AD735878

NOLTR 71-210

A COMPUTER PROGRAM FOR CO_2-N_2-H_2O GASDYNAMIC LASER GAIN AND MAXIMUM AVAILABLE POWER

By
Walter J. Glowacki
John D. Anderson, Jr.

30 OCTOBER 1971

NOL
NAVAL ORDNANCE LABORATORY, WHITE OAK, SILVER SPRING, MARYLAND

Reproduced by
NATIONAL TECHNICAL
INFORMATION SERVICE
Springfield, Va. 22151

NOLTR 71-210

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED

D D C
RECEIVED
JAN 26 1972
C

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY (Corporate author) Naval Ordnance Laboratory, White Oak, Maryland 20910	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

A Computer Program For CO_2-N_2-H_2O Gasdynamic Laser Gain and Maximum Available Power

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

5. AUTHOR(S) (First name, middle initial, last name)

Walter J. Glowacki John D. Anderson, Jr.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
30 October 1971	30	8

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO. ORD 551/020/128/1 U-1754	NOL TR 71-210
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Ordnance Systems Command, Washington, D.C.

13. ABSTRACT

A Fortran IV computer program is presented for the calculation of small-signal gain and maximum available energy for CO_2-N_2-H_2O gasdynamic lasers. In comparison to an earlier version, the present program contains several computational improvements which result in a dramatic, order-of-magnitude reduction in computation time. These improvements are discussed in detail.

DD FORM 1 NOV 65 1473 (PAGE 1)
S/N 0101-807-6801

UNCLASSIFIED
Security Classification

Unclassified
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
gasdynamic laser						
computer program						
CO_2						

DD FORM 1 NOV 65 1473 (BACK)
(PAGE 2)

Unclassified
Security Classification

A COMPUTER PROGRAM FOR CO_2-N_2-H_2O GASDYNAMIC LASER GAIN AND MAXIMUM AVAILABLE POWER

Prepared by

Walter J. Glowacki and John D. Anderson, Jr.

ABSTRACT: A Fortran IV computer program is presented for the calculation of small-signal gain and maximum available energy for CO_2-N_2-H_2O gasdynamic lasers. In comparison to an earlier version, the present program contains several computational improvements which result in a dramatic, order-of-magnitude reduction in computation time. These improvements are discussed in detail.

Naval Ordnance Laboratory
Silver Spring, Maryland

NOLTR 71-210 30 October 1971

A COMPUTER PROGRAM FOR CO_2-N_2-H_2O GASDYNAMIC LASER GAIN AND MAXIMUM AVAILABLE POWER

This report presents a computer program for the calculation of small-signal gain and maximum available energy for CO_2-N_2-H_2O gasdynamic lasers. A complete statement listing and discussion of the program is given.

This work is based on a time-dependent nonequilibrium nozzle analysis for population inversions originally jointly sponsored by the NOL Independent Research Funds and the Office of Naval Research. The improvements made in the computer program, as discussed in the present report, were supported by the Naval Ordnance Systems Command, under ORD Task 551/020/128/1 U-1754.

ROBERT WILLIAMSON II
Captain, USN

L. H. Schindel
L. H. SCHINDEL
By direction

CONTENTS

ILLUSTRATIONS

NOMENCLATURE

a_o	frozen speed of sound in reservoir
c_i	mass fraction of species i
e_{vib}	vibrational energy per unit mass of mixture
e_{vib_I}	vibrational energy per unit mass of mode I (see references 1-3)
$e_{vib_{II}}$	vibrational energy per unit mass of mode II (see references 1-3)
e_{max}	maximum available laser energy
k	Boltzmann constant
L	length of nozzle or duct
N_{001}	particles per unit volume in upper laser level
N_{100}	particles per unit volume in lower laser level
p_o	reservoir pressure
Q	partition function for CO_2
R	specific gas constant of mixture
T_o	reservoir temperature
T_{vib_I}	vibrational temperature of mode I (see references 1-3)
$T_{vib_{II}}$	vibrational temperature of mode II (see references 1-3)
$T_{vib_{II}}^o$	vibrational temperature of mode II when population inversion is zero

X_i	mole fraction of species i
ε_{001}	energy (per particle) of the upper laser level
ε_{100}	energy (per particle) of the lower laser level

BLANK PAGE

I. INTRODUCTION

A time-dependent analysis for the vibrational nonequilibrium nozzle flows associated with CO_2-N_2 gasdynamic lasers is described in references 1-3. Results obtained from this analysis for small-signal laser gain are in reasonable agreement with experimental data (references 4-7).

In 1969, a computer program based upon the above analysis was written at NOL for internal use as a research tool. However, in light of numerous requests for this program, and because of several recent improvements in its computational efficiency, the publication of this program now seems to be appropriate. Therefore, in Appendix A of this report, a complete statement listing (in Fortran IV) and user instructions for the time-dependent gasdynamic laser gain computer program are given.

In comparison to the original 1969 version, the present program contains several recent improvements and additions. They are: (a) a striking, order-of-magnitude reduction in running time, (b) the addition of a short calculation of maximum available laser energy, (c) the inclusion of equilibrium H_2O energy terms in the energy equation. Details of these improvements are described in the following sections. Throughout the remainder of this report, familiarity with the analysis of references 1-3 will be assumed.

II. IMPROVED RUNNING TIME

In the original version, a typical running time for the computation of gain for a given nozzle flow (for a prescribed p_o, T_o, X_i, nozzle shape, size and area ratio) ranged from 300 to 600 seconds on a CDC 6400 digital computer, depending on the particular case. However, several changes have recently been made to the program, resulting in an order-of-magnitude reduction in the computation time. These time-saving changes resulted from the observation that the flow in the subsonic and throat regions reached a steady-state in far fewer time steps than did the flow near the nozzle exit. A typical example was a high area ratio nozzle flow calculated using 38 grid points, 22 of which were finely spaced through the subsonic and throat regions. The other 16 grid points were more coarsely spaced through the remainder of the nozzle. The results showed that, although about 6000 time steps were necessary to reach steady state near the nozzle exit, the subsonic and throat regions reached the final values after only 600 time steps. From these results, it was clear that approximately one-half the computing time could be saved by dropping calculations at the finely spaced grid points as soon as they reach their steady state values.

In order to be able to drop these grid points in an efficient manner, it was necessary to devise a test to determine when the steady state was reached. Examination of existing results showed that the time derivatives of the vibrational energies of modes I and II at the next to last fine-spaced grid point (NB-1) are the best indicators. Because these derivatives oscillate in sign as they damp to zero, their absolute values are required to be less than some given value (.005 was found to be a good value) in three consecutive tests spaced ten time steps apart. When these tests are satisfied, the first NB-1 grid points are dropped from further calculations. The same test procedure is then applied to the next to last grid point (NN-1) and the calculation is terminated when the tests are satisfied.

When the NB-1 fine-spaced grid points are dropped from a nozzle calculation, an even greater time saving can be obtained because the time step can be increased by as much as an order of magnitude, speeding the approach to the steady state. The Courant-Friedrichs-Lewy stability criterion can then be based on the relatively large interval between the more coarsely spaced grid points in the downstream portion of the nozzle rather than on the smaller interval used in the upstream portion. The second stability criterion, based on the characteristic relaxation time for the fastest finite-rate molecular relaxation process occurring within the mixture, also allows a greater time step once the fine-spaced grid points are dropped because the lower temperatures prevailing in the downstream region lead to much longer relaxation times.

The changes described above resulted in about an order of magnitude decrease in computing time and made it feasible to stack multiple cases into a single computer run. The solution for each case could then be used as a starting guess for the next case. Since multiple cases are frequently run to trace out variations with supply conditions, mole fractions, etc., the solutions for successive cases are usually closer to each other than to those for an equilibrium flow. Thus, starting from a previous solution can again cut the computing time. Stacked cases have run in as little as one third the time required for separate runs.

III. MAXIMUM AVAILABLE LASER ENERGY

The present program includes a simple calculation of maximum available energy, as follows. Consider a point in the nozzle of a gasdynamic laser. Using the notation of references 1-3, the translational and rotational temperature at this point is T, the sum of the vibrational energy contained in the excited N_2 and CO_2 (ν_3) is $e_{vib_{II}}$ with an attendent vibrational temperature $T_{vib_{II}}$, the sum of the vibrational energy in the CO_2 (ν_1) and CO_2 (ν_2) modes is e_{vib_I} with an attendent T_{vib_I}, and the population densities of the upper and lower laser levels are N_{001} and N_{100}, respectively. In general, $T_{vib_{II}} > T_{vib_I} > T$. When an inversion exists, then by definition $N_{001} - N_{100} > 0$. The population densities are given by

$$N_{001} = (N_{CO_2}/Q)\ e^{-\varepsilon_{001}/kT_{vib_{II}}}$$

and

$$N_{100} = (N_{CO_2}/Q)\ e^{-\varepsilon_{100}/kT_{vib_I}}$$

where Q is the partition function. We ask the following question: If energy is drained from $e_{vib_{II}}$, and e_{vib_I} is held constant, at what value of $T_{vib_{II}}$ will the population inversion go to zero. Denote this value of $T_{vib_{II}}$ by $(T_{vib_{II}}^{o})$. Then,

$$N_{001} - N_{100} = 0 = (N_{CO_2}/Q)\ (e^{-\varepsilon_{001}/k(T_{vib_{II}}^{o})} - e^{-\varepsilon_{100}/kT_{vib_I}})$$

Hence,

$$\varepsilon_{001}/k(T_{vib_{II}}^{o}) = \varepsilon_{100}/kT_{vib_I}$$

Thus,

$$(T_{vib_{II}}^{o}) = T_{vib_I}\ (\varepsilon_{001}/\varepsilon_{100})$$

When $T_{vib_{II}} > (T_{vib_{II}}^{o})$, an inversion exists and laser power can in principle be extracted. When $T_{vib_{II}} \leq (T_{vib_{II}}^{o})$, no inversion is present, and no power can be extracted.

In order to consider the maximum laser energy available, we assume that $T_{vib_I} = T$.

Then,

$$(T_{vib_{II}}^{o}) = T\ (\varepsilon_{001}/\varepsilon_{100})$$

Using this value of $(T_{vib_{II}}^{o})$, we define a maximum available laser energy as

$$e_{max} \equiv [e_{vib_{II}}\ (T_{vib_{II}}) - e_{vib_{II}}\ (T_{vib_{II}}^{o})] \times 0.409$$

where the factor 0.409 is the quantum efficiency for the CO_2 laser transition at 10.6μ. In the present computer program, the symbol EVMAX represents e_{max}/RT_o.

The quantity e_{max} is a convenient index to gage the amount of power that might be extracted from a gasdynamic laser. However, in reality the actual power extraction is usually less than e_{max} due to losses in the laser cavity.

IV. EQUILIBRIUM H_2O IN THE ENERGY EQUATION

The presence of small amounts of H_2O in the CO_2-N_2 gasdynamic laser expedites the lasing action (references 1-3). In previous versions of the computer program, H_2O was fully included in the vibrational kinetics, i.e., in the computation of the relaxation times. However, because the amount of H_2O in typical gasdynamic lasers was on the order of one percent, the vibrational energy of H_2O was intentionally neglected in the gasdynamic energy equation in order to conserve running time. That is, $e_{vib} = C_{CO_2} e_{vib_{CO_2}} + C_{N_2} e_{vib_{N_2}}$ in the energy equation. Hence, the thermodynamics of the flow did not explicitly see the presence of H_2O. However, for amounts of H_2O on the order of ten percent, $e_{vib_{H_2O}}$ should be explicitly included in the energy equation. Therefore, for completeness in the present computer program, the mixture vibrational energy is precisely written as $e_{vib} = C_{CO_2} e_{vib_{CO_2}} + C_{N_2} e_{vib_{N_2}} + C_{H_2O} e_{vib_{H_2O}}$. The H_2O vibrational energy, $e_{vib_{H_2O}}$, is assumed to be in equilibrium with T. This appears to be justifiable in light of the extremely fast T-V relaxation of H_2O shown in reference 8.

The effect of the above change on gain is shown in Figure 1 as a function of H_2O content. The upper curve was obtained with the older version of the program where $e_{vib_{H_2O}}$ is neglected (this is almost the same effect as saying $e_{vib_{H_2O}}$ is frozen), and the lower curve was obtained with the present program where equilibrium $e_{vib_{H_2O}}$ is included in the energy equation. Note that, for H_2O content from zero to two percent, the differences are small, as expected.

REFERENCES

1. Anderson, J.D., Jr., "Time-Dependent Analysis of Population Inversions in an Expanding Gas," The Physics of Fluids, Vol. 13, No. 8, August 1970, pp 1983-1989

2. Anderson, J.D., Jr., "A Time-Dependent Quasi-One-Dimensional Analysis of Population Inversions in an Expanding Gas," NOLTR 69-200, December 1969

3. Anderson, J.D., Jr., "Numerical Experiments Associated with Gas Dynamic Lasers," NOLTR 70-198, September 1970

4. Anderson, J.D., Jr., Humphrey, R.L., Vamos, John S., Plummer, M.J., and Jensen, R.E., "Population Inversions in an Expanding Gas: Theory and Experiment", to be published in The Physics of Fluids

5. Anderson, J.D., Jr., Humphrey, R.L., Vamos, John S., Plummer, M.J., and Jensen, R.E., "Population Inversions in an Expanding Gas: Theory and Experiment", NOLTR 71-116, 28 June 1971

6. Anderson, J.D., Jr. and Winkler, E.M., "High Temperature Aerodynamics with Electromagnetic Radiation," Proceedings of the IEEE, Vol. 59, No. 4, April 1971, pp 651-658

7. Lee, G. and Gowen, F.E., "Gain of CO_2 Gasdynamic Lasers," Applied Physics Letters, Vol. 18, No. 6, 15 March 1971, pp 237-239

8. Taylor, R.L. and Bitterman, S., "Survey of Vibrational Relaxation Data for Processes Important in the CO_2-N_2 Laser System," Reviews of Modern Physics, Vol. 14, No. 1, January 1969, pp 26-47

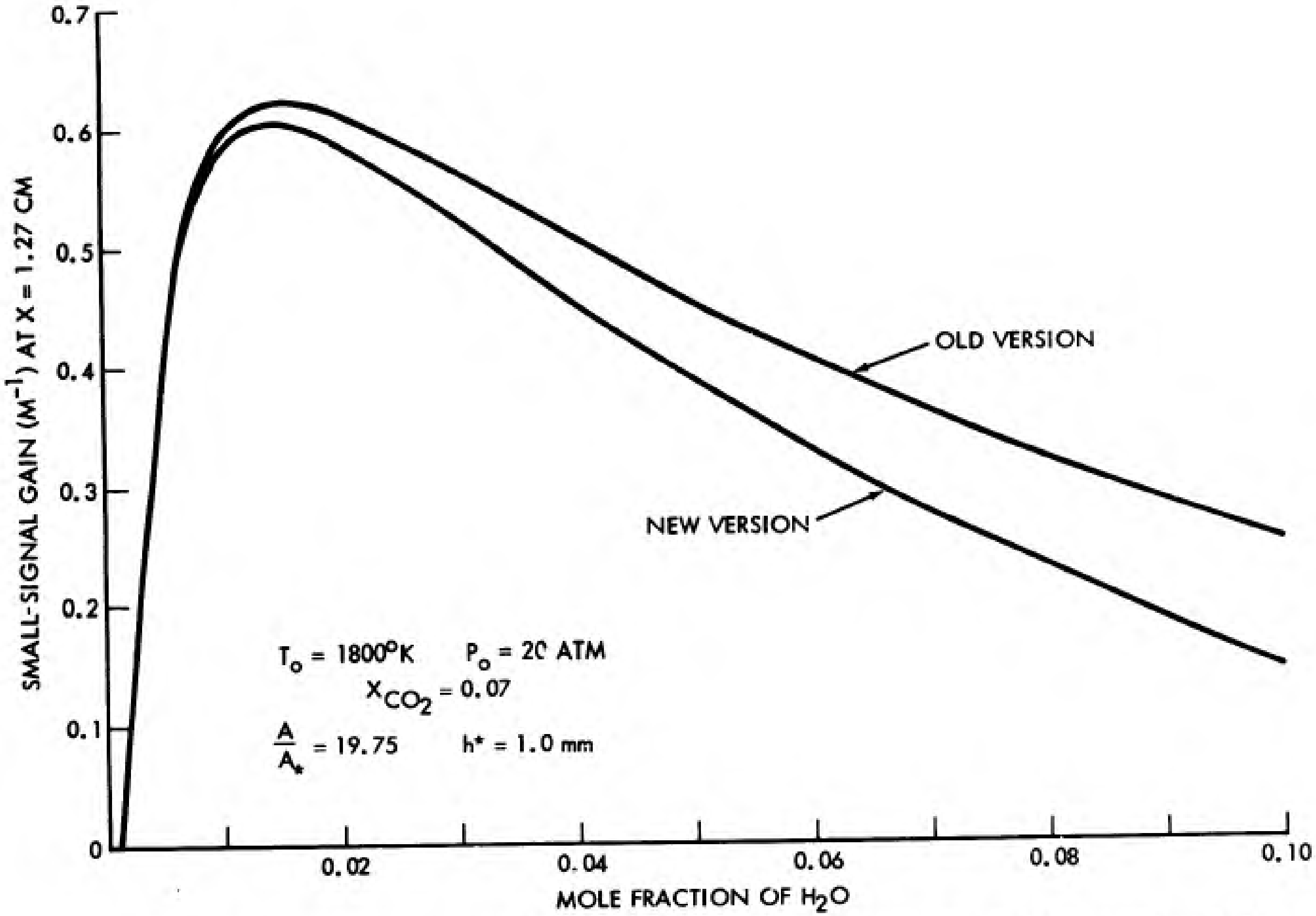

FIG. 1 Calculated small-signal gain as a function of H_2O content at a location 1.27 cm. downstream of a contoured, minimum length nozzle. A comparison is made between results from the old version of the computer program where $e_{vib_{H_2O}}$ is not included in the energy equation, and the new version which does include equilibrium $e_{vib_{H_2O}}$ in the energy equation (see text for details).

BLANK PAGE

APPENDIX A

FORTRAN IV COMPUTER PROGRAM AND USER INSTRUCTIONS

INTRODUCTORY REMARKS

This appendix contains a listing of the Fortran IV computer program* and other information necessary for its use. Brief descriptions of the program input and output are given. Input data and selected results for a sample case are presented in order to illustrate input/output and to provide a test case for program users.

As mentioned previously, the program can be used to compute the flow in supersonic nozzles and ducts (e.g. a laser cavity). Nozzle flow is defined as having a sonic point and being in equilibrium at the first grid point. Duct flow has no sonic point and can be in or out of equilibrium at the first grid point according to the values assigned to the first grid point (see card group 5 under PROGRAM INPUT).

Stacking data sets for cases, each of which is not too different from the preceeding, can result in computer time savings ranging from negligible to substantial. The not-too-different requirement is vague at the present time but derives from the use of the steady-state solution for each case as the first trial solution for the

*The program utilizes a CDC 6400 library function SECOND(A) to monitor the elapsed CP (central processor) time. The CP time is returned both via the argument and via the normal function return. If such a function is not available, the statement SECOND(CPTIM) = 0. Should be entered on the line numbered GDL00140.

succeeding case. This requirement would seem to preclude intermixing nozzle and duct cases and increasing the number of grid points (NN). The use of non-dimensionalized variables through most of the computational procedure contributes greatly to the success of the stacked data approach, permitting larger changes in the input data between successive cases. Although only stacked cases in which the reservoir temperature and pressure and the gas composition were changed systematically have been run, all stacked cases have run successfully and, as far as is known, at least as quickly as the same case run individually.

It should be pointed out that in a parametric variation of the input data over some given range, the required computing time for stacked cases increases rather slowly as the number of cases increases. For example, if the number of cases covering the desired range of data is increased by 100 percent (doubled), the computing time may increase by less than 50 percent. The reason for this is that successive cases (and consequently their solutions) are nearer to each other, reducing the computing time per case. Since additional cases are more costly to run at a later date, one should attempt to include all of the necessary cases in the first run.

PROGRAM INPUT

The data for this program is input from punched cards in up to five card groups, each consisting of one or more cards.

Group 1 Required for each case. This group consists of one card giving the distribution of grid points along the centerline. Since this program is set up to calculate multiple cases (if desired), it will automatically return to this READ after completing each case. Therefore, a blank card is used after the last data set to stop the program normally.

Card Columns	Format Used	Variable Name	Definition
1-5	I5	NN	Total number of grid points to be used is /NN/. If NN<0, card group 3 is omitted. If NN=0, program will stop.
6-10	I5	NB	Number of grid points with first spacing (DXONE in program)
11-15	I5	NT	Grid point at which throat occurs. If no throat (i.e. duct flow), use NT=0.

Group 2 Required for each case. This group consists of one card giving the length in inches of the subsonic (XSUB) and supersonic (XSUP) portions of the flow. The format used is 2F10.5. For duct flow (NT=0), set XSUB equal to zero and XSUP equal to the desired duct length.

Group 3 Required only when NN>0 in card group 1. This group consists of up to 14 cards giving the area ratios A/A_* (where A_* is the area of the throat) at each of the NN meshpoints. The format used is 7F10.5 (seven per card). If NN<0, the area ratios from the preceeding case are used. This eliminates the need for re-reading the same area ratio values a number of times during a multiple case run.

Group 4 Required for each case. This group consists of one card giving the gas mixture and reservoir conditions as indicated below.

Card Columns	Format Used	Variable Name	Definition
1-10	F10.5	FMC	Mole fraction of CO_2
11-20	F10.5	FMH	Mole fraction of H_2O

Card Columns	Format Used	Variable Name	Definition
21-30	F10.5	ST	Reservoir temperature (°K)
31-40	F10.5	SP	Reservoir pressure (atm.)
41-45	I5	JFIN	Upper limit to number of time steps. If zero is read, program sets JFIN=501 for duct flow (NT=0), otherwise sets JFIN=2001. These default values may not be large enough for all cases.
46-50	I5	JTEM	Upper limit to number of iterations in determining vibrational temperatures. If zero is read, program sets JTEM=1000.

Group 5 Required only for duct flow (NT=0). This group consists of one card giving the conditions at the first grid point. If the duct flow calculation is a continuation (e.g. after a nozzle flow calculation), the required values are given in the output of the upstream calculation.

Card Columns	Format Used	Variable Name	Definition
1-10	F10.5	A(1)	Area ratio (must match first value of last group 3 used).
11-20	F10.5	RHO(1)	Density ratioed to reservoir density
21-30	F10.5	U(1)	Flow velocity ratioed to reservoir frozen speed of sound
31-40	F10.5	T(1)	Temperature ratioed to reservoir temperature
41-50	F10.5	EVC(1)	e_{vib_I}/RT_o (see references 1-3)

Card Columns	Format Used	Variable Name	Definition
51-60	F10.5	EVN(1)	$e_{vib_{II}}/RT_o$ (see references 1-3)

PROGRAM OUTPUT

In addition to the input data and the initial conditions, values of the more significant variables are printed every 50 time steps for nozzle calculations or every 10 time steps for duct calculations (which typically converge more quickly). This output is also printed upon convergence of both the fine- and coarse-spaced grid points. Vibrational temperatures and populations and small-signal gain calculated from the steady state results are also printed. The output variables and their identification are given below. For fuller explanations, see references 1-3.

TIME	non-dimensional time ta_o/L
DELTIM	smaller of FLUID DELTIM and KINETIC DELTIM
J	index for time step
CPTIME	elapsed CP (central processor) time
DELCPTIME	CP time since last output step
FLUID DELTIM	increment of TIME allowed by Courant-Friedrichs-Lewy stability criterion
KINETIC DELTIM	increment of TIME allowed by fastest relaxation process
I	index for grid point
DTRHO	time derivative of RHO
DTU	time derivative of U
DTT	time derivative of T
DTEVN	time derivative of EVN
EQEVC	equilibrium value of EVC at translational temperature
EQEVN	equilibrium value of EVN at translational temperature

A	area ratioed to throat area
RHO	density ratioed to reservoir density
U	flow velocity ratioed to reservoir frozen speed of sound
T	temperature ratioed to reservoir temperature
P	pressure ratioed to reservoir pressure
EV	non-dimensional vibrational energy $\frac{e_{vib}}{RT_o}$
EVC	non-dimensional vibrational energy of mode I, e_{vib_I}/RT_o
EVN	non-dimensional vibrational energy of mode II, $e_{vib_{II}}/RT_o$
TAUC	non-dimensional vibrational relaxation time for mode I, $\tau_I\ a_o/L$
TAUN	non-dimensional vibrational relaxation time for mode II, $\tau_{II}\ a_o/L$
T1 (K)	vibrational temperature of mode I (°K)
T2 (K)	vibrational temperature of mode II (°K)
T(K)	translational temperature (°K)
N	number of CO_2 particles per unit volume (cm^{-3})
N001	particles per unit volume in upper laser level (mode II), (cm^{-3})
N100	particles per unit volume in lower laser level (mode I), (cm^{-3})
X(CM)	distance from first grid point (cm)
EVMAX	maximum available laser energy, e_{max}/RT_o
GO	small-signal gain (m^{-1})

SAMPLE CASE

Selected results from a sample nozzle calculation are presented on the following pages to illustrate the program output and to provide a test case for program users. On a CDC 6400 computer, the calculation required 680 time steps and 64 seconds of CP (central processor) time to reach the steady state solution at the nozzle exit. The NB-1 fine-spaced grid points reached the steady state in

440 time steps and 55 seconds of CP time. The tabulated values of TIME indicate that if the fine-spaced grid points had not been dropped, allowing the time step to increase, then approximately (3.4100/.2996)x440 or 5000 time steps would have been required to reach the same degree of convergence (same value of TIME). The required computing time would be about 630 seconds, thus, for this case, the current procedure reduces the required computing time by a factor of ten.

The input data for this sample case is tabulated below.

NN	NB	NT
21	16	11

XSUB	XSUP
.09046	2.26151

A(1)-A(21)

4.20	3.20	2.50	2.10	1.70	1.50	1.40
1.30	1.20	1.10	1.00	1.35	1.70	2.05
2.40	2.75	12.25	16.50	18.50	19.50	19.75

FMC	FMH	ST	SP
.07	.01	2400.	20.

CPTIME = 1.437

INPUT DATA

TEMPERATURE(K)= .24000000E+04 PRESSURE(ATM)= .20000000E+02

CO2= .70000000E-01 N2= .92000000E+00 H2O= .10000000E-01

NA= 21 NB= 16 NT= 11 XSUB= .90460000E-01 XSUP= .22615100E+01

INITIAL CONDITIONS

I	X	A	RHO	T	U	P
1	0.000	.4200E+01	.9903E+00	.9961E+00	.1391E+00	.9865E+00
2	.004	.3200E+01	.9832E+00	.9932E+00	.1839E+00	.9765E+00
3	.008	.2500E+01	.9719E+00	.9887E+00	.2382E+00	.9608E+00
4	.012	.2100E+01	.9592E+00	.9835E+00	.2873E+00	.9434E+00
5	.015	.1700E+01	.9350E+00	.9735E+00	.3641E+00	.9103E+00
6	.019	.1500E+01	.9131E+00	.9643E+00	.422[illegible]+00	.8805E+00
7	.023	.1400E+01	.8972E+00	.9575E+00	.4607E+00	.8591E+00
8	.027	.1300E+01	.8758E+00	.9483E+00	.5083E+00	.8305E+00
9	.031	.1200E+01	.8450E+00	.9349E+00	.5707E+00	.7900E+00
10	.035	.1100E+01	.7954E+00	.9125E+00	.6614E+00	.7258E+00
11	.038	.1000E+01	.6339E+00	.8333E+00	.9129E+00	.5283E+00
12	.042	.1350E+01	.3152E+00	.6302E+00	.1360E+01	.1986E+00
13	.046	.1700E+01	.2278E+00	.5534E+00	.1494E+01	.1261E+00
14	.050	.2050E+01	.1790E+00	.5025E+00	.1577E+01	.8994E-01
15	.054	.2400E+01	.1474E+00	.4650E+00	.1636E+01	.6855E-01
16	.058	.2750E+01	.1253E+00	.4357E+00	.1680E+01	.5458E-01
17	.246	.1225E+02	.2400E-01	.2249E+00	.1969E+01	.5398E-02
18	.435	.1650E+02	.1752E-01	.1983E+00	.2002E+01	.3474E-02
19	.623	.1850E+02	.1553E-01	.1890E+00	.2014E+01	.2936E-02
20	.812	.1950E+02	.1470E-01	.1849E+00	.2019E+01	.2718E-02
21	1.000	.1975E+02	.1451E-01	.1839E+00	.2020E+01	.2668E-02

LOG FORM OF INITIAL CONDITIONS

I	XA	XRHO	XU	XT
1	.1435E+01	-.9697E-02	-.1972E+01	-.3879E-02
2	.1163E+01	-.1697E-01	-.1693E+01	-.6790E-02
3	.9163E+00	-.2853E-01	-.1435E+01	-.1141E-01
4	.7419E+00	-.4161E-01	-.1247E+01	-.1664E-01
5	.5306E+00	-.6716E-01	-.1010E+01	-.2687E-01
6	.4055E+00	-.9089E-01	-.8615E+00	-.3636E-01
7	.3365E+00	-.1084E+00	-.7750E+00	-.4338E-01
8	.2624E+00	-.1326E+00	-.6767E+00	-.5306E-01
9	.1823E+00	-.1684E+00	-.5609E+00	-.6736E-01
10	.9531E-01	-.2289E+00	-.4134E+00	-.9156E-01
11	0.	-.4558E+00	-.9116E-01	-.1823E+00
12	.3001E+00	-.1154E+01	.3074E+00	-.4618E+00
13	.5306E+00	-.1479E+01	.4017E+00	-.5917E+00
14	.7178E+00	-.1720E+01	.4557E+00	-.6882E+00
15	.8755E+00	-.1914E+01	.4920E+00	-.7658E+00
16	.1012E+01	-.2077E+01	.5187E+00	-.8309E+00
17	.2506E+01	-.3730E+01	.6773E+00	-.1492E+01
18	.2803E+01	-.4045E+01	.6942E+00	-.1618E+01
19	.2918E+01	-.4165E+01	.7000E+00	-.1666E+01
20	.2970E+01	-.4220E+01	.7025E+00	-.1688E+01
21	.2983E+01	-.4233E+01	.7031E+00	-.1693E+01

AZERO= 3.2191E+03 XP= 2.9020E+01 RC= 6.5955E-01 RN= 1.0364E+00 CC= 1.0613E-01 CN= 8.8746E-01

INITIAL CONDITIONS FOR NONEQUILIBRIUM VARIABLES

I	X	A	EV	XEV	EVC	EVN
1	0.000	.4200E+01	.6224E+00	-.4742E+00	.1587E+00	.4471E+00
2	.004	.3200E+01	.6193E+00	-.4792E+00	.1581E+00	.4447E+00
3	.008	.2500E+01	.6144E+00	-.4872E+00	.1571E+00	.4409E+00
4	.012	.2100E+01	.6088E+00	-.4962E+00	.1561E+00	.4365E+00
5	.015	.1700E+01	.5981E+00	-.5140E+00	.1540E+00	.4282E+00
6	.019	.1500E+01	.5883E+00	-.5305E+00	.1522E+00	.4205E+00
7	.023	.1400E+01	.5811E+00	-.5428E+00	.1508E+00	.4150E+00
8	.027	.1300E+01	.5713E+00	-.5598E+00	.1489E+00	.4073E+00
9	.031	.1200E+01	.5571E+00	-.5850E+00	.1462E+00	.3962E+00
10	.035	.1100E+01	.5336E+00	-.6281E+00	.1416E+00	.3779E+00
11	.038	.1000E+01	.5151E+00	-.6635E+00	.1255E+00	.3778E+00
12	.042	.1350E+01	.4691E+00	-.7570E+00	.8496E-01	.3777E+00
13	.046	.1700E+01	.4523E+00	-.7934E+00	.7000E-01	.3776E+00
14	.050	.2050E+01	.4415E+00	-.8177E+00	.6026E-01	.3776E+00
15	.054	.2400E+01	.4336E+00	-.8355E+00	.5320E-01	.3775E+00
16	.058	.2750E+01	.4276E+00	-.8495E+00	.4777E-01	.3774E+00
17	.246	.1225E+02	.3864E+00	-.9509E+00	.1301E-01	.3731E+00
18	.435	.1650E+02	.3786E+00	-.9712E+00	.9590E-02	.3689E+00
19	.623	.1850E+02	.3732E+00	-.9855E+00	.8483E-02	.3647E+00
20	.812	.1950E+02	.3685E+00	-.9983E+00	.8009E-02	.3604E+00
21	1.000	.1975E+02	.3642E+00	-.1010E+01	.7897E-02	.3562E+00

TIME = .2996 DELTIM = .0007 J = 440 CPTIME = 85.197 DELCPTIME = 4.930

FLUID DELTIM= .1046E-02 KINETIC DELTIM= .6834E-03

I	DTRHO	DTU	DTT	DTEVC	DTEVN	FOEVC	FOEVN
2	-2.60041E-03	-5.25837E-03	2.02844E-04	2.61634E-04	3.85917E-04	1.58535E-01	4.46634E-01
3	-1.76604E-04	-5.12565E-03	1.72453E-04	2.25089E-04	3.40486E-04	1.57472E-01	4.42273E-01
4	-4.87569E-04	-4.75374E-03	4.02272E-04	5.24086E-04	7.95471E-04	1.54788E-01	4.39471E-01
5	-2.08645E-05	-4.40958E-03	5.95996E-06	1.57751E-05	5.35782E-05	1.55096E-01	4.32547E-01
6	2.72878E-05	-3.95533E-03	-6.36571E-05	-6.04117E-05	-1.10779E-05	1.53249E-01	4.25007E-01
7	7.67335E-05	-3.15747E-03	-5.32321E-04	-6.56426E-04	-8.16656E-04	1.52127E-01	4.20439E-01
8	-2.10373E-04	-2.44161E-03	-5.24254E-04	-6.44376E-04	-7.88631E-04	1.50365E-01	4.13277E-01
9	-8.02143E-05	-1.85460E-03	-1.03856E-03	-1.31432E-03	-1.65124E-03	1.48093E-01	4.04072E-01
10	-4.39415E-04	-1.15667E-03	-8.77835E-04	-1.08376E-03	-1.63419E-03	1.43213E-01	3.84409E-01
11	-1.91929E-04	-1.14587E-03	-1.42307E-03	-1.87771E-03	-1.63304E-03	1.34703E-01	3.50513E-01
12	2.51776E-04	-1.08473E-03	-3.06090E-04	-4.68631E-04	-2.35658E-03	1.03704E-01	2.32347E-01
13	-1.48100E-03	-2.29174E-04	-2.38690E-03	-2.377[illegible]3E-03	-7.11400E-04	8.55880E-02	1.68574E-01
14	1.51923E-03	-1.26146E-03	1.22617E-03	1.14020E-03	-2.60529E-03	7.73098E-02	1.41227E-01
15	-3.25974E-03	-1.40372E-04	-4.52950E-03	-4.49561E-03	1.14744E-03	6.62327E-02	1.06861E-01
17	4.09972E-01	-2.19428E-02	5.17021E-01	5.07472E-01	-2.72532E-01	2.21595E-02	1.03311E-02
18	7.38907E-01	-9.86582E-02	9.39487E-01	2.99742E+00	-6.45679E-01	1.44070E-02	3.58989E-03
19	3.80967E-01	-9.08197E-02	6.85294E-01	3.84279E+00	-5.67233E-01	1.05288E-02	1.47294E-03
20	1.50036E-01	-6.50349E-02	4.32399E-01	3.11002E+00	-3.75885E-01	9.29281E-03	1.03639E-03

I	A	RHO	U	T	P	FV
1	.4200E+01	.9903E+00	.1417E+00	.9961E+00	.9845E+00	.6224E+00
2	.3200E+01	.9879E+00	.1861E+00	.9955E+00	.9835E+00	.6221E+00
3	.2500E+01	.9707E+00	.2445E+00	.9903E+00	.9613E+00	.6170E+00
4	.2100E+01	.9587E+00	.2940E+00	.9870E+00	.9462E+00	.6138E+00
5	.1700E+01	.9324E+00	.3733E+00	.9787E+00	.9126E+00	.6064E+00
6	.1500E+01	.9014E+00	.4373E+00	.9697E+00	.8741E+00	.5971E+00
7	.1400E+01	.8844E+00	.4800E+00	.9642E+00	.8527E+00	.5913E+00
8	.1300E+01	.8562E+00	.5293E+00	.9555E+00	.8181E+00	.5832E+00
9	.1200E+01	.8255E+00	.6033E+00	.9444E+00	.7796E+00	.5748E+00
10	.1100E+01	.7583E+00	.6981E+00	.9204E+00	.6979E+00	.5576E+00
11	.1000E+01	.6605E+00	.9215E+00	.8786E+00	.5803E+00	.5355E+00
12	.1350E+01	.3863E+00	.1262E+01	.7247E+00	.2800E+00	.4931E+00
13	.1700E+01	.2642E+00	.1441E+01	.6334E+00	.1673E+00	.4648E+00
14	.2050E+01	.2165E+00	.1511E+01	.5911E+00	.1280E+00	.4483E+00
15	.2400E+01	.1627E+00	.1619E+01	.5338E+00	.8683E-01	.4314E+00
16	.2750E+01	.1572E+00	.1624E+01	.5272E+00	.8288E-01	.4295E+00
17	.1225E+02	.2944E-01	.1861E+01	.2874E+00	.8460E-02	.3700E+00
18	.1650E+02	.1970E-01	.1932E+01	.2367E+00	.4664E-02	.3575E+00
19	.1850E+02	.1586E-01	.1980E+01	.2060E+00	.3266E-02	.3602E+00
20	.1950E+02	.1498E-01	.2000E+01	.1959E+00	.2935E-02	.3631E+00
21	.1975E+02	.1415E-01	.2020E+01	.1864E+00	.2637E-02	.3681E+00

I	A	EVC	EVN	TAUC	TAUN
1	.4200E+01	.1587E+00	.4471E+00	.8530E-03	.3640E-02
2	.3200E+01	.1586E+00	.4470E+00	.8561E-03	.3655E-02
3	.2500E+01	.1575E+00	.4430E+00	.8797E-03	.3771E-02
4	.2100E+01	.1569E+00	.4406E+00	.8964E-03	.3852E-02
5	.1700E+01	.1553E+00	.4351E+00	.9361E-03	.4048E-02
6	.1500E+01	.1534E+00	.4279E+00	.9851E-03	.4290E-02
7	.1400E+01	.1523E+00	.4233E+00	.1015E-02	.4437E-02
8	.1300E+01	.1506E+00	.4172E+00	.1066E-02	.4692E-02
9	.1200E+01	.1486E+00	.4112E+00	.1129E-02	.5016E-02
10	.1100E+01	.1444E+00	.3990E+00	.1289E-02	.5836E-02

11	.1000E+01	.1389E+00	.3840E+00	.1612E-02	.7593E-02
12	.1350E+01	.1225E+00	.3619E+00	.3903E-02	.2103E-01
13	.1700E+01	.1106E+00	.3477E+00	.7204E-02	.4298E-01
14	.2050E+01	.1037E+00	.3391E+00	.9852E-02	.6872E-01
15	.2400E+01	.9633E-01	.3308E+00	.1538E-01	.1058E+00
16	.2750E+01	.9537E-01	.3304E+00	.1621E-01	.1198E+00
17	.1225E+02	.5833E-01	.3110E+00	.1492E+00	.2390E+01
18	.1650E+02	.3675E-01	.3204E+00	.2400E+00	.5326E+01
19	.1850E+02	.1993E-01	.3402E+00	.3563E+00	.8679E+01
20	.1950E+02	.1377E-01	.3492E+00	.4016E+00	.1010E+02
21	.1975E+02	.9514E-02	.3585E+00	.4523E+00	.1173E+02

TIME = 3.4100 DELTIM = .0130 J = 660 CPTIME = 63.394 DELCPTIME = 1.012

FLUID DELTIM= .5169E-01 KINETIC DELTIM= .1296E-01

I	DTRHO	DTU	DTT	DTEVC	DTEVN	FDEVC	FDEVN
17	1.18369E-03	-2.00990E-04	9.46459E-04	1.20492E-05	-7.58472E-04	2.25270E-02	1.07515E-02
18	1.95821E-03	-2.67309E-04	1.08309E-03	-6.93637E-04	-1.44902E-03	1.87175E-02	6.80529E-03
19	-7.05415E-04	5.13930E-04	-1.00786E-03	-1.71471E-03	2.16769E-04	1.75067E-02	5.74456E-03
20	-1.90545E-03	6.89806E-04	-1.67147E-03	-1.60179E-03	1.32250E-03	1.75523E-02	5.78277E-03

I	A	RHO	U	T	P	FV
16	.2750E+01	.1572E+00	.1624E+01	.5273E+00	.8290E-01	.4295E+00
17	.1225E+02	.2964E-01	.1861E+01	.2896E+00	.8585E-02	.3695E+00
18	.1650E+02	.2177E-01	.1901E+01	.2649E+00	.5747E-02	.3454E+00
19	.1850E+02	.1899E-01	.1918E+01	.2568E+00	.4877E-02	.3300E+00
20	.1950E+02	.1831E-01	.1921E+01	.2571E+00	.4707E-02	.3194E+00
21	.1975E+02	.1765E-01	.1924E+01	.2574E+00	.4542E-02	.3095E+00

I	A	EVC	EVN	TAUC	TAUN
16	.2750E+01	.9532E-01	.3304E+00	.1620E-01	.1127E+00
17	.1225E+02	.5694E-01	.3123E+00	.1479E+00	.2333E+01
18	.1650E+02	.4176E-01	.3032E+00	.2029E+00	.3828E+01
19	.1850E+02	.3291E-01	.2967E+00	.2322E+00	.4679E+01
20	.1950E+02	.2815E-01	.2908E+00	.2409E+00	.4842E+01
21	.1975E+02	.2408E-01	.2850E+00	.2499E+00	.5010E+01

VIBRATIONAL TEMPERATURES AND POPULATIONS

I	A	T1(K)	T2(K)	T(K)	N	N001	N100	(N001-N100)/N	X(CM)	EVMAX	GG
1	4.2000	2390.7	2390.7	2390.7	4.242E+18	4.730E+16	8.676E+16	-9.301E-03	0.00000	-2.7914E-01	-1.2292E-02
2	3.2000	2389.6	2390.3	2389.3	4.231E+18	4.723E+16	8.661E+16	-9.307E-03	.02298	-2.7879E-01	-1.2310E-02
3	2.5000	2377.4	2379.0	2376.8	4.158E+18	4.674E+16	8.593E+16	-9.426E-03	.04595	-2.7690E-01	-1.2559E-02
4	2.1000	2369.7	2372.1	2368.8	4.106E+18	4.637E+16	8.538E+16	-9.500E-03	.06893	-2.7563E-01	-1.2718E-02
5	1.7000	2350.7	2356.1	2348.9	3.994E+18	4.562E+16	8.426E+16	-9.676E-03	.09191	-2.7232E-01	-1.3108E-02
6	1.5000	2329.1	2335.5	2327.2	3.861E+18	4.464E+16	8.287E+16	-9.902E-03	.11488	-2.6916E-01	-1.3590E-02
7	1.4000	2316.0	2322.4	2314.0	3.788E+18	4.411E+16	8.217E+16	-1.005E-02	.13786	-2.6732E-01	-1.3899E-02
8	1.3000	2296.4	2304.7	2293.3	3.667E+18	4.319E+16	8.079E+16	-1.025E-02	.16084	-2.6401E-01	-1.4361E-02
9	1.2000	2273.0	2287.2	2266.5	3.536E+18	4.228E+16	7.928E+16	-1.047E-02	.18381	-2.5900E-01	-1.4903E-02
10	1.1000	2222.6	2251.7	2209.0	3.248E+18	4.016E+16	7.562E+16	-1.092E-02	.20679	-2.4794E-01	-1.6114E-02
11	1.0000	2158.3	2207.7	2108.5	2.829E+18	3.654E+16	6.908E+16	-1.150E-02	.22977	-2.2613E-01	-1.8113E-02
12	1.3500	1963.8	2142.4	1739.3	1.655E+18	2.536E+16	4.598E+16	-1.246E-02	.25275	-1.3430E-01	-2.5579E-02
13	1.7000	1822.2	2100.1	1520.0	1.131E+18	1.983E+16	3.439E+16	-1.286E-02	.27572	-8.2097E-02	-3.1700E-02
14	2.0500	1738.5	2074.4	1418.6	9.272E+17	1.763E+16	2.965E+16	-1.296E-02	.29870	-5.5452E-02	-3.5028E-02
15	2.4000	1650.4	2049.2	1281.1	6.967E+17	1.450E+16	2.344E+16	-1.283E-02	.32168	-2.8326E-02	-3.9702E-02
16	2.7500	1638.3	2048.0	1265.6	6.734E+17	1.422E+16	2.280E+16	-1.275E-02	.34465	-2.4645E-02	-4.0088E-02
17	12.2500	1163.8	1992.9	695.0	1.270E+17	4.857E+15	5.044E+15	-1.469E-03	1.47052	8.5526E-02	-9.7580E-03
18	16.5000	966.2	1964.6	635.7	9.325E+16	4.697E+15	3.561E+15	1.219E-02	2.59639	9.4914E-02	8.9649E-02
19	18.5000	845.6	1944.6	616.2	8.135E+16	4.876E+15	2.829E+15	2.516E-02	3.72226	9.5049E-02	1.9172E-01
20	19.5000	778.1	1926.3	617.0	7.842E+16	5.164E+15	2.499E+15	3.398E-02	4.84813	9.2540E-02	2.5854E-01
21	19.7500	718.4	1908.3	617.7	7.559E+16	5.405E+15	2.164E+15	4.287E-02	5.97400	9.0079E-02	3.2574E-01

PROGRAM LISTING

```
      PROGRAM GDL (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      DIMENSION RHO(95),T(95),U(95),P(95),XI(95),D(95),RHON(95),UN(95),
     1TN(95),XRHO(95),XU(95),XT(95),XA(95),EV(95),XEV(95),EVN(95),
     2XEVC(95),XEVN(95),TAUN(95),EVC(95),TAUNH(95),TAUCH(95),EVCN(95),
     3EVNN(95),TAUC(95),ZDTRHO(95),ZDTU(95),ZDTT(95),RHONN(95),UNN(95),
     4TNN(95),ZDTEVC(95),ZDTEVN(95),EVCNN(95),EVNNN(95),E1(95),E2(95),
     5DTTSTR(95),ROS(95),ADD(95),A(95)
      DATA GAMMA/1.4/, UR/49721.7/, NCASE/0/
      DATA CVT,VTC,VTCC,VTCCC/6100.,3474.,1728.,6084./
      DATA VTH,VTHH,VTHHH/9900.,4290.,10200./
      DATA XMN,XMC,XMH/28.,44.,18./
      DATA EV,EVC,EVN,XEV,XEVC,XEVN/285*(1.),285*(0.)/
      DATA TAUC,TAUN/190*(0.)/, J00/100/, JMODO/50/


  100 READ (5,500) NN,NB,NT
      IF (NN .EQ. 0) STOP
      READ (5,510) XSUB,XSUP
      IF (NN .GT. 0) READ (5,510) (A(I),I=1,NN)
      NN=IABS(NN)
      READ (5,520) FMC,FMH,ST,SP,JFIN,JTEM
      STTTK=ST
      JFINO=2001
      IF (NT .EQ. 0) JFINO=501
      IF (JFIN .EQ. 0) JFIN=JFINO
      IF (JTEM .EQ. 0) JTEM=1000

  105 CPTIMP=SECOND(CPTIM)
      WRITE (6,610) CPTIM
      EV1=EV(1)
      EVC1=EVC(1)
      EVN1=EVN(1)
      FMN=1.-FMC-FMH
      ST=STTTK*1.8
      WRITE(6,611) STTTK,SP,FMC,FMN,FMH,NN,NB,NT,XSUB,XSUP
      IF (NT .NE. 0) GO TO 110
      J00=1
      JMODO=10
      XL=XSUP
      DXONE=1./FLOAT(NN-1)
      DXTWO=DXONE
      READ (5,510) A(1),RHO(1),U(1),T(1),EVC(1),EVN(1)
      GO TO 115
  110  XL=XSUB+XSUP
       DXONE=XSUB/XL/FLOAT(NT-1)
       DXTWO=(XSUP-XSUB*FLOAT(NB-NT)/FLOAT(NT-1))/XL/FLOAT(NN-NB)
  115 XL=XL/12.
      DELTX=DXONE
      AZZ=DXONE/(DXONE+DXTWO)
      ND=NN-1
      ZBLLL=-20.4*(600.0)**(-1.0/3.0)+0.642
      NCASE=NCASE+1
      IF (NCASE .GT. 1) GO TO 140


C     INITIAL CONDITIONS
      X=-DXONE
      DO 130 I=1,NN
      IF(I.LE.NB) X=X+DXONE
      IF(I.GT.NB) X=X+DXTWO
      XI(I)=X
```

```
      DTTSTR(I)=0.0
      IF (NT .NE. 0) GO TO 120
      RHO(I)=RHO(1)
      U(I)=U(1)
      T(I)=T(1)
      EVC(I)=EVC(1)
      EVN(I)=EVN(1)
      GO TO 130
  120 IF(I.EQ.1) V=.1
      IF(I.EQ.NT) V=1.
      IF(I.EQ.NT+1) V=1.5
       IF(I.EQ.NB+1)V=3.5
      C1=1.728*A(I)
  125 C2=(1.+0.2*V**2)
      C3=(C1*V-C2**3)/(C1-1.2*V*C2**2)
      V=V-C3
      IF(.000001.LT.ABS(C3)) GO TO 125
      U(I)=V/SQRT(C2)
      T(I)=1./C2
      RHO(I)=T(I)**2.5
  130 CONTINUE
  140 DO 150 I=1,NN
      P(I)=T(I)*RHO(I)
      XRHO(I)=ALOG(RHO(I))
      XU(I)=ALOG(U(I))
      XT(I)=ALOG(T(I))
      XA(I)=ALOG(A(I))
  150 CONTINUE
      RHON(1)=XRHO(1)
      TN(1)=XT(1)
      IF (NT .EQ. 0) UN(1)=XU(1)
      WRITE(6,612)
      WRITE(6,613) (I,XI(I),A(I),RHO(I),T(I),U(I),P(I),I=1,NN)
      WRITE(6,614)
      WRITE(6,615) (I,XA(I),XRHO(I),XU(I),XT(I),I=1,NN)

      XM=FMC*XMC+FMN*XMN+FMH*XMH
      RC=XM/XMC
      RN=XM/XMN
      RH=XM/XMH
      CC=FMC/RC
      CN=FMN/RN
      CH=FMH/RH
      AZERO=SQRT(GAMMA*UR*ST/XM)
      TRES=XL/AZERO
      SR=UR/XM
      SRST=SR*ST
      WRITE(6,616) AZERO,XM,RC,RN,CC,CN
      J=1
      J02=50
      IF (J00 .NE. 1) J02=200
      JMOD=JMODO
      NWG1=1
      NWG2=2
      NWG3=NB-1
      NWG4=0
      NWG5=1
      IF (NT .EQ. 0) NWG3=NN/2
      TIME=0.0
      CPTIMP=SECOND(CPTIM)
      IF (J00 .EQ. 1) GO TO 160
      NWG5=NN
```

```
160 WRITE(6,617)
    DO 170 I=1,NWG5
    X=XI(I)
    TD=ST*T(I)
    IF (NT .EQ. 0) GO TO 165
    YA=CVT/TD
    YB=EXP(YA)
    YC=YB-1.0
    YACCC=VTCCC/TD
    YBCCC=EXP(YACCC)
    YCCCC=YBCCC-1.0
    EVN(I)=(YACCC*CC*RC/YCCCC+YA*CN*RN/YC)*T(I)
    IF (I.GT.NT-1) EVN(I)=EVN(NT-1)-0.0225*(X-0.03253)
    YAC=VTC/TD
    YBC=EXP(YAC)
    YCC=YBC-1.0
    YACC=VTCC/TD
    YBCC=EXP(YACC)
    YCCC=YBCC-1.0
    EVC(I)=CC*RC*(YAC/YCC+2.0*YACC/YCCC)*T(I)
165 EV(I)=EVC(I)+EVN(I)
    YAH=VTH/TD
    YBH=EXP(YAH)
    YCH=YBH-1.0
    YAHH=VTHH/TD
    YBHH=EXP(YAHH)
    YCHH=YBHH-1.0
    YAHHH=VTHHH/TD
    YBHHH=EXP(YAHHH)
    YCHHH=YBHHH-1.0
    EVH=(YAH/YCH+2.0*YAHH/YCHH+YAHHH/YCHHH)*RH*T(I)*CH
    EV(I)=EV(I)+EVH
    XEV(I)=ALOG(EV(I))
    XEVC(I)=ALOG(EVC(I))
    XEVN(I)=ALOG(EVN(I))
    WRITE(6,613) I,X,A(I),EV(I),XEV(I),EVC(I),EVN(I)
170 CONTINUE
    EVCN(1)=XEVC(1)
    EVNN(1)=XEVN(1)
    IF (J00 .GT. 1) GO TO 200

    EVR=EV(1)/EV1
    EVCR=EVC(1)/EVC1
    EVNR=EVN(1)/EVN1
    XEVR=ALOG(EVR)
    XEVCR=ALOG(EVCR)
    XEVNR=ALOG(EVNR)
    DO 180 I=2,NN
    EV(I)=EV(I)*EVR
    IF (NCASE .GT. 1) EVC(I)=EVC(I)*EVCR
    IF (NCASE .GT. 1) EVN(I)=EVN(I)*EVNR
    XEV(I)=XEV(I)+XEVR
    XEVC(I)=XEVC(I)+XEVCR
    XEVN(I)=XEVN(I)+XEVNR
    X=XI(I)
    WRITE(6,613) I,X,A(I),EV(I),XEV(I),EVC(I),EVN(I)
180 CONTINUE
```

```
C*****CALCULATION OF DELTIM
  200 DO 205 I=NWG1,NN
      BA=SQRT(T(I))
      D(I)=DELTX/(U(I)+BA)
      IF (I .EQ. NWG1) DELTXX=D(I)
      IF(D(I).LT.DELTXX) DELTXX=D(I)
  205 CONTINUE
      DELTYY=AMIN1(TAUC(NWG1),TAUN(NWG1))
      DELTXX=DELTXX/1.5
      DELTYY=0.8*DELTYY
      DELTIM=AMIN1(DELTXX,DELTYY)
      TIME=TIME+DELTIM
      TEST=MOD(J,JMOD)
      IF (TEST .GT. 0.01) GO TO 210
      CPTIMP=CPTIM
      DCPTIM=SECOND(CPTIM)-CPTIMP
      WRITE (6,620) TIME,DELTIM,J,CPTIM,DCPTIM
      WRITE(6,621) DELTXX,DELTYY
      WRITE (6,622)



C*****CALCULATION OF RELAXATION TIMES
  210 DO 220 I=NWG1,NN
      TAUCH(I)=TAUC(I)
      TAUNH(I)=TAUN(I)
      TD=T(I)*STTTK
      ZTD=TD**(-1.0/3.0)
      PD=P(I)*SP
      PDC=PD/1.0E-6
      ZAL=17.8*ZTD-1.808
      ZBL=-20.4*ZTD+0.642
      IF(TD.LT.600.0) ZBL=ZBLLL
      ZAP=10.0**ZAL
      ZBP=10.0**ZBL
      ZCP=4.0*ZAP
      ZA=ZAP/PDC
      ZB=ZBP/PDC
      ZC=ZCP/PDC
      ZD=FMC/ZA+FMN/ZC+FMH/ZB
      TAUC(I)=1.0/(ZD*TRES)
      ZAP=(1.3E5)*ZTD**4.9
      ZBP=0.27*ZAP
      ZCP=5.5E-2
      ZA=ZAP/PDC
      ZB=ZBP/PDC
      ZC=ZCP/PDC
      ZD=FMC/ZB+FMN/ZA+FMH/ZC
      ZEL=93.0*ZTD-4.61
      ZGL=15.4*ZTD+0.722
      ZEP=10.0**ZEL
      ZGP=10.0**ZGL
      ZE=ZEP/PDC
      ZG=ZGP/PDC
      ZH=(FMC+FMN)/ZE+FMH/ZG
      ZI=FMC*ZD+FMN*ZH
      TAUN(I)=(FMC+FMN)/(ZI*TRES)
  220 CONTINUE
```

```
      DO 230 I=NWG2,ND
      IF(I.EQ.NB) GO TO 230
      DELX=DXTWO
      IF (I .LT. NB) DELX=DXONE
C*****CALCULATION OF X DERIVATIVES (FIRST STAGE IN MACCORMACKS METHOD)
      DRHO=(XRHO(I+1)-XRHO(I))/DELX
      DU=(XU(I+1)-XU(I))/DELX
      DT=(XT(I+1)-XT(I))/DELX
      DA=(XA(I+1)-XA(I))/DELX
      IF(I.EQ.NT) DA=0.0
      DEVC=(XEVC(I+1)-XEVC(I))/DELX
      DEVN=(XEVN(I+1)-XEVN(I))/DELX
      DEV=(XEV(I+1)-XEV(I))/DELX
C*****CALCULATION OF TIME DERIVATIVES (FIRST STAGE)
      DTRHO=-U(I)*(DA+DU+DRHO)
      DTU=-(DT+DRHO)*T(I)/(GAMMA*U(I))-U(I)*DU
      TD=ST*T(I)
      PD=SP*P(I)
      YA=CVT/TD
      YB=EXP(YA)
      YC=YB-1.0
      EQEV=YA*T(I)/YC
      YACCC=VTCCC/TD
      YBCCC=EXP(YACCC)
      YCCCC=YBCCC-1.0
      E1(I)=CC*RC*YACCC/YCCCC*T(I)+EQEV*CN*RN
      YAC=VTC/TD
      YBC=EXP(YAC)
      YCC=YBC-1.0
      YACC=VTCC/TD
      YBCC=EXP(YACC)
      YCCC=YBCC-1.0
      E2(I)=CC*RC*(YAC/YCC+2.0*YACC/YCCC)*T(I)
      EQEVN=E1(I)
      EQEVC=E2(I)
      DTEVC=(EQEVC/EVC(I)-1.0)/TAUC(I)-U(I)*DEVC
      DTEVN=(EQEVN/EVN(I)-1.0)/TAUN(I)-U(I)*DEVN
      DTEV=(EVC(I)*DTEVC+EVN(I)*DTEVN)/EV(I)
      YAH=VTH/TD
      YBH=EXP(YAH)
      YCH=YBH-1.0
      YAHH=VTHH/TD
      YBHH=EXP(YAHH)
      YCHH=YBHH-1.0
      YAHHH=VTHHH/TD
      YBHHH=EXP(YAHHH)
      YCHHH=YBHHH-1.0
      ADD(I)=RH*(YAH*YAH*YBH/YCH/YCH+2.0*YAHH*YAHH*YBHH/YCHH/YCHH
     * +YAHHH*YAHHH*YBHHH/YCHHH/YCHHH)*CH*DTTSTR(I)
      ROS(I)=0.4/(CC*RC+CN*RN+1.2*CH*RH)
      DTT=-ROS(I)*(U(I)*DU+U(I)*DA+EV(I)*DTEV/T(I)+U(I)*EV(I)*DEV/T(I)+
     1ADD(I))-U(I)*DT
C*****CALCULATION OF VARIABLES AT NEW TIME (FIRST STAGE)
      RHON(I)=XRHO(I)+DTRHO*DELTIM
      UN(I)=XU(I)+DTU*DELTIM
      EVCN(I)=XEVC(I)+DTEVC*DELTIM
      EVNN(I)=XEVN(I)+DTEVN*DELTIM
      TN(I)=XT(I)+DTT*DELTIM
      ZDTRHO(I)=DTRHO
      ZDTU(I)=DTU
      ZDTT(I)=DTT
```

```
      ZDTEVC(I)=DTEVC
      ZDTEVN(I)=DTEVN
  230 CONTINUE
      UN(1)=XU(1)
      IF (NT .EQ. 0) GO TO 240
      UN(1)=2.0*UN(2)-UN(3)
      I=NB
      RHON(I)=RHON(I-1)-(RHON(I-1)-RHON(I+1))*AZZ
      UN(I)=UN(I-1)-(UN(I-1)-UN(I+1))*AZZ
      TN(I)=TN(I-1)-(TN(I-1)-TN(I+1))*AZZ
      EVCN(I)=EVCN(I-1)-(EVCN(I-1)-EVCN(I+1))*AZZ
      EVNN(I)=EVNN(I-1)-(EVNN(I-1)-EVNN(I+1))*AZZ

  240 DO 250  I=NWG2,ND
      IF(I.EQ.NB) GO TO 250
      DELX=DXTWO
      IF (I .LT. NB) DELX=DXONE
C*****CALCULATION OF X DERIVATIVES (SECOND STAGE)
      DRHO=(RHON(I)-RHON(I-1))/DELX
      DU=(UN(I)-UN(I-1))/DELX
      DT=(TN(I)-TN(I-1))/DELX
      DA=(XA(I)-XA(I-1))/DELX
      IF(I.EQ.NT) DA=0.0
      DEVC=(EVCN(I)-EVCN(I-1))/DELX
      DEVN=(EVNN(I)-EVNN(I-1))/DELX
      DEV=(EVC(I)*DEVC+EVN(I)*DEVN)/EV(I)
      TD=ST*T(I)
      YAH=VTH/TD
      YBH=EXP(YAH)
      YCH=YBH-1.0
      YAHH=VTHH/TD
      YBHH=EXP(YAHH)
      YCHH=YBHH-1.0
      YAHHH=VTHHH/TD
      YBHHH=EXP(YAHHH)
      YCHHH=YBHHH-1.0
      ADDEV =RH*(YAH*YAH*YBH/YCH/YCH+2.0*YAHH*YAHH*YBHH/YCHH/YCHH
     * +YAHHH*YAHHH*YBHHH/YCHHH/YCHHH)*CH*DT*T(I)/EV(I)
      DEV=DEV+ADDEV
C*****CALCULATION OF NEW TIME DERIVATIVES (SECOND STAGE)
      EQEVN=E1(I)
      EQEVC=E2(I)
      DTRHO=-U(I)*(DA+DU+DRHO)
      DTU=-(DT+DRHO)*T(I)/(GAMMA*U(I))-U(I)*DU
      DTEVC=(EQEVC/EVC(I)-1.0)/TAUC(I)-U(I)*DEVC
      DTEVN=(EQEVN/EVN(I)-1.0)/TAUN(I)-U(I)*DEVN
      DTEV=(EVC(I)*DTEVC+EVN(I)*DTEVN)/EV(I)
      DTT=-ROS(I)*(U(I)*DU+U(I)*DA+EV(I)*DTEV/T(I)+U(I)*EV(I)*DEV/T(I)+
     1ADD(I))-U(I)*DT
C*****CALCULATION OF AVERAGE TIME DERIVATIVES
      DTRHO=0.5*(DTRHO+ZDTRHO(I))
      DTU=0.5*(DTU+ZDTU(I))
      DTEVC=0.5*(DTEVC+ZDTEVC(I))
      DTEVN=0.5*(DTEVN+ZDTEVN(I))
      DTT=0.5*(DTT+ZDTT(I))
      DTTSTR(I)=DTT
      IF (I .NE. NWG3) GO TO 245
      ABSDTC=ABS(DTEVC)
      ABSDTN=ABS(DTEVN)
      ABSDT=AMAX1(ABSDTC,ABSDTN)
C*****CALCULATION OF FINAL VALUES AT NEW TIME
```

```
  245 RHONN(I)=XRHO(I)+DTRHO*DELTIM
      UNN(I)=XU(I)+DTU*DELTIM
      EVCNN(I)=XEVC(I)+DTEVC*DELTIM
      EVNNN(I)=XEVN(I)+DTEVN*DELTIM
      TNN(I)=XT(I)+DTT*DELTIM
      IF(TEST.GT.0.01) GO TO 250
      WRITE(6,623) I,DTRHO,DTU,DTT,DTEVC,DTEVN,EQEVC,EQEVN
  250 CONTINUE

C*****EXTRAPOLATION TO END POINTS
      I=NN
      RHONN(I)=2.0*RHONN(I-1)-RHONN(I-2)
      UNN(I)=2.0*UNN(I-1)-UNN(I-2)
      TNN(I)=2.0*TNN(I-1)-TNN(I-2)
      EVCNN(I)=2.0*EVCNN(I-1)-EVCNN(I-2)
      EVNNN(I)=2.0*EVNNN(I-1)-EVNNN(I-2)
      IF (NT .EQ. 0) GO TO 255
      I=1
      UNN(I)=2.0*UNN(I+1)-UNN(I+2)
      XU(I)=UNN(I)
      U(I)=EXP(XU(I))
      I=NB
      RHONN(I)=RHONN(I-1)-(RHONN(I-1)-RHONN(I+1))*AZZ
      UNN(I)=UNN(I-1)-(UNN(I-1)-UNN(I+1))*AZZ
      TNN(I)=TNN(I-1)-(TNN(I-1)-TNN(I+1))*AZZ
      EVCNN(I)=EVCNN(I-1)-(EVCNN(I-1)-EVCNN(I+1))*AZZ
      EVNNN(I)=EVNNN(I-1)-(EVNNN(I-1)-EVNNN(I+1))*AZZ

  255 DO 260 I=NWG2,NN
      XRHO(I)=RHONN(I)
      XU(I)=UNN(I)
      XT(I)=TNN(I)
      XEVC(I)=EVCNN(I)
      XEVN(I)=EVNNN(I)
      RHO(I)=EXP(XRHO(I))
      U(I)=EXP(XU(I))
      EVC(I)=EXP(XEVC(I))
      EVN(I)=EXP(XEVN(I))
      EV(I)=EVC(I)+EVN(I)
      T(I)=EXP(XT(I))
      P(I)=RHO(I)*T(I)
      TD=ST*T(I)
      YAH=VTH/TD
      YBH=EXP(YAH)
      YCH=YBH-1.0
      YAHH=VTHH/TD
      YBHH=EXP(YAHH)
      YCHH=YBHH-1.0
      YAHHH=VTHHH/TD
      YBHHH=EXP(YAHHH)
      YCHHH=YBHHH-1.0
      EVH=(YAH/YCH+2.0*YAHH/YCHH+YAHHH/YCHHH)*RH*T(I)*CH
      EV(I)=EV(I)+EVH
      XEV(I)=ALOG(EV(I))
  260 CONTINUE

      IF(TEST.GT.0.01) GO TO 270
      WRITE(6,624)
      WRITE(6,625)(I,A(I),RHO(I),U(I),T(I),P(I),EV(I),I=NWG1,NN)
      WRITE(6,626)
      WRITE(6,627) (I,A(I),EVC(I),EVN(I),TAUC(I),TAUN(I),I=NWG1,NN)
```

```
  270 J=J+1
      IF (J .LT. J02) GO TO 200
C*****TEST FOR CONVERGENCE OF SOLUTION
      IF (NWG4 .EQ. 3) GO TO 280
      IF (MOD(J,10) .NE. 0) GO TO 295
      IF (ABSDT .GT. .005) GO TO 290
      NWG4=NWG4+1
      IF (NWG4 .EQ. 3) JMOD=J
      GO TO 200
C*****DROP UPSTREAM POINTS NOW AT STEADY STATE
  280 IF (NWG1 .EQ. NWG2) J=JFIN
      JMOD=JMODO
      NWG1=NWG3+1
      NWG2=NWG1
      NWG3=NN-1
      NWG4=1
      DELTX=DXTWO
      IF (ABSDT .LE. .005) GO TO 295
  290 NWG4=0
  295 IF(J.LT.JFIN) GO TO 200

C*****CALCULATION OF VIBRATIONAL TEMPERATURES AND POPULATIONS
  300 WRITE(6,630)
      TT=T(1)*ST
      TTN=TT
      DO 340 I=1,NN
      TEVC=EVC(I)/CC*ST/RC
      JXX=1
  310 JXX=JXX+1
      IF(JXX.GT.JTEM)GO TO 350
      YAC=VTC/TT
      YBC=EXP(YAC)
      YCC=YBC-1.0
      YACC=VTCC/TT
      YBCC=EXP(YACC)
      YCCC=YBCC-1.0
      YCC=VTC/YCC
      YCCC=VTCC/YCCC
      ZTEST=TT*(TEVC-YCC-2.*YCCC)/(YBC*YCC**2+2.*YBCC*YCCC**2)
      TT=TT*(1.+ZTEST)
      AZTEST=ABS(ZTEST)
      IF(AZTEST.GT.1.0E-8) GO TO 310
      TEVN=CN*RN
      TEVC=CC*RC
      JYY=1
      EVN111=EVN(I)*ST
  320 JYY=JYY+1
      IF(JYY.GT.JTEM) GO TO 350
      YACCC=VTCCC/TTN
      YBCCC=EXP(YACCC)
      YCCCC=YBCCC-1.0
      YA=CVT/TTN
      YB=EXP(YA)
      YC=YB-1.0
      YCCCC=VTCCC/YCCCC
      YC=CVT/YC
      ZTEST=TTN*(EVN111-TEVC*YCCCC-TEVN*YC)
      ZTEST=ZTEST/(TEVC*YBCCC*YCCCC**2+TEVN*YB*YC**2)
      TTN=TTN*(1.+ZTEST)
      AZTEST=ABS(ZTEST)
      IF(AZTEST.GT.1.0E-8) GO TO 320
```

```
 330 TK=T(I)*STTTK
     PK=P(I)*SP*1.01325E6
     ZNK=PK*FMC/(TK*1.38E-16)
     YAC=VTC/TT
     YACC=VTCC/TT
     YACCC=VTCCC/TTN
     ZA=EXP(-YAC)
     ZAA=1.0-ZA
     ZB=EXP(-YACC)
     ZBB=(1.0-ZB)**2
     ZC=EXP(-YACCC)
     ZCC=1.0-ZC
     ZLL=ZNK*ZA*ZAA*ZBB*ZCC
     ZUL=ZNK*ZC*ZAA*ZBB*ZCC
     ZIN=(ZUL-ZLL)/ZNK
     W=10.6E-6
     WT=5.38
     WC=W**2/(4.0*3.1415*WT)
     WC1=8.0*8.317E3/3.1415
     WCO2=SQRT(WC1*TK*(XMC+XMC)/(XMC*XMC))
     WN2=SQRT(WC1*TK*(XMC+XMN)/(XMC*XMN))
     WH=SQRT(WC1*TK*(XMC+XMH)/(XMC*XMH))
     WV=FMC*(1.3E-18)*WCO2+FMN*(0.87E-18)*WN2+FMH*(0.38E-18)*WH
     WJ=45.6/TK*EXP(-234.0/TK)
     WG=WC*ZIN*FMC*WJ/WV
     X=XI(I)
     Z=X*XL*12.0*2.54
     TTKEL=TT/1.8
     TTNKEL=TTN/1.8
C*****MAX AVAILABLE ENERGY
     TTZ=1.76*TK*1.8
     YACCC=VTCCC/TTZ
     YBCCC=EXP(YACCC)
     YCCCC=YBCCC-1.0
     YA=CVT/TTZ
     YB=EXP(YA)
     YC=YB-1.0
     ZEVN=(CC*RC*YACCC/YCCCC+CN*RN*YA/YC)*TTZ/ST
     EVMAX=(EVN(I)-ZEVN)*0.409
     WRITE(6,631) I,A(I),TTKEL,TTNKEL,TK,ZNK,ZUL,ZLL,ZIN,Z,EVMAX,WG
 340 CONTINUE


     J00=1
     GO TO 100
 350 WRITE(6,632)
     STOP


 500 FORMAT(4I5)
 510 FORMAT(7F10.5)
 520 FORMAT (4F10.5,2I5)


 610 FORMAT (1H1 74X,8HCPTIME = F8.3)
 611 FORMAT(14H0    INPUT DATA //2X,16H TEMPERATURE(K)=,E14.8,4X,
    * 15H PRESSURE(ATM)=,E14.8//2X,5H CO2=,E14.8,4X,4H N2=,E14.8,4X,
    * 5H H2O=,E14.8//3X,3HNN=I3,5X,3HNB=I3,5X,3HNT=I3,5X,5HXSUB=E14.8
    *,5X,5HXSUP=E14.8//)
 612 FORMAT(1H0,18HINITIAL CONDITIONS/4X,1HI,12X,1HX,12X,1HA,13X,
    *3HRHO,13X,1HT,14X,1HU,14X,1HP/)
```

(CONTINUED)

```
613 FORMAT(2X,I3,5X,F10.3,5E15.4)                                       GDL05510
614 FORMAT(1H0/2X,30HLOG FORM OF  INITIAL CONDITIONS/4X,1HI,9X,2HXA,    GDL05520
   *12X,4HXRHO,12X,2HXU,13X,2HXT/)                                      GDL05530
615 FORMAT(2X,I3,4E15.4)                                                GDL05540
616 FORMAT (1H0,6HAZERO= 1PE11.4,5X,3HXM= E11.4,5X,3HRC= E11.4,5X,      GDL05550
   *3HRN= E11.4,5X,3HCC= E11.4,5X,3HCN= E11.4)                          GDL05560
617 FORMAT(1H0,47HINITIAL CONDITIONS FOR NONEQUILIBRIUM VARIABLES/4X,   GDL05570
   *1HI,12X,1HX,12X,1HA,13X,2HEV,12X,3HXEV,12X,3HEVC,12X,3HEVN/)        GDL05580
620 FORMAT (9H1  TIME = F7.4, 7X, 8HDELTIM = F6.4, 7X,3HJ =I4,23X,      GDL05590
   * 8HCPTIME = F8.3, 6X, 11HDELCPTIME = F6.3//)                        GDL05600
621 FORMAT(27X,13HFLUID DELTIM=,E10.4,15X,15HKINETIC DELTIM=,E10.4)     GDL05610
622 FORMAT (1H0/9X,1HI 7X,5HDTRHO,11X,3HDTU,12X,3HDTT,11X,5HDTEVC,      GDL05620
   *   10X,5HDTEVN,10X,5HEQEVC,10X,5HEQEVN  )                           GDL05630
623 FORMAT (I10,1P7E15.5)                                               GDL05640
624 FORMAT(1H0/9X,1HI,15X,1HA,13X, 3HRHO,13X,1HU,14X,1HT,14X,1HP,       GDL05650
   * 13X,2HEV)                                                          GDL05660
625 FORMAT(7X,I3,5X,6E15.4)                                             GDL05670
626 FORMAT(1H0/9X,1HI,15X,1HA,13X,3HEVC,12X,3HEVN,12X,4HTAUC,11X,       GDL05680
   * 4HTAUN)                                                            GDL05690
627 FORMAT(7X,I3,5X,5E15.4)                                             GDL05700
630 FORMAT (43H1  VIBRATIONAL TEMPERATURES AND POPULATIONS              GDL05710
   *//120H0  I      A       T1(K)    T2(K)    T(K)          N       N001 GDL05720
   *      N100    (N001-N100)/N   X(CM)        EVMAX          G0       ) GDL05730
631 FORMAT (1H0 I3,F9.4,3F8.1,2X,1P3E11.3,E12.3,0PF11.5,1P2E13.4)       GDL05740
632 FORMAT(45H ITERATION FOR VIBRATIONAL TEMPERATURES FAILS)            GDL05750
    END                                                                 GDL05760
```