E. Barnert, Research Assistant.

The authors wish to express sincere appreciation to Mr. R. Kamm, Special Assistant to Dr. Goethert, for his active cooperation in the project; to Mr. R. Nygaard, Engineer, and to Mr. J. Goodman, Mr. J. Boazman, and Mr. J. Rothert, Technicians, for their cooperation and assistance in the Combustion Laboratory; to the personnel at ESF and CCO of ARO, Inc., especially Mr. J. McCabe and Mr. W. Armstrong, for their cooperation.

This report was submitted by the authors in August, 1971.

This report is bound in two parts. Part I contains technical details of the report. Part II consists of a User's Manual for the computer program developed for calculating pollutant production in a turbojet afterburner.

The program has been operated on the IBM 370/155 using FORTRAN IV G Level 18. To keep within B partition it was necessary to overlay INIT with INTEG, SETUP, SETYP, SETHP, GAUSS. All pages of this manual are effective 31 July 1971.

Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

[Signature]

ERNEST C. SIMPSON
Director, Turbine Engine Division
Air Force Aero Propulsion Laboratory
ABSTRACT

An experimental and theoretical study has been made of the history of the pollutants carbon monoxide (CO), unburned hydrocarbons (HC) and nitrogen oxides (NO) in a turbojet afterburner. Experimental traverses at several axial stations were performed in a simulated afterburner in which exhaust from a J-47 combustor can, operated at medium power, was mixed with fuel spray. Experiments were carried out both in a non-bypass and in a bypass configuration (secondary air was mixed with primary exhaust). The non-bypass tests were carried out at high combustor efficiency, and yielded the following: CO = 300 ppm, HC less than 10 ppm, NO = 100 ppm. In the bypass tests, fuel distribution was nonuniform and combustor efficiency was low. The concentrations obtained were CO = 10,000 ppm, HC = 1000 ppm, NO = 100 ppm. The theoretical analysis consisted of a computer program for reacting flow with turbulent mixing. The computer program was very slow and therefore of limited usefulness in terms of cost and questionable results, since it could not be checked against experiment. Infrared measurements of NO in the combustion tunnel were attempted. Indications were obtained of NO at the 5.3 micron band, but quantitative measurements were not obtained.
TABLE OF CONTENTS

SECTION .................................................. PAGE

I.  INTRODUCTION ........................................ 1

II. COMPUTER PROGRAM SYSTEM CAPABILITIES .......... 3

   2.1 Purpose .............................................. 3
   2.2 General Description of the system ................. 3
   2.3 Functions Performed ................................ 4
   2.4 Function Descriptions .............................. 4
   2.5 Usage INstructions ................................. 20
   2.6 Operating Instructions ............................. 24

III. LISTING OF PROGRAM ................................. 31

Preceding page blank
GLOSSARY

R  Radial position - inches
T  Temperature - °K
S  Temperature - °R
U  Velocity - ft/sec
Y  Species mass fractions
ROU  Mass flux - lb/(sec sq. ft.)
PSI  Stream function \( \sqrt{\text{lb/sec}} \)
UPR  Velocity at previous location - ft/sec
YPR  Species mass fraction at previous location
HST  Static enthalpy  Btu/lb
HT  Total enthalpy  Btu/lb
TPR  Temperature at previous location °R
HTPR  Total enthalpy at previous location Btu/lb
CM  Molecular weight (average)
X  Mole fraction
H  Species chemical enthalpy  Btu/lb mole
W  Species production rate  lb/lb
E  Species production per  lb/lb ft
YO  Initial fuel mass fraction
TO  Time since start of burning
TAU  Burning time
ISP  Number of species
IRE  Number of reactions
IMU  IMU = Number of 3 body reactions
UM  Turbulent viscosity  lb/ft sec
SECTION I

INTRODUCTION

This users manual was prepared under contract number F33615-71-C-1125, and represents Item Number 0001 Sequence Number A002 under requirements of that contract. The purpose of writing the computer program was to provide the Air Force with a means of estimating air pollution concentrations in the exhaust gases from afterburning turbojet engines.
SECTION II

COMPUTER PROGRAM SYSTEM CAPABILITIES

2.1 PURPOSE

The purpose of this computer program is to solve the partial differential equations of change for momentum, species, and enthalpy, to obtain radial and axial profiles of velocity, temperature, and species concentrations in a turbulent reacting fluid. A finite difference approximation to the equations is made.

The chemical species considered and chemical reactions involved are those pertinent to burning of a liquid hydrocarbon fuel in a hot exhaust gas from a primary combustor. By controlling the input a calculation of the mixing and combustion in a bypass type engine may also be made. While the primary purpose of the research has been to estimate levels of the pollutants NO, CO, and unburned hydrocarbons, in exhaust gases, the levels of several other species must be taken into account in order to calculate pollutant levels.

2.2 GENERAL DESCRIPTION OF THE SYSTEM

A FORTRAN program has been written to achieve the purpose described above. The differential equations, transformed from (length, radius), to (length, stream function) coordinates, have been approximated by finite difference equations.

Initially, certain thermodynamic data for the chemical species are read, followed by data necessary to calculate chemical reaction rates. Then, initial data of temperature, velocity, and species concentrations as a function of radial position in an axisymmetric flow are read in. The program does not read any more data.

From the initial profiles as a function of radius, a conversion to the stream function variable is made, with appropriate interpolation.

An iteration scheme is used to solve for dependent variables (velocity, enthalpy, species concentration) as a function of axial distance. The program proceeds stepwise from the initial condition, until a maximum axial distance is reached. The outer boundary condition is an inviscid reacting flow. When the flow variables at the stream function location adjacent to the inviscid flow deviate from the inviscid flow by 1 part in $10^4$, the viscous flow region is expanded by one step in stream function. The number of stream function
steps is allowed to increase to 50, if necessary. If more expansion is needed, the stream function step size is doubled, and number of steps is halved. This process repeats itself as necessary.

2.3 FUNCTIONS PERFORMED

There are 13 subdivisions of the program.

1. MAIN Program
2. Subroutine SPECIE (KK1, KK2, NU)
3. Subroutine THERMO (KRIS, JA, IT)
4. Subroutine INIT
5. Subroutine TRANSP (NLOW, NHIGH)
6. Subroutine INTEG (NLOW, NHIGH, IN)
7. Subroutine SETUP (RUP)
8. Subroutine SETYP (RUP)
9. Subroutine SETHP (RUP)
10. Subroutine GAUSS
11. Subroutine CHANGE (RUP)
12. Subroutine PRINT (KUSK, KUN)
13. Subroutine CHEMIE (KRYS, KRAS, IJ)

The MAIN program controls the flow of information, causing THERMO, CHEMIE, and INIT to read in initial data. It then proceeds with the computation and prints out results at every 50th axial station, until stopped by either reaching the maximum desired axial distance (ZMAX feet) or exceeding the time limitation.

2.4 FUNCTION DESCRIPTIONS

2.4.1 Main Program

Summary of Operations performed by MAIN

The main program first sets certain parameters, namely ISP = 14, IRE = 17, IMU = 11, which are in labelled common ZZZ. There are, respectively, the number of species, the number of reactions, and the number of the last reaction which involves three bodies (see CHEMIE). In addition, the variable TAU is set equal to $0.4 \times 10^4$, which is the hydrocarbon burning time in seconds. TAU is in labeled common XXX. Certain other parameters are also set, so as to cause THERMO to read thermodynamic data, and CHEMIE to read chemical reaction rate data. The main program then calls upon INIT to read in velocity, temperature, and species concentrations as a function of radius. The first item actually read is NNN, which defines the number of radial locations considered. INIT converts the indepen-
dent variable from radius to stream function, and interpolates to
obtain initial dependent variables, velocity, temperature, and
species concentrations, at equal intervals of stream function.
The stream function interval is denoted by $\Delta \psi$. $\Delta x$ is the
interval in feet, and is defined to be 0.00001. $RUP = \Delta x/\Delta \psi$ is defined. $XMAX = 0.7$ (feet) is defined as the maximum axial dis-
tance to be considered.

The main program calls upon SPECIE to calculate mole fractions,
molecular weights, and mass fluxes at all radial locations, and
calls upon THERMO to calculate static enthalpies of the species.
Total enthalpies are calculated. The TRANSP subprogram is called
upon to calculate a turbulent viscosity, and INTEG to calculate ra-
dius as a function of stream function, and also to calculate certain
coefficients ($AP, APR$) needed for subsequent finite difference cal-
culations.

The main program writes the data on initial profiles as a function
of stream function. It then proceeds in a stepwise fashion to cal-
culate the flow variables as a function of axial location.

After the writing of the initial data, there is very little necessity
or possibility for operator or programmer intervention in the main
program. The number of radial locations, $NNN$, is a major variable
which is controlled initially and internally. The variables $IK$ and
$IA$ define the locations of the inner and outer bounds of the viscous
flow region. In the present application, the inner bound is taken
to be the center-line. The outer bound is always $IA = NNN-1$.
Species calculations run from 1 to $ISP$, which is defined previously.

A variable $KUSK$ defines the number of iterations which have been per-
formed at an axial location. The iterations stop after $KUSK = 3$.
Prior to the start of calculations, the axial position, $Z$, is incre-
mented. $IA$ is checked to see if the maximum number of streamlines
permitted has been reached. If $IA = NILS - 1$, a doubling of stream
function step size is incurred, together with a reduction of the
number of steps considered. This is done through the subroutine
CHANGE. In the present program, $NILS = 51$ is defined as data.
NILS must be odd. The resulting profiles are written after CHANGE.

Whether or not a change occurs, the boundary values at $NNN$ are cal-
culated at the incremented axial position. The boundary value for
velocity, $UEO = U(NNN)$ never changes since the inviscid flow is con-
sidered to occur at constant pressure. The velocities at the other
radial locations are calculated through SETUP and GAUSS. If $|u(IA) -
UEO)/UEO| > TOL$, an indicator, $IAP$ is set up to tell the program to
increase the viscous region (this is not done until after all iter-
ations are complete). The species concentrations are calculated at
$NNN$, and then in the viscous region through SETUP and GAUSS. If
the relative difference in molecular weights at $IA$ and $NNN$ exceeds
$TOL$, the indicator $IAP$ is set. The total enthalpy at $NNN$, $HT(NNN)$
$= HEO$ is constant in the inviscid flow. In the viscous region, new
total enthalpies, HT(I) are calculated through SETUP and GAUSS. If the relative difference in total enthalpies at IA and NNN exceeds TOL, the indicator IAP is set.

In the first iteration, species production terms at the previous location are used. In the second and third iterations the species production terms at the midpoint of the interval are approximated by a Taylor's series expansion. The same process of calculating boundary values, then viscous flow values, of the flow variables is followed. In between iterations, TRANSP and INTEG are called upon to obtain improved values of turbulent viscosity and the terms AP, APR used in SETUP, SETYP, SETHP.

After the third iteration, IAP is checked to see if the flow field should be expanded. If it should be it is expanded. If it should not, previous values of the flow variables are identified, and the program returns to increment another axial step. The results are printed every 50 steps, as defined by the variable KYS. KYS is set to 50 near the start of the main program. If this causes too much printing, KYS could be set to a larger value.

Inputs to MAIN

There are no inputs read into MAIN by tape, cards, etc.

Expected Outputs

After INIT, the MAIN program writes initial profiles of selected flow variables as a function of stream function.

After CHANGE, the MAIN program writes resultant profiles of selected flow variables.

Other writing is done by other subroutines.
2.4.2 SPECIE (KK1, KK2, NU)

Purpose and Uses of SPECIE

SPECIE is capable of calculating CM(I) (average molecular weight), X(J,I) (mole fractions of ISP species) and ROU(I) (mass flux), from axial location KK1 to KK2

If NU = 0, CM(I) and X(J,I) are transmitted
If NU < 0, ROU(I) is also transmitted
If NU = 0, only ROU(I) is transmitted

Inputs

There are no system inputs on cards or tape to SPECIE

Expected output

There are no system outputs on cards or tape from SPECIE
Summary of Operations Performed by THERMO

In the initial phases of the program, JA is set to be greater than zero, which causes THERMO to read in coefficients for calculating species enthalpies. When JA is less than or equal to zero, and KRIS equals 100, species enthalpies are calculated at T(IT) (temperature allocation IT); if KRIS is different from 100, temperature is determined at a static enthalpy HST(IT), by means of a Newton-Raphson process.

System Inputs

A total of ISP sets of coefficients A, followed by ISP coefficients B. These are, respectively, the 300-1000°K and 1000-5000°K coefficients $a_i - a_0$ from Table V of NASA SP-3001, for the appropriate compounds. For fuel, a value of -23900R calories per gram mole of C$_{10}$H$_{22}$ has been estimated to be the enthalpy of fuel at 477°F (estimated boiling point) from the elements at 298°K. The program also reads SENS, heats of formation at 298°K. These are not used in the present program.

System Outputs

The coefficients B are printed after being read in.
2.4.4 INIT

Summary of Operations Performed by INIT

INIT is called only once. It reads in NNN, the number of streamlines considered, then NNN cards containing UE(I) (velocity) SE(I) (temperature) and R(I) (radius) with R(I) starting at zero and increasing monotonically. Initial profiles of species mass fractions (YE(J,I)) are then read in at R(I). Stream functions are calculated for the initial profile, and the data are interpolated to equal intervals of stream function with the interval DPSI = PSI/(NNN-1), where PSI(I) is the value of stream function at location I.

System Inputs

NNN is an integer less than or equal to NILS (main program). UE(I) is in feet per second, SE(E) in degrees, Rankine, R(I) in inches. Species mass fractions are introduced per card up to ISP, for each R(I).

System Outputs

Nothing is written on card or tape from INIT.
2.4.5 TRANSP (NLOW, NHIGH)

Summary of Operations Performed by TRANSP

TRANSP calculates a turbulent viscosity, \( \mu_{\text{M}} \), according to Schetz's displacement thickness model, as described in the report. A DATA statement defines \( R_J \), an inner jet radius, in feet.

System Inputs and Outputs

Nothing is read or written by TRANSP.
2.4.6 INTEG (NLOW, NHIGH, IN)

Summary of Operations Performed by INTEG

INTEG calculates radial values (Rll) as a function of stream function. It also calculates AP(I) and APR(I) at locations NLOW to NHIGH.

If IN = 0, AP = APR = SP
If IN is not zero, AP = .5 (SP + APR)

where SP = UM * R(I) **2 * ROU(I)/PSI(I)

is a term which arises in the transformation of equations to stream function coordinates.

System Inputs and Outputs.

No information is read to or written from INTEG.
2.4.7 SETUP (RUP)

Summary of Operations Performed by SETUP

SETUP calculates coefficients for solution of the finite difference equations for velocity. These are transmitted to labelled COMMON UPSET and used by GAUSS. The indicator KOR is set equal to -10.

System Inputs and Outputs

There are no INPUT/OUTPUT Statements in SETUP.
2.4.8 SETYP (RUP)

Summary of OperationsPerformed by SETYP

SETYP calculates coefficients for solution of the finite difference equations for species mass fractions, to be used by GAUSS. The indicator KOR is set to 0. A DATA statement defines CN which is the inverse of the turbulent Prandtl number. CN is taken to be 1 in the present study.

System Inputs and Outputs

There are no INPUT/OUTPUT Statements in SETYP.
2.4.9 SETHP (RUP)

Summary of Operations Performed by SETHP

SETHP calculates coefficients for solution of the finite difference equations for total enthalpy, to be used by GAUSS. The indicator KOR is set to +10. A DATA statement sets CN = 1 as in SETYP, in the present program.

System Inputs and Outputs

There are no INPUT/OUTPUT statements in SETHP.
2.4.10 GAUSS

Summary of Operations Performed by GAUSS

Depending on whether KOR (see previous three subroutines) is negative, zero, or positive, GAUSS solves for velocity (U), species mass fractions (Y(J,I)) or total enthalpy (HT) through the viscous flow field. When KOR = 0, the mole fractions (X(J,I)) and average molecular weights (CM) are also determined. When KOR is positive, static enthalpy (HST) is also determined. In addition, GAUSS determines the outermost location at which the flow variables U, CM, HT deviate significantly from the inviscid external flow. This is defined to be IAP, as discussed in MAIN.

The highest location at which significant deviation occurs in each branch is defined to be NO(I), I = 1, 2, 3. Quantities NE(I), defining the innermost location at which the flow variables deviate significantly from an inviscid center flow, are also defined but are not used in the present program. An inner, reacting, inviscid flow could not be included in the program without some modification (essentially everything done in the stepwise calculations at NNN would have to be done at the inner boundary). An inner equilibrium flow could be included with slight modification.

The "significant deviation" mentioned above is defined in DATA as TOL, and is taken to be 1.0 x 10^-4 in the present program.

System Inputs and Outputs

There are no INPUT/OUTPUT statements in GAUSS.
2.4.11 CHANGE (RUP)

Summary of Operations Performed by CHANGE

CHANGE is called upon when the viscous flow field has expanded to its maximum size. The value of DPSI (stream function step size) is doubled, and the number of intervals is halved. In the present program, NNN is reduced from 51 to 26. Values of the flow variables are also transferred so that the flow variables correspond to the proper stream function values.

System Inputs and Outputs

There are no INPUT Statements in CHANGE. A printed output indicates that the number of points has been halved, and gives values of certain variables which are the distance downstream from the start in inches, and the distance interval in inches. The values are duplicated because at one time, DELX was changed when DPSI was changed. This is no longer done.
2.4.12 PRINT (KUSK, KUN)

Summary of Operations Performed by PRINT

PRINT lists pertinent variables when called upon by the MAIN program.

System Output

The first data listed are

Z(foot), ZX(inches), DELX(foot), DPSI, NEL, NEU, IA, KUSK, KUN

Then at each of NNN locations, I(radial location), PSI, R, S(Temperature in °R) AP, APR, HT in one line, followed by Y(I,J) (2 lines) W(I,J) (2 lines) E(I,J) (2 lines) and a final line with YTOT, CM, ROU. This output represents the fundamental output for the program.
2.4.13 CHEMIE (KRYS, KRAS, IT)

Purpose and Uses of CHEMIE

This subroutine calculates quantities necessary to calculate the species production terms for inclusion in the species conservation equations. It also reads in the parameters for calculating reaction rates. If (KRYS + KRAS) is not less than zero, the parameters for the reactions rates are read in and listed. If (KRYS + KRAS) is less than zero, the reaction rates, species production, and linearization terms are calculated. If KRAS is greater than zero, species production terms (W) are calculated.

If KRAS is equal to zero, linearized terms are calculated.

The reactions considered are:

\[
\begin{align*}
\text{H} + \text{O}_2 & = \text{O} + \text{OH} \\
\text{H}_2 + \text{O} & = \text{OH} + \text{H} \\
\text{H}_2 + \text{OH} & = \text{H} + \text{H}_2\text{O} \\
\text{OH} + \text{OH} & = \text{H}_2\text{O} + \text{O} \\
\text{H}_2 + \text{M} & = \text{H} + \text{H} + \text{M} \\
\text{H}_2\text{O} + \text{M} & = \text{H} + \text{OH} + \text{M} \\
\text{N}_2 + \text{M} & = \text{N} + \text{N} + \text{M} \\
\text{O}_2 + \text{M} & = \text{O} + \text{O} + \text{M} \\
\text{NO} + \text{M} & = \text{N} + \text{O} + \text{M} \\
\text{N}_2\text{O} + \text{M} & = \text{N}_2 + \text{O} + \text{M} \\
\text{NO}_2 + \text{M} & = \text{NO} + \text{O} + \text{M} \\
\text{CO} + \text{OH} & = \text{CO}_2 + \text{H} \\
\text{NO} + \text{O} & = \text{N} + \text{O}_2 \\
\text{NO} + \text{N} & = \text{N}_2 + \text{O} \\
\text{NO} + \text{O}_2 & = \text{NO}_2 + \text{O} \\
\text{NO} + \text{NO} & = \text{N}_2\text{O} + \text{O} \\
\text{C}_n\text{H}_{2n+2} + \frac{n}{2} \text{O}_2 & \rightarrow \text{nCO} + (n+1)\text{H}_2 \\
\end{align*}
\]
System Inputs and Outputs

When called upon near the start of the program, CHEMIE reads data for calculating chemical reaction rates. These are, in order,

- CW (molecular weights)
- AU (collision terms in forward rate constants)
- CU (collision terms in backward reaction rate constants)
- E (forward rate activation energy/R)
- D (backward rate activation energy/R)
- GNUF(I,J) (stoichiometric coefficients for species I in forward reaction J)
- GNUB(I,J) (stoichiometric coefficients for species I in backward reaction J)
- AF, BF, AB, BB (respectively; temperature coefficient, density coefficient for forward reaction, similarly for reverse)
- EN (number of carbon atoms in fuel \( C_n \) \( H_{2n+2} \))

Certain of the above are listed after being read in, namely, CW, AU, CU, E, D, GNUF, GNUB.
2.5 Usage Instructions

2.5.1 Preparation of Inputs

Titles and Description of Inputs

The basic inputs to the program, initial profiles, are read by the subroutine INIT. The titles of the inputs, read in INIT, are

- **NNN** - Number of inputs, number of streamlines considered initially.
- **UE, SE, R** - One set per card.

**UE** is specified initial velocity, (ft/sec); specified initial temperature (degrees Rankin); at **R**, radial position in inches.

**YE(J,I)** A total of ISP species mass fractions, arranged in the order, H, O, H₂O, OH, O₂, H₂, N₂, N, NO, N₂O, NO₂, CO, CO₂, fuel at **R**. This requires two cards for the present value of ISP = 14.

The data are read on cards.

The cards must be read in the sequence:

- NNN Card
- **UE, SE, R** - Total of NNN cards
- **YE** - One set of cards for each **R** (2 NNN cards in the present program).

Limitations

The first **R** is taken to be zero. **R** must increase from card to card. Negative values for any variable would have no physical meaning.

Formats

```plaintext
READ(5,100) NNN
READ(5,101)(UE(L),SE(L),R(L),L=1,NNN)
READ(5,102)((YE(J,I),J=1,ISP),I=1,NNN)
```

```
100 FORMAT(15)
101 FORMAT(3F10.0)
102 FORMAT(7D10.0)
```
Relationship of Inputs to Outputs

These are the fundamental starting values for the differential equations. All output profiles are related to these.

In particular, selected flow variables are printed in MAIN directly after INIT. These are dependent on PSI, rather than R, as are all subsequent outputs of flow variables. Only values at the end points should be the same.

2.5.2 Results of Operation

Description of Results

The major results are profiles of velocity, temperature, and species mass fraction at specified axial locations. The output comes on the printer.

Format and Content

The output instruction and format are as follows

\[
\text{Z} = 2X/12. \\
\text{WRITE} (6, 30) Z, Z, Z, \text{DELX}, \text{DPSI}, \text{NEL}, \text{NEU}, \text{IKS}, \text{KUSK}, \text{KUN} \\
\text{DO} 20 \ I = 1, 14 \\
\text{YT} = 0, 0 \\
\text{DO} 10 \ J = 1, \text{ISP} \\
\text{YT} = \text{YT} + Y(J, I) \\
\text{WRITE} (6, 40) \ I, \text{PSI}(I), R(I), S(I), U(I), \text{AP}(I), \text{APR}(I), \text{HT}(I) \\
\text{WRITE} (6, 50) \ I, (Y(J, I), J = 1, \text{ISP}) \\
\text{WRITE} (6, 50) \ I, (W(J, I), J = 1, \text{ISP}) \\
\text{WRITE} (6, 50) \ I, (E(J, I), J = 1, \text{ISP}) \\
\text{WRITE} (6, 60) \ I, \text{YT}, \text{CM}(I), \text{RDU}(I) \\
\text{CONTINUE} \\
\text{WRITE}(6, 60) \ \text{KQ}, \text{UM} \\
\text{RETURN} \\
\text{C} \\
\text{FORMAT} (', 4(2X, F15, 10), 5(3X, 15), 7H \text{OUTPUT}, //) \\
\text{FORMAT} (', 2X, 14, 7(2X, E14, 7)) \\
\text{FORMAT} (2X, 14, 7(2X, E14, 7)) /, 6X, 7(2X, F14, 7)) \\
\text{FORMAT} (2X, 14, 3(2X, E14, 7))
\]

Use of Outputs

The outputs contain information on the pollutant concentrations, and other concentrations. Hence the use of output, at the end of the afterburner, will be to predict pollutant concentrations. Intermediate results indicate the evolution of the pollutants.

21
A Sample input is presented in this and the following page

<table>
<thead>
<tr>
<th>J</th>
<th>Jf</th>
</tr>
</thead>
<tbody>
<tr>
<td>286</td>
<td>2145</td>
</tr>
<tr>
<td>286.2145</td>
<td>0.368</td>
</tr>
<tr>
<td>285</td>
<td>2144</td>
</tr>
<tr>
<td>282</td>
<td>2136</td>
</tr>
<tr>
<td>278</td>
<td>2130</td>
</tr>
<tr>
<td>270</td>
<td>2120</td>
</tr>
<tr>
<td>262</td>
<td>2110</td>
</tr>
<tr>
<td>254</td>
<td>2100</td>
</tr>
<tr>
<td>254</td>
<td>2088</td>
</tr>
<tr>
<td>250</td>
<td>2076</td>
</tr>
<tr>
<td>246</td>
<td>2062</td>
</tr>
<tr>
<td>242</td>
<td>2050</td>
</tr>
<tr>
<td>238</td>
<td>2038</td>
</tr>
<tr>
<td>234</td>
<td>2022</td>
</tr>
<tr>
<td>232</td>
<td>2004</td>
</tr>
<tr>
<td>230</td>
<td>1980</td>
</tr>
<tr>
<td>228</td>
<td>1950</td>
</tr>
<tr>
<td>226</td>
<td>1910</td>
</tr>
<tr>
<td>224</td>
<td>1872</td>
</tr>
<tr>
<td>223</td>
<td>1840</td>
</tr>
<tr>
<td>223</td>
<td>1830</td>
</tr>
<tr>
<td>223</td>
<td>1830</td>
</tr>
<tr>
<td>223</td>
<td>1830</td>
</tr>
<tr>
<td>223</td>
<td>1830</td>
</tr>
<tr>
<td>223</td>
<td>1830</td>
</tr>
<tr>
<td>223</td>
<td>1830</td>
</tr>
</tbody>
</table>
2.6 Operating Instructions

2.6.1 Operating Procedures

Initiation of Program

The program is on cards, together with data. For a particular computer, proper control cards must be introduced. Input is on cards, output on printer.

Maintaining Computer Program Operation

The program will continue to operate, as long as too large a step size is not used. If too large a step size is used, the procedure is unstable, and is characterized by logs of negative numbers, due to temperature going negative. The program must be started with a smaller step size (DELX).

Termination and Restart

Termination is achieved when \( Z > Z_{\text{MAX}} \). If interruption of the program is necessary, some subroutine would have to be written to identify the necessity for interruption at statement 40, MAIN, store program and all data, and restart at the same point.

2.6.2 Operator Inputs

Initial inputs, other than the basic initial profile mentioned previously, include the thermodynamic and chemical data initially read in THERMO and CHEMIE.

The input statements and formats, and data used in the present version, are shown below.

In THERMO

```
180 READ (5,260) ((A(K,J),K=1,6),J=1,ISP)
DO 190 J=1,ISP
190 CONTINUE
READ (5,270) (B(K,J),K=1,6)
WRITE (6,280) (B(K,J),K=1,6)
WRITE (6,210) M, R(M), PSI(M), HTS(M), HS(M), FA, FB, TP, T(M)
GO TO 90
```

In CHEMIE

```
210 FORMAT (///,2X,14,B12X,E13.5),1X,4HTHER,///)
220 FORMAT (///,2(2X,15),612X,E15.8))
230 FORMAT (2X,15,412X,E15.8))
240 FORMAT (7X,15,712X,E15.8))
250 FORMAT (2X,15,712X,E15.8),//)
260 FORMAT (4D15.8,/,2D15.8)
270 FORMAT (7D10,0)
280 FORMAT (///,1X,6E20.8,///)
```
The Data used in the present version are as follows:

<table>
<thead>
<tr>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
<th>Value 4</th>
<th>Value 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.250000000E 01</td>
<td>0.25470497E 05</td>
<td>0.30218894E 01</td>
<td>0.21737248E 02</td>
<td>0.37542203E 05</td>
</tr>
<tr>
<td>0.90777547E 12</td>
<td>0.29137190E 05</td>
<td>0.4156016E 01</td>
<td>0.17244334E 02</td>
<td>0.56982316E 05</td>
</tr>
<tr>
<td>0.14233624E 11</td>
<td>0.30728870E 05</td>
<td>0.38234708E 02</td>
<td>0.11187229E 05</td>
<td>0.12466819E 05</td>
</tr>
<tr>
<td>0.52546551E 13</td>
<td>0.35852778E 04</td>
<td>0.37189946E 02</td>
<td>0.25167288E 02</td>
<td>0.85837353E 05</td>
</tr>
<tr>
<td>0.27082180E 11</td>
<td>0.10576706E 04</td>
<td>0.28460849E 01</td>
<td>0.41932116E 02</td>
<td>0.96119332E 05</td>
</tr>
<tr>
<td>0.33093421E 11</td>
<td>0.96729372E 03</td>
<td>0.36916148E 01</td>
<td>0.13325952E 02</td>
<td>0.26503100E 05</td>
</tr>
<tr>
<td>0.99772324E 13</td>
<td>0.10628336E 04</td>
<td>0.251479370E 02</td>
<td>0.11243791E 03</td>
<td>0.29647506E 00</td>
</tr>
<tr>
<td>0.12595466E 00</td>
<td>0.125612770E 05</td>
<td>0.41696476E 00</td>
<td>0.11197237E 02</td>
<td>0.96224670E 05</td>
</tr>
<tr>
<td>0.23821170E 01</td>
<td>0.10350556E 00</td>
<td>0.11167634E 04</td>
<td>0.69583165E 00</td>
<td>0.18701926E 03</td>
</tr>
<tr>
<td>0.33445630E 00</td>
<td>0.22234297E 00</td>
<td>0.71489750E 00</td>
<td>0.19727719E 05</td>
<td>0.37212530E 01</td>
</tr>
<tr>
<td>0.37871332E 00</td>
<td>0.21709526E 02</td>
<td>0.30757337E 00</td>
<td>0.34737728E 00</td>
<td>0.77216410E 00</td>
</tr>
<tr>
<td>0.21701000E 00</td>
<td>0.10378119E 00</td>
<td>0.63459175E 00</td>
<td>0.16280701E 11</td>
<td>0.48352602E 00</td>
</tr>
<tr>
<td>0.250000000E 01</td>
<td>0.25470497E 05</td>
<td>0.25372567E 01</td>
<td>0.18422190E 04</td>
<td>0.88017921E 08</td>
</tr>
<tr>
<td>0.55743608E 15</td>
<td>0.29232007E 05</td>
<td>0.26707532E 01</td>
<td>0.30317115E 02</td>
<td>0.83351570E 06</td>
</tr>
<tr>
<td>0.61973548E 14</td>
<td>0.29888894E 05</td>
<td>0.28895544E 01</td>
<td>0.98835061E 03</td>
<td>0.21879904E 06</td>
</tr>
<tr>
<td>0.38452940E 15</td>
<td>0.38811792E 04</td>
<td>0.35976129E 01</td>
<td>0.78415033E 03</td>
<td>0.22386670E 06</td>
</tr>
<tr>
<td>0.33460204E 15</td>
<td>0.11927918E 04</td>
<td>0.30436897E 01</td>
<td>0.61187110E 03</td>
<td>0.73993551E 08</td>
</tr>
<tr>
<td>0.25939791E 14</td>
<td>0.85469102E 03</td>
<td>0.28567671E 01</td>
<td>0.15976315E 02</td>
<td>0.62566254E 06</td>
</tr>
<tr>
<td>0.76897070E 14</td>
<td>0.89017445E 03</td>
<td>0.24422610E 01</td>
<td>0.12276187E 03</td>
<td>0.84992719E 07</td>
</tr>
<tr>
<td>0.12511050E 14</td>
<td>0.56148821E 05</td>
<td>0.31529360E 01</td>
<td>0.14059955E 02</td>
<td>0.57084620E 06</td>
</tr>
<tr>
<td>0.73720783E 00</td>
<td>0.14852208E 04</td>
<td>0.46265479E 00</td>
<td>0.30216807E 00</td>
<td>0.12156014E 05</td>
</tr>
<tr>
<td>0.15849701E 13</td>
<td>0.81355666E 04</td>
<td>0.41266769E 00</td>
<td>0.12638663E 00</td>
<td>0.10948410E 05</td>
</tr>
<tr>
<td>0.14454395E 13</td>
<td>0.23403788E 04</td>
<td>0.29511519E 00</td>
<td>0.15523567E 00</td>
<td>0.61914110E 06</td>
</tr>
<tr>
<td>0.29782732E 14</td>
<td>0.12158127E 05</td>
<td>0.44124926E 00</td>
<td>0.31922846E 00</td>
<td>0.12978240E 05</td>
</tr>
</tbody>
</table>
| 0.18742986E 13 | 0.48944043E 05 | 0.23900000E 05 | 0.52097595E 05 | 0.59556579E 05 | 0.57797939E 05 | 0.93125190E 05 | 0.80074980E 05 | 0.26415799E 05 | 0.94050998E 05 | 0.59470925E 05 | 0.25 | 25
In CHEMIE, the INPUT/OUTPUT STATEMENTS are

```plaintext
READ(5,600) (CW(J), J=1, ISP)
WRITE(6,700) (CW(J), J=1, ISP)
READ(5,601) (AU(K), K=1, IRE)
WRITE(6,701) (AU(K), K=1, IRE)
READ(5,601) (CU(K), K=1, IRE)
WRITE(6,701) (CU(K), K=1, IRE)
READ(5,602) (E(K), K=1, IRE)
WRITE(6,702) (E(K), K=1, IRE)
READ(5,602) (D(K), K=1, IRE)
WRITE(6,702) (D(K), K=1, IRE)
WRITE(6,703)
READ(5,605) ((GNUF(J,K), J=1, ISP), K=1, IRE)
WRITE(6,705) GNUP
READ(5,605) ((GNUB(J,K), J=1, ISP), K=1, IRE)
WRITE(6,705) GNB
605 FORMAT(7F10.0)
705 FORMAT(2X,7F12.5)
DO 5 K=1, IRE
  5 READ(5,606) AF(K), BF(K), AB(K), BB(K)
606 FORMAT(4F10.0)
```

Data are

```
1.000  16.  18.016  17.00e  32.  2.016  28.016  CW
14.008  30.008  44.016  46.008  28.011  44.011  142.286
.26000E 13.20000E 09
.55000E 14.14000E 13.32000E 15.83000E 14.70000E 18
.15000E 17.61000E 15.10000E 15.90000E 15.18200E 14
.20000E 17.47000E 15.13300E 11.70000E 14.58000E 11
.14200E 15.10000E 10
8504. 4026. 3019. 503. 55353. 62196. 113171. 59378. 75841.
30696. 37237. 3875. 19675. 0. 23651. 32105. 69145.
503. 3019. 10618. 9108. 0. 0. 0. 0. 0.
10769. 0. 13712. 3563. 37992. 0. 14090. 0.
```
The GNUF are

1.

1. 1.

1.

2.

1. 1.

1. 1.

1. 1.

1. 1.

1. 1.

1. 1.

2. 5.

1.
The GNUB are

1.  1.

2.  1.

2.  2.

1.  1.  /

1.  1.  1.

1.  1.  1.

1.  1.  1.

1.  1.  1.

10.  11.
The AF, BF, AB, BB are:

<table>
<thead>
<tr>
<th></th>
<th>1.0</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>-1.</td>
<td>2.</td>
</tr>
<tr>
<td>1.0</td>
<td>1.</td>
<td>2.</td>
</tr>
<tr>
<td>-0.5</td>
<td>1.</td>
<td>2.</td>
</tr>
<tr>
<td>-1.</td>
<td>1.</td>
<td>2.</td>
</tr>
<tr>
<td>-1.5</td>
<td>1.</td>
<td>2.</td>
</tr>
<tr>
<td>-1.</td>
<td>1.</td>
<td>2.</td>
</tr>
<tr>
<td>1.</td>
<td>1.</td>
<td>1.</td>
</tr>
<tr>
<td>1.</td>
<td>1.</td>
<td>1.</td>
</tr>
<tr>
<td>0.5</td>
<td>1.</td>
<td>1.</td>
</tr>
<tr>
<td>1.</td>
<td>1.</td>
<td>1.</td>
</tr>
</tbody>
</table>

This card all zeros

EN is

10.
C MAIN PROGRAM RRR00034

C READ IN INITIAL PROFILES (VELOCITY, TEMPERATURE AND CONCENTRATIONS)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION D(14),DES(14)
COMMON X(51),S(51),U(51),V(14,51),R0U(51),T(51),P5I(51)
COMMON UPR(51),YPR(14,51),HST(51),HT(51),YPR(51),HTPR(51)
COMMON CM(51),X(14,51),AP(51),APR(51),UEO,UEI,RHUE,RHUD,YE,YOD
COMMON HEI,HEO,CEI,CHO,OPS,NNN,NEL,NEU
COMMON /TTT/VOEI,VOEO
COMMON /TOK/H(14)
COMMON /SA/SX,ZX,UM
ISKA,BUK,PIK,IKS,IK,IA,KO,KOR
COMMON /SU/DLX
COMMON /SIG/PA(14,14),PT(14),W(14,51)
COMMON /SYLT/ E(14,51)
COMMON /VVV/IZE
COMMON /CEN/KRISE
COMMON /SA/AREX
COMMON /SSS/IAP
COMMON /VVS/YO(51)
COMMON/ZZ/ISP,IRE,IMU
COMMON /VP/YEI(14),YE0(14)
COMMON /RRR/TO(51)
COMMON /XXX/TAU
COMMON /UUU/ZZ(51)
KQ +VE INDICATES REGIME 2 HAS BEEN REACHED
DATA NLS/51,KRIS,JA,TT/10,10,10,17,KRYS,KRAS/-10,20/
DATA /7/0.0,KZ,509,<<N,507,KR1/10,KTILE/10,KUN/0/
DATA KTELL/3,UE/1.000/
ISP=14
IRE=17
IMU=11
TAU=40-3
IZE=0
KAN=10
KOM=0
KRYS=50
KRIS=-10
KQ=10
ZK=0.0
CALL THERMO(KRIS,JA,IT)
I,J=1
KRAS=20
KRY5=-II
CALL CHEMIF (KRY5,KRAS,J)
WRITE (6,390)
CALL INIT
NIA=NXI
DELY=.00001
RUP=DELY/(DPS1**2)
ZMAX=.7
AREX=DELY
KK1=1
KK2=NXI
NNI=10
CALL SPECIE (KK1,KK2,NI)
J=10
KRIS=100
DO 20 I=1,NNI
IFRI(I)=II(I)
IT=1
CALL THERMO (KRIS,JA,IT)
HST=0.0
DO 10 J=1,ISCP
YPR(I,J)=Y(J,1)
10 HST=HST+X(J,1)*Y(J,1)
HST(I)=HST+ROD(R,GM(I))
HT(I)=(HT(I)**2)/(2.*(E32.17778.16)+HST(I))
TPR(I)=5(I)
20 HTPR(I)=HT(I)
HFL=HT(1)
HFN=HT(NXI)
NLow=1
NHIG=NNI-1
CALL TRANSFO (NLow,NHI)
IN=0
CALL INTEG (NLow,NHIG,IN)
WRITE (6,370) NNI
WRITE (6,380) (I,RO(I)+PST(I),U(I),T(I),S(I),CM(I),V(6,I),I=1,NNI)
WRITE (6,380) (I,RO(I)+PST(I),HT(I),HST(I),HTPR(I),X(6,I),X(3,I),I=1,NNI)
30 CONTINUE
DUPX=AREX
DELX=AREX
40   Z=Z+DELX
    IF (Z.GT.ZMAX) GO TO 310
    ZX=Z*12.
    KUN=KUN+1
    KUSK=0
    NEU=NEU
    NEV=NEL
    IA=NNN-1
    NEU=IA-4
    NEV=NEU
    IK=NEL-4
    IF (IK.LE.1) IK=1
    IKS=(IA-IK)+1
    IKU=IKS-1
    RUP=DELX/(DPS1**2)
    IF (IA.GE.(NILS-1)) GO TO 50
    GO TO 60
50   CONTINUE
    KUN=0
    KUM=0
    CALL CHANGE(RUP)
    Z=ZX/12.
    IK=1
    IA=NNN-1
    NEU=IA-4
    IF (IK.LE.1) IK=1
    IKS=(IA-IK)+1
70   N1NA=NNN
    WRITE (6,370) NNN
    WRITE (6,380) (1,*,(1),PSI(I),U(I),Y(I),S(I),CM(I),Y(6,1),I=1,NNN)
    KUSK=KUSK+1
    UED=U(NNN)
    CALL SETUP (RUP)
    CALL GAUSS
    KRY=-11
    KRAS=5
    IJ=NNN
    CALL CHEMIE(KRY,CRAS,IJ)
    DO 400 J=1,ISP
400   E(J,NNN)=W(J,NNN)/U(NNN)
    DO 401 J=1,ISP
401   Y(J,NNN)=DELX*E(J,NNN)+YPR(J,NNN)
    KK1=NNN
KK2=NNN
NU=10
CALL SPECIE(KK1,KK2,NU)
DO 402 J=1,1SP

402 VEN(J)=Y(J,NNN)
CMO=CM(NNN)
JA=10
KRI=10
IT=NNN
HST(NNN)=HT(NNN)-(U(NNN)**2)/(2.*32.17*778.75)
CALL THERMO(KRI,JA,IT)
TEO=S(NNN)
NU=0
CALL SPECIE(KK1,KK2,NU)
RHJO=RHO(NNN)
DO 110 I=IK,IA
IF (KQ,GT,0) GO TO 70
IF (I,LT,IK) GO TO 90
70 IF (I,GT,IA) GO TO 90

I=1
CALL CHEMIE(KRYS, KRAS, IJ)
DO 80 J=1,1SP
80 ET(J,I)=S*(U(I)+UPR(I))*W(J,I)/(U(I)+UPR(I))
GO TO 110
B 99
B 100
110 DO 100 J=1,1SP
100 ET(J,I)=O,0
100 W(J,I)=O,0
110 CONTINUE
CALL SETYP(RUP)
CALL GAUSS
IF (KRISE,GT,0) GO TO 320
CALL SETHP(RUP)
CALL GAUSS
KKI=IK
KK2=IA
JA=10
KRI=10
DO 120 I=IK,IA
IT=1
CALL THERMO(KRI,JA,IT)
120 CONTINUE
NU=0
CALL SPECIE(KK1,KK2,NU)
B 103
B 104
B 105
B 106
B 107
B 108
B 109
B 110
B 111
B 112
B 113
B 114
B 115
B 116
B 117
B 118
B 119
B 120
B 121
B 122
B 123
B 124
B 125
B 126
<table>
<thead>
<tr>
<th>Line</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>130</td>
<td>NLOW=1K</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NNHIGH=1A</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IN=+10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL TRANSQ(NLOW,NNHIGH)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL INE (NLOW,NNHIGH,IN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IF(KUN=EQ,50) KAN=10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IF(KAN=LT,0) GO TO 623</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL PRINT (KUSK,KUN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KAN=10</td>
<td></td>
</tr>
<tr>
<td>623</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UEO=U(NNN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL SETUP (RIJ)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL GAUSS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KRA=0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KUSK-KUSK+1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DAD=(S(NNN)-T(QR(NNN)))/1.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>T(NNN)=5*(T(NNN)+TPR(NNN))/1.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DO 403 J=1,ISP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DID(J)=Y(J,NNN)-YPR(J,NNN)</td>
<td></td>
</tr>
<tr>
<td>403</td>
<td>Y(J,NNN)=5*(Y(J,NNN)+YPR(J,NNN))</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TJ=NNN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL CHEMIE(KRYS,KRA,TJ)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DO 404 KI=1,ISP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUM=0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DO 405 J=1,ISP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DES(J)=DID(J)*PA(J,KI)</td>
<td></td>
</tr>
<tr>
<td>405</td>
<td>SUM=SUM+DES(J)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RA=W(KI,NNN)/U(NNN)*2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BE=(DAD*PT(KI)+SUM)/II(NNN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>E(KI,NNN)=5*(RA+BE)</td>
<td></td>
</tr>
<tr>
<td>404</td>
<td>CONTINUE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DO 406 J=1,ISP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Y(J,NNN)=DELX*E(J,NNN)+YPR(J,NNN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KKI=NNN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KK2=NNN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NU=10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CALL SPECIE(KKI, KK2, NU)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DO 407 J=1,ISP</td>
<td></td>
</tr>
<tr>
<td>407</td>
<td>YEDT(J)=Y(J,NNN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CM=CH(NNN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>JA=10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>KRS=10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>JT=NNN</td>
<td></td>
</tr>
</tbody>
</table>
HST(NNN)=H(T(NNN)-(U(NNN)**2)/(2.*32.177876)
CALL THERMO(KRIS,JA,IT)
TEO=S(NNN)
NU=0
CALL SPECIE(K12,NU)
RMUO=ROU(NNN)
DO 220 I=1,I
IF (K0,GT,0) GO TO 140
IF (I,LT,1) GO TO 200
IF(I,GT,IA) GO TO 200
140   DAD=(S(I)-TPR(I))/1.8
      T(I)=0.6*(T(I)+TPR(I)/1.8)
DO 150 J=1,ISP
510   Y(J,I)=Y(J,I)-YPR(J,I)
150   Y(J,I)=0.6*(Y(J,I)+YPR(J,I))
TI=1
CALL CHEMIE(KRYS,KRAS,T)
DO 190 KT=1,ISP
      SUM=0.0
DO 170 J=1,ISP
      IF (DAR5(PA(J,KT),GT,0,1D=30) GO TO 160
      DES(J)=0.0
      GO TO 170
160   DES(J)=D10(J)*PA(J,KT)
170   SUM=SUM+DES(J)
      BA=(U(I)*UPR(I))*W(KI,I)/(U(I)*UPR(I))
510   BE=(DAD*PT(KI)+SUM)/U(I)
180   E(KI,I)=0.6*(BA+BE)
190   CONTINUE
      GO TO 220
200   DO 210 J=1,ISP
210   E(J,I)=0.0
220   CONTINUE
CALL SETYP(RUP)
CALL GAUSS
IF (KRIS,GT,0) GO TO 320
CALL SETHP(RUP)
CALL GAUSS
DO 230 I=1,TA
  IT=1
CALL THERMO(KRIS,JA,IT)
230   CONTINUE
NU=0
XX1=IK
XX2=IA
CALL SPECIE (XX1,XX2,NU)
IF (KUSK.EQ.3) GO TO 240
GO TO 130
240 CONTINUE
IN=0
IF(IAP.LT.IA) GO TO 280
NEU=IA-3
IA=NNN
NNN=NNN+1
IF(NNN.GT.NILS) GO TO 50
PSI(NNN)=PSI(NNN-1)+DPSI
U(NNN)=U(NNN-1)
UPR(NNN)=UPR(NNN-1)
S(NNN)=S(NNN-1)
T(NNN)=S(NNN)/1.6
YPR(NNN)=YPR(NNN-1)
DO 408 J=1,ISP
Y(J,NNN)=Y(J,NNN-1)
X(J,NNN)=X(J,NNN-1)
408 YPR(J,NNN)=YPR(J,NNN-1)
VO(NNN)=VO(NNN-1)
VO(NNN)=TO(NNN-1)
HY(NNN)=HY(NNN-1)
HSY(NNN)=HSY(NNN-1)
ZY(NNN)=ZY(NNN-1)
ROU(NNN)=ROU(NNN-1)
NLOW=IK
NHIGH=NNN-1
CALL TRANSP(NLOW,NHIGH)
CALL INTEG(NLOW,NHIGH,IN)
270 CONTINUE
280 CONTINUE
IK=NEL-4
IF (IK.LE.1) IK=1
IKS=IK-1
NLOW=IK
NHIGH=IK
CALL TRANSP(NLOW,NHIGH)
CALL INTEG (NLOW,NHIGH,IN)
DO 300 I=IK,NNN
UPR(I)=U(I)
300 CONTINUE
<table>
<thead>
<tr>
<th>Step</th>
<th>Instruction</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>STOP</td>
<td>F 0001</td>
</tr>
<tr>
<td>2</td>
<td>CALL PRNT</td>
<td>F 0002</td>
</tr>
<tr>
<td>3</td>
<td>G0 TO 624</td>
<td>F 0003</td>
</tr>
<tr>
<td>4</td>
<td>IF KUM=0</td>
<td>F 0004</td>
</tr>
<tr>
<td>5</td>
<td>IF KUM=1</td>
<td>F 0005</td>
</tr>
<tr>
<td>6</td>
<td>IF KUM=2</td>
<td>F 0006</td>
</tr>
<tr>
<td>7</td>
<td>IF KUM=3</td>
<td>F 0007</td>
</tr>
<tr>
<td>8</td>
<td>IF KUM=4</td>
<td>F 0008</td>
</tr>
<tr>
<td>9</td>
<td>IF KUM=5</td>
<td>F 0009</td>
</tr>
<tr>
<td>10</td>
<td>IF KUM=6</td>
<td>F 000A</td>
</tr>
<tr>
<td>11</td>
<td>IF KUM=7</td>
<td>F 000B</td>
</tr>
<tr>
<td>12</td>
<td>IF KUM=8</td>
<td>F 000C</td>
</tr>
<tr>
<td>13</td>
<td>IF KUM=9</td>
<td>F 000D</td>
</tr>
<tr>
<td>14</td>
<td>IF KUM=10</td>
<td>F 000E</td>
</tr>
<tr>
<td>15</td>
<td>IF KUM=11</td>
<td>F 000F</td>
</tr>
<tr>
<td>16</td>
<td>IF KUM=12</td>
<td>F 0010</td>
</tr>
<tr>
<td>17</td>
<td>IF KUM=13</td>
<td>F 0011</td>
</tr>
<tr>
<td>18</td>
<td>IF KUM=14</td>
<td>F 0012</td>
</tr>
<tr>
<td>19</td>
<td>IF KUM=15</td>
<td>F 0013</td>
</tr>
<tr>
<td>20</td>
<td>IF KUM=16</td>
<td>F 0014</td>
</tr>
<tr>
<td>21</td>
<td>IF KUM=17</td>
<td>F 0015</td>
</tr>
<tr>
<td>22</td>
<td>IF KUM=18</td>
<td>F 0016</td>
</tr>
<tr>
<td>23</td>
<td>IF KUM=19</td>
<td>F 0017</td>
</tr>
<tr>
<td>24</td>
<td>IF KUM=20</td>
<td>F 0018</td>
</tr>
<tr>
<td>25</td>
<td>IF KUM=21</td>
<td>F 0019</td>
</tr>
<tr>
<td>26</td>
<td>IF KUM=22</td>
<td>F 001A</td>
</tr>
<tr>
<td>27</td>
<td>IF KUM=23</td>
<td>F 001B</td>
</tr>
<tr>
<td>28</td>
<td>IF KUM=24</td>
<td>F 001C</td>
</tr>
<tr>
<td>29</td>
<td>IF KUM=25</td>
<td>F 001D</td>
</tr>
<tr>
<td>30</td>
<td>IF KUM=26</td>
<td>F 001E</td>
</tr>
<tr>
<td>31</td>
<td>IF KUM=27</td>
<td>F 001F</td>
</tr>
<tr>
<td>32</td>
<td>IF KUM=28</td>
<td>F 0020</td>
</tr>
<tr>
<td>33</td>
<td>IF KUM=29</td>
<td>F 0021</td>
</tr>
<tr>
<td>34</td>
<td>IF KUM=30</td>
<td>F 0022</td>
</tr>
<tr>
<td>35</td>
<td>IF KUM=31</td>
<td>F 0023</td>
</tr>
<tr>
<td>36</td>
<td>IF KUM=32</td>
<td>F 0024</td>
</tr>
<tr>
<td>37</td>
<td>IF KUM=33</td>
<td>F 0025</td>
</tr>
<tr>
<td>38</td>
<td>IF KUM=34</td>
<td>F 0026</td>
</tr>
<tr>
<td>39</td>
<td>IF KUM=35</td>
<td>F 0027</td>
</tr>
</tbody>
</table>

Legend:
- **KUM**: Current Kum
- **IF**: If condition
- **STOP**: Stop instruction
- **CALL PRNT**: Call print instruction
- **GO TO**: Go to instruction
- **CONTINUE**: Continue instruction
- **DELX**: Delete instruction
- **AREX**: Arex instruction
- **KUM=0**: Current Kum equals zero
- **KUM=1**: Current Kum equals one
- **KUM=2**: Current Kum equals two
- **KUM=3**: Current Kum equals three
- **KUM=4**: Current Kum equals four
- **KUM=5**: Current Kum equals five
- **KUM=6**: Current Kum equals six
- **KUM=7**: Current Kum equals seven
- **KUM=8**: Current Kum equals eight
- **KUM=9**: Current Kum equals nine
- **KUM=10**: Current Kum equals ten
- **KUM=11**: Current Kum equals eleven
- **KUM=12**: Current Kum equals twelve
- **KUM=13**: Current Kum equals thirteen
- **KUM=14**: Current Kum equals fourteen
- **KUM=15**: Current Kum equals fifteen
- **KUM=16**: Current Kum equals sixteen
- **KUM=17**: Current Kum equals seventeen
- **KUM=18**: Current Kum equals eighteen
- **KUM=19**: Current Kum equals nineteen
- **KUM=20**: Current Kum equals twenty
- **KUM=21**: Current Kum equals twenty-one
- **KUM=22**: Current Kum equals twenty-two
- **KUM=23**: Current Kum equals twenty-three
- **KUM=24**: Current Kum equals twenty-four
- **KUM=25**: Current Kum equals twenty-five
- **KUM=26**: Current Kum equals twenty-six
- **KUM=27**: Current Kum equals twenty-seven
- **KUM=28**: Current Kum equals twenty-eight
- **KUM=29**: Current Kum equals twenty-nine
- **KUM=30**: Current Kum equals thirty
360 FORMAT (/,'2X,13HSTEP DECREASE,3(2X,I20,10).
380 FORMAT (2X,14,TD1K,7)                   B 228
370 FORMAT (14,2X,4HNNN,=,14,/)            B 230
390 FORMAT (*)                                 B 232-
END                                              B 233
SUBROUTINE SPECIE (KK1,KK2,NU)                C 1
IMPLICIT REAL*8(A-H,Z)-                    
DIMENSION G(14),CW(14)                       C 6
COMMON R(51),S(51),U(51),Y(14,51),ROU(51),T(51),PSI(51)  C 6
COMMON UPR(51),YPR(14,51),HST(51),HT(51),TPR(51),HTPR(51)
COMMON CM(51),X(14,51),API(51),APR(51),UED,UEI,RHUE,RHUI,IEI,IEO  C 6
COMMON HEI,HEO,CH1,CM1,OPSI,N11,NEL,NEH
COMMON/W227/ISP,IRE,IMU
DATA P/1.000/
DATA CW/1.008,16.0,18.016,17.008,32.0,2.016,28.016,14.008,30.008,  
46.016,46.008,28.011,44.011,142.286/
DO 40 I=KK1,KK2                                C 9
IF (NU.EQ.0) GO TO 30                           C 10
CM(I)=0.0                                      C 11
GM=0.0                                         C 12
DO 10 J=1,ISP                                  C 14
G(J)=Y(J,1)/CM(1)                               C 15
10   GM=GM+G(J)                                  C 17
DO 20 J=1,ISP                                  C 18
X(J,1)=G(J)/GM                                  C 19
20   CM(I)=CM(I)+X(J,1)*G(J)                     C 20
IF (NU.EQ.10) GO TO 40                         C 21
30   CONTINUE                                    C 22
RM=1545.9/CM(1)                                C 23
RO*P2116.27/RM*T(I)*1.9                        C 24
ROUT(I)=RO*U(I)                                 C 25
40   CONTINUE                                    C 26-
RETURN                                          C 27
END                                              C 28-
SUBROUTINE THERMO (KATS,JA,II)                 C 29
C THERMODYNAMIC SUBROUTINE, BASED ON NASA SP-3001  C 30
C REFERENCE TEMPERATURE 298.15 DEG. K            C 31
C SPECPSIM,H20,OH2O,H2,HO2,H2O,H2,N2            C 32
C FRYS=-VE: IMPLIES FROZEN FLOW, HENCE SENS(I) NOT ZERO C 33
C JA +VE: I READ IN CONSTANTS                   C 34
C JA -VE I CALCULATE ENTHALPIES                C 35
IMPLICIT REAL*8(A-H,Z)-                    
DIMENSION A(6,14),B(6,14),SENS(14)
COMMON R(S1), S(S1), U(S1), V(S1), AOU(S1), T(S1), PSI(S1)
COMMON UPR(S1), YPR(S1), HST(S1), HT(S1), TPR(S1), HTPR(S1)
COMMON CM(S1), X(S1), AP(S1), APE(S1), UEO, UET, RHUE, RHUD, TEI, TEU
COMMON HEI, HE0, CAEI, CAO, DSPS1, NN, NEL, NEU
COMM/227/ISP, IRE, IMU
COMMON /INSK/ H(14)
M=IT
10 CONTINUE
   IF (JA) 10, 10, 180
   CONTINUE
   IF (KRIS, EQ, 100) GO TO 100
   NR=0
   TP=S(M)
   T(M)=TP/1.8
   TOETP
50 CONTINUE
   HAS=0.0
   GO TO 100
   CONTINUE
   DO 40 J=1, ISP
50   HAS=HAS+X(J, M)*H(J)
   HTS=HAS*1000./CH(M)
   NR=NR+1
   IF (NR, GE, 20) GO TO 90
   CHH=HST(M) = HTS
   CHK=OAS(CHKH/HST(M))
   IF (CHK, LE, 0.001) GO TO 90
  IF (NR, GE, 2) GO TO 80
   TPETP
   FA=CHH
   IF (CHK) 90, 90, 50
   TP=TP+1.
   GO TO 70
50   TP=TP+1.
   TM=TP/1.8
   GO TO 20
   TA=TP
   FA=CHH
   IF ((FB=FA), EQ, 0.0) GO TO 200
   TP=TA-FA*(15-TH)/(FB-FA1)
   TM=TP/1.8
   FA=FB
   TA=TA
   GO TO 20
90 CONTINUE
S(M)=T(M)*1.8
GO TO 170
100 CONTINUE
C=1.98726
DO 160 J=1,ISP
IF (T(M)=1000.) 110,110,120
1=1T(M)*T(M)+A(6,J)/T(M)
GO TO 130
120 HA=B(1,J)+[B(2,J)/2.+B(3,J)/3.+B(4,J)/4.+B(5,J)/5.]*T(M)*T(M) D 65
1=1T(M)*T(M)+B(6,J)/T(M)
130 IF (K(K)) .EQ. +100) GO TO 170
GO TO 30
140 CONTINUE
H(J)=HA*G*T(M)/1000.
GO TO 160
150 H(J)=HA*G*T(M)/1000.
GO TO 160
160 CONTINUE
IF (K(K) .EQ. +100) GO TO 170
GO TO 30
170 CONTINUE
RETURN
180 READ (5,260) ((A(K,J),K=1,6),J=1,ISP)
DO 190 J=1,ISP
READ (5,260) (B(K,J),K=1,6)
WRITE (6,280) (B(K,J),K=1,6)
190 CONTINUE
READ (5,270) (SENS(J),J=1,ISP)
GO TO 170
200 WRITE (6,210) M,R(M),PS(M),HST(M),HTS,FA,FR,TP,T(M)
GO TO 90
C
210 FORMAT (///,2X,I4,8(I2X,E13.5),1X,4HMFHR,///)
220 FORMAT (///,2X,I5,8(12X,E15.8))
230 FORMAT (2X,I5,8(I2X,E15.8))
240 FORMAT (2X,I5,8(I2X,E15.8))
250 FORMAT (2X,I5,8(I2X,E15.8))
260 FORMAT (4D15.8/2015.H)
270 FORMAT (7D10.0)
280 FORMAT (///,1X,6E20.8,///)
END
SUBRITIE INIT
IMPLICIT REAL*M*A=10.)-2)
DIMENSION UE(51),SE(51),YE(14,51),PS(51)
COMMON RT(51),ST(51),UT(51),YT(14,51),KNU(51),TT(51),PS(151)
COMMON UPR(51),UPR(51),HST(51),HT(51),TRP(51),RTPR(51)
COMMON CM(51),X(14,51),APR(51),APR(51),UFU,IUE1,RUDE,RHU,TE1,TEU
COMMON HE1,HEU,CMF1,CMF1,CMF1,CMF1,NNN,NNN,NNN,NNN
COMMON /TTT/ YDEI,YDEU
COMMON /Z2Z/ YDEI,YDEU
COMMON /RRR/ YDEI,YDEU
COMMON /YPP/ YE1,YE1,YE1
COMMON /Z00/ YE1,YE1,YE1
COMMON /VYY/ YE1,YE1
READ(5,100) NNN
READ(5,101)(UE(L),SE(L),R(L),L=1,NNN)
READ(5,102)(YE1,L,J=1,ISP),L=1,NNN)
DO 40 L=1,NNN
SIL=SE(L)
IF(L.GT.1)T(L)=SIL/1.08
U(L)=UE1(L)
40 CONTINUE
DO 50 J=1,ISP
30 Y(E1,J)=YE1(J,J)
40 CONTINUE
40 KK1=1
KK2=NNN
NU=10
CALL SPECIE (K1,KK2,NU)
PS1=0.
FRM=0.
PS1=0.
DO 50 L=1,NNN
IF(L.GT.1)GO TO 50
DONE(1,L)=RI-11/12,
FRI=RI(L)*RI(L)/120.
DPS=DEC(RF1*FRM)
FRM=FRI
PS1=PS1+DPS
END
PS1=DSORT(DS1)
30 CONTINUE
UE1=11
UEU=U(51)
TE1=51
TEU=5(S)
CMF1=CM(11)
CMF1=CM(51,51)
DO 60 J=1,ISP
  YET(IJJ)=Y(IJJ)
60 YET(IJJ)=Y(IJJ,NNN)
DPST=PS(NNN)/(NNN-1)
RMUE=ROU(I)
RMUO=ROU(NNN)
PSI(I)=0.
DO 2 J=2,NNN
2 PSI(IJJ)=PSI(J-1)+DPST
NMI=NNN-1
DO 3 J=1,NMI
DO 4 K=1,NNN
DIF=PSI(JJJ)-PS(K)
IF(DIF) .LT.0.
6 UT(IJJ)=UE(K)
ST(JJJ)=SE(K)
T(IJJ)=T(IJJ)/L
DO 8 L=I,ISP
8 Y(LJJ)=YE(L,K)
GO TO 3
5 RAT=PS(TJJJ)-PS(K-1)/PS(K)-PS(K-1)
UT(IJJ)=UE(K-1)+RAT*(UE(K)-UE(K-1))
ST(JJJ)=SE(K-1)+RAT*(SE(K)-SE(K-1))
T(IJJ)=T(IJJ)/L
DO 9 L=I,ISP
9 Y(LJJ)=YE(L,K-1)+RAT*(YE(L,K)-YE(L,K-1))
GO TO 3
4 CONTINUE
3 CONTINUE
PSI(NNN)=PS(NNN)
NEL=I
NEU=22
DO 11 J=1,NNN
11 YOT(IJJ)=Y(IPJJ)
DO 12 J=NEL,NNN
12 Z(IJJ)=0.
DO 14 J=1,NNN
14 YUE(IJJ)=0.
YOE=YE(IPJJ)
YOE=YE(IPJJ,NNN)
RETURN
100 FORMAT(15)
101 FORMAT(3F10.0)
102 FORMAT(7D10.0)
END
SUBROUTINE TRANS(NLOW,NHIGH)
[IMPLICIT REAL*8(A-H,O-Z)]
COMMON R(51),S(51),U(51),V(51),T(51),PSI(51)
COMMON UPR(51),VPR(51),AUPR(51),APT(51),H(51),TPR(51),HTPR(51)
COMMON CH(51),X(14,51),AP(51),APT(51),UDEQ,UE1,RHUE,HDUE,TE1,TE0
COMMON HE,HE0,CMET,CMO,DPST,NNN,NEL,NEU
COMMON /SAX/ZX,UM
DATA RJ/7.2/
GOEFF=0.036/RJ
L=NLOW
LL=NHIGH+1
SUM=0.
DO I=K,L LL
ARG=DABS(CROU(I,L)=ROU(I)/ROU(I)*PSI(I))
SUM=SUM+ARG
SUM=SUM+DPST
UM=SUM*GOEFF
RETURN
END
SUBROUTINE INTEG (NLOW,NHIGH,IN)
[IMPLICIT REAL*8(A-H,O-Z)]
COMMON R(51),S(51),U(51),V(51),T(51),PSI(51)
COMMON UPR(51),VPR(51),AUPR(51),APT(51),H(51),TPR(51),HTPR(51)
COMMON CH(51),X(14,51),AP(51),APT(51),UDEQ,UE1,RHUE,HDUE,TE1,TE0
COMMON HE,HE0,CMET,CMO,DPST,NNN,NEL,NEU
COMMON /SAX/ZX,UM
C INTEGRATION ROUTINE BACK TO PHYSICAL PLANE
C IF IN GT THAN 0.0 A IS AVERAGED
C IF IN EQ TO 0.0 A IS BASED ON PAST TIME ROW.
INIT=1
L=NLOW
LL=NHIGH+1
DO 10 K=LL
IF (INIT==30,30,30)
DPST=PSI(K)
IF (TMGEQ,0.01) GO TO 20
IK=K
FI=PSI(K)/ROU(K)
B=FI+BN
R(K)=1DSRT(1)+2.
RR=(4*(K)+12)+2
10 CONTINUE
G 1
G 2
G 3
G 4
G 5
G 6
G 7
G 8
G 9
G 10
G 11
G 12
G 13
G 14
G 15
G 16
G 17
G 18
G 19
G 20
G 21
SP=UM*R(U)K*RR/PSI(K)
F0=F1
INIT=10
GO TO 40

20 R(K)=0.0
B=0.0
F0=0.0
SP=0.0
INIT=10
GO TO 40

30 IK=K
F1=PSI(K)/R(U)K
BR=(F0+F1)*PSI
K=B+H
R(K)=DISORT(B)+12
RR=R(K)/12.**2
IF (PSI(K)=E5,0.0) GO TO 70
SP=UM*R(M)K*RR/PSI(K)
F0=F1

40 IF (IN=EQ,0) GO TO 50
APR(K)=5.(SP+APR(K))
GO TO 60

50 APR(K)=SP
APR(K)=SP

60 CONTINUE
RETURN

70 WRITE (6,RO) K,LL,LL,2X,0(U)K)
STOP

80 FORMAT (1113I15, STOP,3X,312X,I4,23X,F15.7)
END

SUBROUTINE SETUP (KUP)
IMPLICIT REAL*(A-H,O-Z)
COMMON R(51),S(51),V(51),V14,51),KOU(51),T(51),PSI(51)
COMMON UPR(51),UPR(14,51),NST(51),HT(51),TPR(51),HTPR(51)
COMMON CM(31),C14,51),APR(51),APR(14,51),U,E17,HE1,HE2,HE3,HE4,HE5,HE6
COMMON HE1,HE2,HE3,HE4,HE5,HE6
COMMON TRPSET/ AT14,51,CT14,51,MT14,51,ALPT(51),94K,A
COMMON 754X7 2X,OM
IF (KUP=1)
ON 50 K=1,IKS

G 23
G 24
G 25
G 26
G 27
G 28
G 29
G 30
G 31
G 32
G 33
G 34
G 35
G 36
G 37
G 38
G 39
G 40
G 41
G 42
G 43
G 44
G 45
G 46
G 47
G 48
G 49
G 50
G 51
G 52
H 1
H 2
H 3
H 4
H 5
H 6
H 7
H 8
H 9
H 10
H 11
[=1+1]
IF (K.EQ.1) GO TO 10
IF (K.EQ.1)GO TO 20
A(I)=A*(AP(I)-.25*AP(I+1)+.25*AP(I-1))
B(I)=2.*(PS(I)*AP(I)+Q)
C(I)=.25*AP(I+1)-.25*AP(I-1)
BAK=2.*(PS(I)-AP(I)+Q)
D(I)=A(I)*UPR(I-1)+BAK*UPR(I)+C(I)*UPR(I+1)
ALP(I)=D(I)-A(I)*C(I-1)/ALP(I-1)
V(I)=D(I)-1+A(I)*V(I-1)+1/ALP(I-1)
GO TO 50
10 A(I)=0.0
IF (I.EQ.1) GO TO 20
B(I)=2.*(PS(I)*AP(I)+Q)
C(I)=2.*NUMQ
D(I)=D(I)-2.*NUMQ)*UPR(I)+C(I)*UPR(I+1)
30 ALP(I)=B(I)
V(I)=D(I+1)
GO TO 50
40 C(I)=0.0
A(I)=A*(AP(I)-.25*AP(I+1)+.25*AP(I-1))
B(I)=2.*(PS(I)*AP(I)+Q)
BUK=2.*(PS(I)-AP(I)+Q)
PISK=2.*(AP(I-1)+.25*AP(I+1)-.25*AP(I-1))
D(I)=A(I)*UPR(I-1)+BUK*UPR(I)+2.*PISK*UEO
ALP(I)=R(I)-A(I)*C(I-1)/ALP(I-1)
V(I)=D(I)+A(I)*V(I-1)+1/ALP(I-1)
50 CONTINUE
GO 60 I+1=K
60 CONTINUE
IF (I.EQ.1) GO TO 10
RETURN
C
END
SUBROUTINE SETUP (UPR)
IMPLICIT REAL*4(A,H,O=2)
COMMON R(S1),S1(S1),U1(S1),V1(S1),RID(S1),T(S1),PS1(S1)

COMMON UPR(51), YPR(14,51), HST(51), HT(51), TPR(51), HTPR(51)
COMMON CM(51), XI(14,51), AP(51), APR(51), UEO, UEI, RHUE, RHOU, TS1, TEL1
COMMON HET, MED, CMEI, CM0, D, PSI, XNN, NEL, NEF
COMMON /UPSET/ A(51), B(51), C(51), D(14,51), E(14,51), ALP(51), BAK, A
ISKA, RUK, PISK, IKS, IK, IA, KO, KOR
COMMON /SYLT/ F(14,51)
COMMON /ZZZ/ ISP, IRE, IMU
COMMON /SU/ DEX
COMMON /VP/ YET(14), YE0(14)
COMMON /SA/ ZK, JH
DATA CN/1, 0, 0/0, 0
Q=R1P
I=IK-1
DU 100 K=1, IKS
I=I+1
IF (K, EQ, 1) GO TO 20
IF (K, EQ, IKS) GO TO 80
A(I)=CN*A(I)
R(I)=2.*ST(I)+AP(I)*Q*CN
C(I)=CN*C(I)
BAK=2.*ST(I)*AP(I)*Q*CN
ALP(I)=R(I)-A(I)*C(I-1)/ALP(I-1)
DAX=2.*ST(I)*DAX
DU 10 J=1, ISP
D(I,J)=A(I)+VPR(J,1-1)+BAK*YPR(J,1)*CT(I)*YPR(J,1+1)+E(J,1)*DAX
V(J,1)=DT(J,1)+A(I)*V(J,1-1)/ALP(I-1)
GO TO 100
20 A(I)=Q*0
IF (IK, EQ, 1) GO TO 40
R(I)=2.*ST(I)*AP(I)*Q*CN
C(I)=CN*C(I)
ASKA=CN*ASKA
BAK=2.*ST(I)*AP(I)*Q*CN
DAX=2.*ST(I)*DAX
DU 30 J=1, ISP
D(T,J,1)=2.*ASKA*YF(I,1)+BAK*YPR(J,1)*CT(I)*YPR(J,1+1)+E(J,1)*DAX
GO TO 60
40 R(I)=1.+2.*UN*CN*0
C(I)=2.*CM*CN*0
DU 50 J=1, ISP
50 DT(J,1)=(1. + 2.*CM*CN*0)*YPR(J,1)+CT(I)*YPR(J,1+1)+E(J,1)*DAX
60 ALP(I)=R(I)
DO 70 J=1,ISP
  70 V(J,1)=U(J,1)
    GO TO 100
  80 C(I,1)=0.0
    A(I,1)=CN*I(1)
     R(I,1)=2.0*(PS(I,1)+C(I,1)+CN)
     HUK=2.0*(PS(I,1)-C(I,1)+CN)
     VSK=CN*I(PH)
     VL(T,1)=R(T,1)-A(T,1)*(T-1)/ALP*(T-1)
     DAX=2.0*PS(I,1)*DELX
     DO 90 J=1,ISP
      90 V(J,1)=AT(I,1)*VRP(J,1-1)*BUK*VRP(J,1)+ PISK*VEND(J,1)+E(J,1)*DAX
      I=PISK*VRP(J,1)
     CONTINUE
  100 CONTINUE
     DO 110 I=1,IA
     110 CONTINUE
     KOK=0
    RETURN
END
SUBROUTINE SETUP (RUP)
IMPLICIT REAL*(A-H,O-Z)
COMMON R(51,1),S(51,1),U(51,1),V(14,51),ROU(51,1),T(51,1),PS(51)
COMMON UPR(51,1),VRP(14,51),HT(51,1),HTPR(51,1),HIPR(51,1)
COMMON CMST(51,1),T(14,51),APST(51,1),APRP(51,1),UOE,UEM,RUE,CRMU,TE1,TE0
COMMON /UPSET/ AT(51,1),DT(51,1),CT(51,1),C(14,51),V(14,51),ALP(51,1),BAK,IA
COMMON /SAX/,ZA,IK,IA,KS,IA,ST,IA,ST,IA,ST,IA,ST
DATA CN*/IONO/
RUP=CN*
I=IK-1
DO 30 K=1,IKS
  30 I=I+1
IF (IK.EQ.1) GO TO 10
IF (IK.EQ.KS) GO TO 40
RAK=2.0*PS(I,1)*APR(I,1)+CN
DT(1,1)=AT(1,1)+HTPR(1,1)+RAK*HTPR(I-1)+C(I,1)+HTPR(I-1)
VL(1,1)=D(1,1)+AT(I,1)+VL(I,1-1)/ALP(I-1)
GO TO 20
10 IF (IK.EQ.1) GO TO 20
RAK=2.0*PS(I,1)*APR(I,1)+CN
ASKA=UP(1) + 0.25*AP(1) + 0.25*AP(1-I) + CN
D11(I) = 2.5 * ASKA + HET + HK + HTPR(1) + CT(I) + HTPR(1+1)
GO TO 30
20 D11(I) = D11(I-1) - 2.5 * HUP + CN + HTPR(1) + CT(I) + HTPR(I+1)
30 V1(I) = D11(I) + 1.1
GO TO 50
40 BUK=2.5 * (PS(1) - AP(1) + CN)
PI = 2.5 * (AP(1) + 0.25*AP(1+1) - 0.25*AP(I-1) + CN)
D11(I) = 1.1 * HTPR(I-1) + BUK * HTPR(1) + 2.5 * PI + HEO
V1(I) = D11(I) + 1.1 * V1(I-1) / ALP(I-1)
50 CONTINUE
60 CONTINUE
70 KIR=10
RETURN
END

SUBROUTINE GAUSS
IMPLICIT REAL (A-H, O-Z)
DIMENSION GI(I), CH(I)
DIMENSION NE(3), NOT(3)
COMMON R(51), S(51), U(51), Y(14,51), ROU(51), T(51), PSI(51)
COMMON UPR(51), VPR(14,51), HST(51), A(51), IPN(51), HTPR(51)
COMMON CM(51), XI(14,51), AP(51), APR(51), UED, UET, RHE, RHO U, TE, TED
COMMON HET, HEQ, CMET, CHG, OPI, NNN, NEI, NEU
COMMON /KSET/ A(51), B(51), C(51), M(14,51), V(14,51), ALP(51), HAK, A
ISKA, BUK, PI, KFS, TKS, KIA, K0, KOR
COMMON /SSS/ TAP
COMMON /IOO/ IZ(51)
COMMON /ZZZ/ ISP, IRE, IMU
COMMON /CEN/ KRISE
DATA TOL/1.00E-06/ 11
DATA CW/1.00E-16, 1.00E-16, 1.00E-17, 1.00E-18, 3.20E-2, 0.016, 0.016, 0.008, 3.0, 0.008,
1.45, 0.15, 0.45, 0.004, 0.28, 0.011, 0.44, 0.011, 1.42, 0.067
IF (NIT(3) .LT. TA) GO TO 1
WRITE (6, 3101) NIT(1), NIT(2), NIT(3)
1 NIT(3) = 0
11 NIT(3) = 10
KNS = 10
KOS = 10
IF (KUR) 10.70, 170
10 DO 50 M=1, IKS
TH = TH - 1
IF (M.EQ.1) GO TO 20
V(18) = (V(11) + C(18) * U(11 + 1)) / ALP(18)
GO TO 30

K 19

20 V(18) = (V(11) + C(18)) / ALP(18)

K 20

30 CONTINUE
IF (KOS.GT.0) GO TO 40
IF (DABS(U(18) - UE0).GT.TOL) GO TO 50

K 21

40 CONTINUE
IF (KAS.GT.0) GO TO 60
IF (DABS(U(18) - UE1).GT.TOL) GO TO 60

K 22

KAS = 10
NFI = IR
GO TO 60

K 23

50 NFI = IR
KOS = 10

K 24

60 CONTINUE
IF (KAS.LT.0) NFI = IR
GO TO 300

K 25

70 DO 160 M = 1, IK

K 26

160 IR = -1
IF (M.GT.1) GO TO 90

K 27

DO 80 J = 1, ISP
Y(J, IR) = (V(J, IR) + C(J) * Y(J, IR + 1)) / ALP(IR)

K 28

80 CONTINUE
GO TO 110

K 29

90 DO 100 J = 1, ISP

K 30

100 Y(J, IR) = V(J, IR) / ALP(IR)

K 31

110 CONTINUE
CMI(18) = 0.0
GM = 0.0

K 32

120 DO 120 J = 1, ISP

K 33

120 GT(J) = Y(J, IR) / GM

K 34

130 CM(I) = CM(I) + X(J, IR) * CW(J)

K 35

140 IF (M.EQ.1) GO TO 160
IF (KOS.GT.0) GO TO 140
IF (DABS(CM(18) - CM0).GT.TOL) GO TO 150

K 36

150 CONTINUE
IF (KAS.GT.0) GO TO 160
IF (DABS(CM(18) - CME1).GT.TOL) GO TO 160

K 37
\begin{verbatim}
NE(2)=18
KAS=+10
GO TO 140
150 ND(2)=18
KOS=+10
160 CONTINUE
IF (KAS.LT.0) NE(2)=1K
GO TO 300
170 DO 220 M=1,IKS
I8=19-1
IF (M.EQ.1) GO TO 180
HT(IR)=(V(I1,I8)+C(I1)*H'(I8+1)) / ALP(I8)
HST(IB)=HT(IB)-(U(IB)**2)/(2.*32.*17.*778.*16)
GO TO 190
180 HT(IR)=V(I1,I8)/ALP(I8)
HST(IB)=HT(IB)-(U(IB)**2)/(2.*32.*17.*778.*16)
GO TO 220
190 CONTINUE
IF (KOS.GT.0) GO TO 200
200 IF (KAS.TT.0) GO TO 210
CONTINUE
IF (KAS.GT.0) GO TO 220
IF (KAS.TT.0) GO TO 230
KAS=+10
IF (L3)=IR
GO TO 220
210 ND(3)=IR
KOS=+10
220 CONTINUE
IF (KAS.LT.0) NE(3)=IK
IF (NE(1),EQ.1,OR,NE(2),EQ.1,OR,NE(3),EQ.1) KD=+10
IF (NE(1),LT,NE(2)) GO TO 230
G(N,NE(2))
GO TO 240
230 G(N,NEF(1))
240 IF (MFT.LT,NEF(3)) GO TO 250
NFL=1
GO TO 240
250 NFL=1
260 IF (NFL.GT,NEF(2)) GO TO 270
270 NFL=1
\end{verbatim}
280 IF (MU.GT.NN(3)) GO TO 290
    IAP=NN(3)
    GO TO 300
290    IAP=MU
    IF (NEL.EQ.1) K0=10
    MN=NU+3-IA
    IF (TA.GT.0) GO TO 300
    TO=1-IA-6
    MNP1=MNP1+1
    DO 600 KK=1,MNP1
600    ZZ(1)*MNN)=2X/L2.
300    RETURN
C
310 FORMAT (2X,5MGDAUS,2X,3110)
320 FORMAT (2X,14,F22.10)
END
SUBROUTINE CHANGE(RUPT)
IMPLICIT REAL*R(A=H(0=Z)
COMMON R(51),S(51),T(51),Y(14,51),ROU(51),T(51),PS1(51)
COMMON UPR(51),VPR(14,51),HST(51),HT(51),TPR(51),KPR(51)
COMMON CM(51),X(14,51),AP(51),AV(51),UFR,UE1,RE01,RE,TE1,TEL
COMMON HFI,HEQ,CMN,CMO,PSI1,NNN,NNN,NEL,NEU
COMMON /SAI/ ZA
COMMON /RRR/ TO(51)
COMMON /AUU/ ZT(51)
COMMON /222/ IPR,TRF,TRU
COMMON /51/ DELX
COMMON /V1V/ YD(51)
COMMON /5A/ ARFX
PS1=PS1(NNN)
NPN=(NNN-1)/2+1
DPSI=PS1/NNN-11
DAX=DELX*X12.
2X=X
1X=(2X/12.)=DELX
ARFX=DELX
RUP=DELX/(DPSI**2)
W=DELX*X12.
2X=X*DELX
Z1X=DAX*12.
2X=X
DN 40 M1=1,NNN
IF ((M1-2).LT.0) GO TO 30

K 109
K 111
K 113
K 114
K 115
K 116
K 117
K 118
K 119
K 120
MA = MA + 2
10
PSI(M1) = PSI(MA)
AP(M1) = AP(MA)
AP2(M1) = AP2(MA)
R(M1) = R(MA)
U(M1) = U(MA)
UPR(M1) = UPR(MA)
UTP(M1) = UTP(MA)
4TPR(M1) = 4TPR(MA)
YD(M1) = YD(MA)

U(M1) = U(MA)
Z(M1) = Z(MA)
S(M1) = S(MA)
T(M1) = T(MA)
TPR(M1) = TPR(MA)

ON Z0 J = 0, ISJ

Y(J, M1) = Y(J, MA)
20
VPR(J, M1) = VPR(J, MA)
GO TO 40

MA = 1
30
GO TO 10

CONTINUE

NIL = NIL / 2 + 1
NIT = NIT / 2 + 1
NFL = NIL

41
NFI = NIT

WRITE (6, 50)
WRITE (5, 50) NEL, NFI
WRITE (6, 70) IAX, IAX, ZI, X, IX
RETURN

C

50 FORMAT (117, 2X, ZB, MK, NP POINTS HALVED)

60 FORMAT (2X, 5B, NEL, +15, 5B, NUE, +15)


51

12, F14.10, 77771

END

SUBROUTINE PRINT (KUSK, KUN)
IMPLICIT REAL (A-H, O-Z)
COMMON R(S1), S(S1), T(S1), Y14, S1, RON(S1), T(S1), PSI(S1)
COMMON UPR(S1), YPR(S1), RST(S1), HT(S1), TPR(S1), HTPR(S1)
COMMON CM(S1), X(X(S1), S1), AP(S1), APR(S1), NUE, UE1, RUE, RUD, UE1, UE1
COMMON HET, HEN, CM, CM1, DPE, NKN, NFL, NEU
COMMON /SAX/, ZX, XM
COMMON UOSET/ A(51), B(51), C(51), D(14,51), E(14,51), ALP(51), RAK, A  
ISKA, BUK, PISK, IRS, IKS, IAK, KOK  
COMMON /ZZZ/ ISP, IRE, IMU  
COMMON /SU/ DELX  
COMMON /S1G/ PA(14,14), PS(14), WI(14,51)  
COMMON /SYLT/ EI(14,51)  
Z = 2X/12.  
WRITE (6,30) Z, 2X, DELX, DDF, ISP, NEL, NEU, IRS, IKS, KUSK, KUN  
DO 20 I = 1, 1A  
VTOT = 0.0  
DO 10 J = 1, ISP  
10 VTOT = VTOT + (J - 1)  
WRITE (6,40) T, PSI(I), R(I), S(I), UT(I), AP(I), APR(I), MT(I)  
WRITE (6,50) T, (YJ, J = 1, ISP)  
WRITE (6,50) T, (WJ, J = 1, ISP)  
WRITE (6,50) T, (EJ, J = 1, ISP)  
WRITE (6,60) T, VTOT, GMT(I), ROO(I)  
20 CONTINUE  
WRITE (6,60) KO, UM  
RETURN  
END  
SUBROUTINE CHEMIE (KRYS, KRS, 1J)  
C SPECRT PRODUCTION SUBROUTINE - W/RH = U - 1./FEET  
C INPUT REQUIRED PRESSURE (ATM), AXIAL STEP SIZE (FT), # OF RADIAL MESH  
C POINTS (NNN), TEMP (DEG.K) AND SPECIE MASS FRACTIONS  
C REACTIONS CONSIDERED:  
H + O2 = OH + O  
O + H2 = OH + H  
H2 + OH = H + H2O  
2OH = O + H2O  
H2 + M = 2H + M  
H2O + M = OH + H + M  
OH + M = O + H + M  
O2 + M = 2O + M  
H2 + O2 = OH + OH  
F RYS = -VEY FROZEN FLOW  
F RYS = +VEY FINITE RATE CHEMISTRY  
IMPLICIT REAL*8(A-H, O-2)  
DIMENSION (CWTIR, ADU(22), C(JU(22), E(JU(22), D(22), A(22))
DIMENSION C(22), G(18)
DIMENSION AF(22), BF(22), AR(22), BR(22)
DIMENSION Z(14), F(22), R(22), FW(22), BW(22), CR(22)
DIMENSION PR(22), PK(18,22)
DIMENSION GNUP(14,17), GNUR(14,17)
COMMON R(51), S(51), U(51), V(14,51), ROU(51), T(51), PST(51)
COMMON (IPR(51), YPR(14,51), HST(51), HT(51), TPR(51)), TPR(51)
COMMON CM(51), X(14,51), AP(51), APR(51), UEN, UET, AHUE, AHUD, TED
COMMON REI, HEI, CMN, DNP, NNN, NEL, NEU
COMMON /SIG/ PA(14,14), PT(14), W(14,51)
COMMON /U00U/ ZZ(51)
COMMON /Z27/IP, IRE, IMU
COMMON /VYY/ IZE
COMMON /SU/ DELX
COMMON /SAX/ ZX, UM
COMMON /RRA/ TO(51)
COMMON /XXX/ TAU
COMMON /VVV/ VO(51)
KF=KRY+KRAS
P=1.00
IF (KF.LT.0) GO TO 50
CNV=453.5924/30.4801**3
CNV2=CNV**2
READ(5,600) (CW(J), J=1,ISP)
WRITE(6,700) (CW(J), J=1,ISP)
READ(5,601) (AU(K), K=1,IRE)
WRITE(6,701) (AU(K), K=1,IRE)
READ(5,601) (CV(K), K=1,IRE)
WRITE(6,701) (CV(K), K=1,IRE)
READ(5,602) (TE(K), K=1,IRE)
WRITE(6,702) (TE(K), K=1,IRE)
READ(5,602) (UT(K), K=1,IRE)
WRITE(6,702) (UT(K), K=1,IRE)
WRITE(6,703)
READ(5,505) (GNUP(J,K), J=1,ISP), K=1,IRE)
WRITE(6,705) GNUP
READ(5,605) (GNUR(J,K), J=1,ISP), K=1,IRE)
WRITE(6,705) GNUR
605 FORMAT(1/F10.0)
705 FORMAT(2X, 7F12.5)
DO 5 K=1, IRE
5 READ(5,405) AF(K), BF(K), AR(K), BB(K)
606 FORMAT(4F10.0)
READ(5,606) EN

IMP=IMP+1
DO 1 I=1,IRE
A(I)=DLOG(AUT(I)*CNV)
IF(I.GT.0. AND. I.LT.IMP) GO TO 2
C(I)=DLOG(CUT(I)*CNV)
GO TO 1
2 C(I)=DLOG(CUT(I)*CNV2)
1 CONTINUE
A(IRE)=DLOG(AUT(IRE))
C(IRE)=DLOG(CUT(IRE))
GO TO 20

50 I=1
IF(KRAS.EQ.0) GO TO 401
TO(I)=TO(I)+.5*(DX*(T(I)/UT(I)+1./UPR(I)))

401 CONTINUE
P=1.
GM=0.0
DO 10 J=1,ISP
G(J)=Y(J,1)/CW(J)
10 GM=GM+G(J)
SM=0.0
DO 11 J=1,ISP
Z(J)=G(J)/GM
11 SM=SM+Z(J)/CW(J)
RME=1.545375SM
RMP=2116.27/(RME*T(I)**1.4)
DO 13 LK=1,IRE
F(LK)=DEXP(T(LK)/E(LK)/T(I))
R(LK)=DEXPIC(LK)-D(LK)/T(I))
13 CONTINUE
B(5)=B(5)/T(I)
F(7)=F(7)/T(I)**0.5
F(8)=F(8)/T(I)
F(9)=F(9)/T(I)**1.5
F(11)=F(11)/T(I)
F(13)=F(13)/T(I)
B(13)=B(13)/T(I)
F(15)=F(15)/T(I)**0.5
B(15)=B(15)/T(I)**0.5
F(17)=F(17)/T(I)**0.3/T(I)**111.0-0.5
FW(1)=F(1)**RODG(1)**G(5)
RW(1)=R(1)**RODG(4)**G(2)
FW(2) = F(2) * R * G(6) * G(1)
BW(2) = B(2) * R * G(4) * G(1)
FW(3) = F(3) * R * G(6) * G(4)
BW(3) = B(3) * R * G(1) * G(3)
FW(4) = F(4) * R * G(4) * G(4)
BW(4) = B(4) * A * G(3) * G(2)
FW(5) = F(5) * R * G(6) * G(6)
FW(6) = F(6) * R * G(3) * G(6)
FW(7) = F(7) * R * G(7) * G(6)
BW(7) = B(7) * R * A * G(9) * G(8) * GM
FW(8) = F(8) * R * G(5) * G(6)
BW(8) = B(8) * R * A * G(1) * G(2) * G(2) * GM
FW(9) = R * G(9) * G(4) * GM
BW(9) = B(9) * R * A * G(8) * G(2) * GM
FW(10) = F(10) * R * G(10) * G(6)
BW(10) = B(10) * R * A * G(7) * G(2) * GM
FW(12) = F(12) * R * G(12) * G(4)
FW(13) = F(13) * R * G(9) * G(2)
FW(14) = F(14) * R * G(9) * G(8)
BW(14) = B(14) * R * A * G(7) * G(12)
FW(15) = F(15) * R * G(4) * G(5)
FW(16) = F(16) * R * G(9) * G(9)
BW(16) = B(16) * R * A * G(10) * G(2)
FW(17) = F(17) * R * G(14) * G(15)
IF(P1 - [GE.1]) CONTINUE
Z2(11) = ZX12*
IF(X14 < 0) 17
IF(X14 > 0) 18
IF(X0 < 0) Z2(0)
IF(X14 > 0) 18
17 FW(17) = 0.0
18 BW(17) = 0.00
IF(Z0 * GT.0) FW(17) = 0.
DO 15 M = 1, IE
CR(M)=FW(M)-BW(M)
PR1(M)=(FW(M)*((AF(M)-BF(M))*T(I)+E(M)))-BW(M)*((AR(M)-BR(M))*T(I)+
M1(M))/(T(I)**2)

15 CONTINUE
PR1(I,RE)=FW(I,RE)*(1.0/(T(I)-55.5)+E(I,RE)/(T(I)**2))
PR1(RE)=0.
IF(IZE.GT.0) PR1(RE)=0.
1:(KNAS,GT.0) GO TO 3
[REM]=1/RE-1
[DO 71 I=1,ISP
[DO 71 J=1,REM]
PK(I,J)=FW(J)*(GNUM(I,J)+G(I,J)-BF(J)*SM)-BW(J)*((GNUM(I,J)+G(I,J))
-L-99(J)*SM)
[IF(J,GE,5,AND,J.LE,1M) GO TO 72
GO TO 71
72 PK(I,J)=PK(I,J)+(FW(J)-BW(J))/GM
71 PK(I,J)=PK(I,J)/GW(I,1)
[DO 73 I=1,ISP
PK(I,RE)=0.0
73 CONTINUE
IF(IZE.GT.0) GO TO 3
3 CONTINUE
[IF(KNAS.EQ.0) GO TO 4
W1(I,J)=CWI(1)*-CR(1)+CR(2)+CR(3)+2.0*CR(5)+CR(6)+CR(12))
W2(J,I)=CWI(2)*-CR(1)+CR(2)+CR(4)+2.0*CR(7)+CR(8)+CR(10)+CR(11)-CR(1
+3)+CR(14)+CR(15)+CR(16)
W3(I,J)=CWI(3)*CR(3)+CR(4)+CR(5)
W4(I,J)=CWI(4)*CR(1)+CR(2)+CR(3)-2.0*CR(4)+CR(6)-CR(17)
W5(I,J)=CWI(5)*-CR(1)-CR(10)+CR(13)-CR(15)-EN/2.0*CR(17)
W6(I,J)=CWI(6)*-CR(2)-CR(3)-CR(5)+EN/1.0*CR(17)
W7(I,J)=CWI(7)*-CR(7)+CR(10)-CR(14)
W8(I,J)=CWI(11)*-CR(1)-CR(2)+CR(7)+CR(9)-CR(13)-CR(14)
W9(I,J)=CWI(9)*-CR(9)+CR(11)-CR(13)-CR(14)-CR(15)-2.0*CR(15)
W10(I,J)=CWI(10)*-CR(10)+CR(16)
W11(I,J)=CWI(11)*-CR(11)+CR(15)
W12(I,J)=CWI(12)*-CR(12)+EN*CR(17)
W13(I,J)=CWI(13)*4(CR(12))
W14(I,J)=CWI(14)*-EN(17)
4 CONTINUE
PR1(1)=CWI(1)*-PR1(1)+PR1(2)+PR1(3)+2.0*PR1(5)+PR1(6)+PR1(12))
PR1(12)=CWI(2)*PR1(1)+PR1(2)+PR1(4)+2.0*PR1(8)+PR1(9)+PR1(10)+PR1(11)
PR1(13)+PR1(14)*PR1(15)+PR1(16)


```plaintext
PY(3)=CW(3)*((PRT(3)+PRT(4)-PRT(6))
PY(4)=CW(4)*((PRT(1)+PRT(2)-PRT(3)-2.0*PRT(4)+PRT(6)-PRT(2))
PY(5)=CW(5)*(-PRT(1)-PRT(8)+PRT(13)-PRT(15)-EN/2.0*PRT(17))
P(6)=CW(6)*(-PRT(2)-PRT(3)+PRT(5)+(EN+1.0)*PRT(17))
PY(7)=CW(7)*(-PRT(7)+PRT(10)+PRT(14))
P(8)=CW(8)*(-PRT(9)+PRT(13)-PRT(14))
P(9)=CW(9)*(-PRT(9)+PRT(11)-PRT(13)-PRT(14)-PRT(15)-2.0*PRT(16))
P(10)=CW(10)*(-PRT(10)+PRT(14))
P(11)=CW(11)*(-PRT(10)+PRT(15))
P(12)=CW(12)*(-PRT(11)+EN*PRT(17))
P(13)=CW(13)*(PRT(12))
P(14)=CW(14)*(-PRT(17))
DU 21 L=1,15P
PAIL(1)=CW(1)*(-PK(L,1)+PK(L,2)+PK(L,3)+2.0*PK(L,5)+PK(L,6)+PK(L,12))
PAIL(2)=CW(2)*(-PK(L,1)+PK(L,2)+PK(L,4)+2.0*PK(L,8)+PK(L,9)+PK(L,10)
1+PK(L,11)-PK(L,13)+PK(L,14)+PK(L,15)+PK(L,16))
PAIL(3)=CW(3)*PK(L,3)+PK(L,4)-PK(L,6))
PAIL(4)=CW(4)*(PK(L,1)+PK(L,2)-FK(L,3)-2.0*PK(L,4)+PK(L,6)-PK(L,12)
1))
PAIL(5)=CW(5)*-(PK(L,1)+PK(L,4)+PK(L,13)-PK(L,15)-EN/2.0*PK(L,17))
PAIL(6)=CW(6)*-(PK(L,1)-PK(L,3)-PK(L,5)+EN+1.0)*PK(L,17))
PAIL(7)=CW(7)*-(PK(L,1)-PK(L,3)+PK(L,10)+PK(L,14))
PAIL(8)=CW(8)+2.0*PK(L,7)+PK(L,9)+PK(L,13)+PK(L,14))
PAIL(9)=CW(9)*-(PK(L,9)+PK(L,11)-PK(L,13)-PK(L,14)-PK(L,15)-2.0*PK
1(L,16))
PAIL(10)=CW(10)*-(PK(L,10)+PK(L,16))
PAIL(11)=CW(11)*-(PK(L,11)+PK(L,15))
PAIL(12)=CW(12)*-(PK(L,12)+EN*PK(L,17))
PAIL(13)=CW(13)*-(PK(L,12))
PAIL(14)=CW(14)*-(PK(L,17))
21 CONTINUE
IF(KRYS,NE,-10) GO TO 20
20 CONTINUE
RETURN
600 FORMAT(7*10.0)
601 FORMAT(5*10.4)
602 FORMAT(4*H4.4)
700 FORMAT(2X,1F14.4)
701 FORMAT(2X,9E14.4)
702 FORMAT(2X,9F14.2)
703 FORMAT(*1*)
```
2442261000 01.1227618700E-03 -.49927190E-07 .21400A3000E-10
-12511080E-14 .5614882100E 05
.3152936000F 01.1405995500E-02 -.570784620E-06 .1062820900E-09
-737207830E-14 .9852204400E 06
.4626547900E 01.3021680700E-02 -.12150140E-05 .2959952200E-09
-15A497010E-13 .8536645000E 06
.4613921900E 01.2638663900E-02 -.10948450E-05 .2081425000E-09
-146543910E-13 .2340378000E 04
.2951151900E 01.1552556700E-02 -.619114110E-06 .1135033600E-09
-74457320E-14 -.142318270E 05
.4427926000E 01.3192896000E-02 -.129742300E-05 .2414744600E-09
-167429860E-13 -.489440430E 05
.7159908600E 02
-.213479840E 05
52097.7 59556.6 -57797.9 9312.5
193024.6 21600. 19490. 8007.5 -25615.7 -94051.8 -59670
1.008 16. 18.016 17.008 32. 2.016 28.017
14.008 30.008 .44.016 46.008 28.011 44.011 142.286
.26000E 13.20000E 09
.45000E 14.14000E 13.32000E 15.83000E 14.70000E 18
.15000E 17.61000E 15.10000E 15.90000E 15.18200E 14
.20000E 17.47000E 15.33000E 11.70000E 14.580000E 11
.14200E 15.10000E 10
.504. 40426. 3019. 503. 55353. 62196. 113171. 59378. 75441.
30649. 37237. 3875. 19675. 0. 23851. 32105. 6914.
503. 3019. 10618. 9108. 0. 0. 0. 0. 0. 10769. 0. 13712. 3563. 37992. 0. 14090. 0.
1. 1.
1.
1.
1.
1.
2.
1.
An experimental and theoretical study has been made of the history of the pollutants carbon monoxide (CO), unburned hydrocarbons (HC) and nitrogen oxides (NOx) in a turbojet afterburner. Experimental traverses at several axial stations were performed in a simulated afterburner in which exhaust from a J-47 combustor can, operated at medium power, was mixed with fuel spray. Experiments were carried out both in a non-bypass and in a bypass configuration (secondary air was mixed with primary exhaust). The non-bypass tests were carried out at high combustor efficiency, and yielded the following: CO = 300 ppm, HC < 10 ppm, NOx = 100 ppm. In the bypass tests, fuel distribution was non-uniform and combustor efficiency was low. The concentrations obtained were CO = 10000 ppm, HC = 1000 ppm, NOx = 100 ppm. The theoretical analysis consisted of a computer program for reacting flow with turbulent mixing. The computer program was very slow and therefore of limited usefulness in terms of cost and questionable results, since it could not be checked against experiment. Infrared measurements of NO in the combustion tunnel were attempted. Indications were obtained of NO at the 5.3 micron band, but quantitative measurements were not obtained.
### 14. KEY WORDS

<table>
<thead>
<tr>
<th>KEY WORDS</th>
<th>LINK A</th>
<th>LINK B</th>
<th>LINK C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turbine engine pollution</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exhaust emissions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computer model</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pollution</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>