LA-6517-MS

Informal Report

UC-34

Reporting Date: September 1976

Issued: September 1976

GRASER: A Kinetics Code for Study of Neutron-Pumped Gamma-Ray Laser Systems

by

John W. Pettit*
George C. Baldwin**

*Summer Graduate Student.
**Long-Term Visiting Staff Member.
Both from Rensselaer Polytechnic Institute, Troy, NY 12181.

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

UNITED STATES
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy $4.00 Microfiche $2.25

GRASER: A KINETICS CODE FOR STUDY OF NEUTRON-PUMPED

GAMMA-RAY LASER SYSTEMS

by

John W. Pettit and George C. Baldwin

ABSTRACT

Description, listing, sample plots and operating instructions are given of a computer code for the CDC-6600 which calculates the time-dependence of populations of isomeric nuclear states and the time- and space-dependences of the vector potential of a plane wave of recoilless gamma radiation initiated by spontaneous emission and amplified by stimulated emission in a medium of specified properties. The code is based upon a continuous-function approximation of the neutron flux arising from a fast source burst of user-specified intensity in an infinite hydrogenous moderator. A variety of properties of the nuclear transition and of the medium may be specified by the user, including single-resonance and 1/v capture parameters of the parent isotope; statistical weight factors, isomer inversion ratio and lifetimes of the graser levels; and degree of line broadening, Debye-Waller factor and nonresonant absorption coefficient of the medium.

I. INTRODUCTION

GRASER is a computer code that solves the Kinetics equations of the type I, directly pumped gamma-ray lasers that are described in the documents, "Kinetics of Stimulated Emission in Neutron-Pumped Nuclear Laser Systems," by G. C. Baldwin and "Notes on Graser Seminars," by J. W. Pettit et al. GRASER calculates the development of the wave vector potential as a function of two dimensions: X, distance along the axis of the graser body, and time, t. The wave vector potential is that initiated by a single spontaneously emitted gamma ray at X=0. GRASER generates a family of plots that describe the pumping as well as the development of the wave vector potential. These plots are placed on separate subplot areas, one above the other, that have a common time scale so that relationships between points on the two plots can be easily ascertained. Copies of sample plots are included to clarify this description.

II. DATA INPUT TO GRASER

There are twenty-four variables that GRASER reads in to evaluate the graser kinetics of a particular transition in a solid of specified properties. These variables are stored on an indirect access permanent file called DATA. The variables in DATA can be changed to any value by the person operating program GRASER. A call to execute program GRASER first executes program GHANGE, which has the job of reading in all variables on DATA, asking the operator which variable he desires to change and then inputting the new value of that variable. This procedure is repeated until the operator signals that he desires that program GRASER be executed at which time program GHANGE places all the changed variables into DATA and GHANGE terminates. Program GHANGE will verify all of the changes made by the operator by outputting to the operator the values as they are being written onto DATA. Program GRASER is then executed. First it reads the current contents of DATA as input, then begins analyzing the kinetics of the transition given by DATA. In program GRASER, the variables read in from DATA are assigned meanings according to the order in which they are read in. The inputted variables are first read into array XDAT(I) and program GRASER prints out the variables as they are read in so that the operator may ascertain that the variables have their desired values before proceeding with the plotting. The order in which the variables are read in, the meaning assigned to each, the required units and input format are given in Table I.

III. METHOD OF COMPUTATION

Program GRASER first computes a number of parameters from the data read in from DATA. These are:

$\lambda_2 = 1/\tau_2$, decay constant of upper graser level;

$\lambda_1 = 1/\tau_1$, decay constant of lower graser level;

$G = (2I_2 + 1)(2I_1 + 1)$, statistical weight factor;

$R = 5.37 \times 10^{-4} * E_\gamma^2/A$, recoil energy in eV;

$f = \mathrm{Exp}(-2/3 * R/K\Theta)$, Mössbauer fraction at $T = 0$ K;

$\sigma_\gamma = (2.45 \times 10^{-15}/E_\gamma^2) * G * f * \tau_1/(1 + \alpha) * \Gamma * (\tau_1 + \tau_2)$, asymptotic value of the resonant absorption or stimulated emission cross section, cm^2;

$\Delta\Gamma = (\lambda_1 + \lambda_2)(\Gamma - 1.0)/2.0$, excess half-breadth of recoilless line;

$\mu_h = \mu_h(\text{barns}) \times 1.0 \times 10^{-24}$ cm^2, host absorption cross section, cm^2;

$\mu_p = \mu_p(\text{barns}) \times 1.0 \times 10^{-24}$ cm^2, parent absorption cross section, cm^2;

$$\mu = \left.\begin{array}{ll} (N_h - N_p)\mu_h + N_p\mu_p, & \text{if } N_h \neq 0 \\ = N_p\mu_p, & \text{if } N_h = 0 \end{array}\right\} \text{ nonresonant absorption coefficient, cm}^{-1};$$

$X_{max} = X_{end}/\mu$ $\Big\}$ distances in cm over which to calculate the
$X_{min} = X_{start}/\mu$ $\Big\}$ vector potential;

$t_{max} = 2.4 * \tau_2$, maximum time in seconds at which pumping and vector potential are calculated

Program GRASER now solves the four first-order, coupled differential equations which describe the pumping. The solutions are given at twenty-six values of time between zero and t_{max}. The equations are

$$\mathring{N}_p = -R(t)\ N_p(t)\ , \tag{1}$$

$$\mathring{N}_2 = \xi\eta\ R(t)\ N_p(t) - \lambda_2\ N_2(t)\ , \tag{2}$$

$$\mathring{N}_1 = \xi(1-\eta)\ R(t)\ N_p(t) + \lambda_2\ N_2(t) - \lambda_1\ N_1(t)\ , \tag{3}$$

$$\mathring{Q} = N_2(t) - G\ N_1(t)\ . \tag{4}$$

The symbols have the following meanings:

$N_p(t)$ is the parent nuclide number density as a function of time. This is included to allow for depletion of the parent.

$N_2(t)$ is the number density of the upper graser level as a function of time.

$N_1(t)$ is the number density of the lower graser level as a function of time.

$Q(t)$ is the time-integral of the inversion density, $N^*(t)$, given by $N^*(t) = N_2(t) - G\ N_1(t)$, where G is the statistical weight factor.

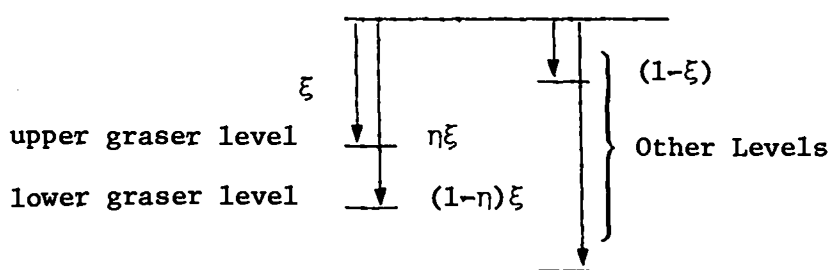
The rate function R(t) is defined as

$$R(t) = \int_0^\infty \sigma_{cap}(V)\ \Phi(V,t)\ dV\ e^{-t/t_n},$$

where V is the velocity of the neutrons, $\sigma_{cap}(V)$ is the velocity-dependent neutron capture cross section of the parent nuclide and $\Phi(V,t)$ is the time- and velocity-dependent neutron flux per unit velocity, given by

$$\Phi(V,t) = S\ (V\Sigma_s t)^2\ e^{-(V\Sigma_s t)}.$$

The above expression for the neutron flux per unit velocity is strictly valid only for an infinite homogeneous hydrogenous moderator, so we attach a leakage term e^{-t/t_n} to allow for the loss of neutrons out of the sample. t_n is the neutron lifetime, Σ_s is the macroscopic scattering cross section of the hydrogeneous moderator and S is the number of fast neutrons injected by the neutron source burst at t = 0. The neutron capture cross section is described by a single effective neutron capture resonance at energy E_o with resonance integral I_γ and a 1/V contribution whose thermal capture cross section is σ_{th}. The capture rate at a time t is thus given by $R(t)N_p(t)$. The formation of the graser isomeric levels from neutron capture and subsequent decay of the compound nucleus is described by the variables ξ and η.



ξ is the fraction of all captures that go to either graser level in a time short compared to the lifetime of the graser level. $(1-\xi)$ is the fraction of all captures that go to other levels. η is the fraction, of those captures that go to either graser level, which go to the upper graser level. $(1-\eta)$ is the fraction of ξ that go to the lower level. This essentially gives two degrees of freedom to describe decay processes that may be quite complex. To clarify these concepts, consider as an example the nuclide ^{82}Kr which goes to ^{83}Kr upon neutron capture and has a Mössbauer transition, between a 9.2-keV level and the ground state, of lifetime 2×10^{-7} seconds. Measurements of activation cross sections indicate that two-thirds of all captures go to an isomeric level above the 9.2-keV Mössbauer level. This capture level has a lifetime of 1.85 hours, so that it is useless for feeding the Mössbauer level, although all decays eventually proceed through the upper Mössbauer level. So in this case $\xi = 0.33$ and $\eta = 1.0$ since all captures that do not get stuck in the 1.85-hour level do go to the upper isomeric level and the ground state is not fed directly by pumping.

GRASER calls a library-supplied ordinary differential equation solver subroutine named ODE (D204A ODE in PIM-2) with absolute and relative errors of 0.000001 and will print out a flag if it cannot solve the equations to the specified tolerances. This situation has never arisen over the few months that GRASER has been used to date, but if it does, ODE will calculate the solution as accurately as it can in 500 steps and then continue, so that a fatal execution error is avoided.

GRASER next computes the vector potential as a function of time, holding X constant. This is done for five equally spaced values of X between the ranges of X supplied by the program operator as input. These computations are stored to be plotted later. The equation used to evaluate the vector potential, A(x,t), is

$$A(x,t) = e^{-\frac{(\lambda_2 t + \mu X)}{2}} [1.0 + e^{-\Delta\Gamma t}(\mathrm{BESSL} - 1.0)],$$

$$\text{where BESSL} = I_o\{2\sqrt{Q(t)X}\} \text{ if } Q(t)X > 0,$$
$$= J_o\{2\sqrt{Q(t)X}\} \text{ if } Q(t)X < 0,$$
$$\text{and} \quad \Delta\Gamma = (\lambda_1 + \lambda_2)(\Gamma - 1.0)/2.0.$$

This completes the computational portion of GRASER. The remainder of GRASER is devoted to plotting the output.

IV. OUTPUT SECTION OF GRASER

The finished output consists of two subplots on a single page. The lower plot gives $N_p(t)$, $N_1(t)$, $N_2(t)$ and $Q(t)$, and the upper plot gives $A(x,t)$ vs t for five values of X within the ranges specified by the user. The time scale is normalized to units of $\lambda_2 t$. In order to convert the scale to absolute time units, one multiplies by τ_2. Similarly, for labeling of the distance X, it is given in units of absorption lengths (i.e., $\mu x = 3$, $\mu x = 6$, etc); in order to convert to centimeters, one divides by μ, the absorption coefficient. A number of other parameters are also listed to the right of the plots. Sample copies of the output are appended for illustration.

GRASER uses the graphics package DISSPLA to generate the plots. The output may be sent either to a Tektronics terminal, to a Zeta plotter or to the FR80 microfiche by specifying the value of the last input variable IPLOT.

IPLOT = 1, output sent to Tektronics terminal,

IPLOT = 2, output sent to Zeta plotter,

IPLOT = 3, output sent to FR80 microfiche.

V. OPERATING DETAILS

GRASER is the name of a procedure file that does all of the necessary file manipulations and operations. There are four files associated with the graser program. A list of these is:

1. GRASER: A procedure file
2. TEST2 : The name of the program that does the computation
3. DATA : A data file
4. GHANGE: A program that is used to change the data on DATA interactively
5. DAT : A temporary working storage file used by GHANGE

These files and programs are set up for operation on the KRONOS operating system and reside on machine zero.

To use GRASER, one must first sign on machine zero and enter the BATCH subsystem. Then one gets file GRASER with the control statement GET, GRASER CR*. To execute the procedure file GRASER, one types in —GRASER CR. This will cause the proper files to be retrieved and the DISSPLA graphics package to be attached to the working files. Program GHANGE is first executed. It reads in the file DATA, outputs it to the terminal and asks the program operator what variable he desires to change. Table I defines each variable and gives its ordinal number. When the integer number corresponding to one datum is entered, program GHANGE then asks for the new value of that variable. This must be entered with the format indicated on Table I. When it is entered, GHANGE again asks for a variable to be changed and this procedure is repeated as often as the program operator desires. When the program operator is satisfied that the data file contains the proper values for a run, he enters zero when GHANGE next asks what variable is to be changed. This causes GHANGE to print out the amended data list, write it back on to DATA and store it.

Program TEST2 is then executed with no additional interaction with the program operator. Program TEST2 performs all of the computation and plotting described in the previous section. TEST2 first prints out the data as it has been read in from DATA and also outputs a number of parameters that have been calculated. TEST2 then prints a message that plotting is commencing and asks the operator to enter line speed and term type when executing the program from a Tektronics terminal. At this point, while TEST2 is waiting for input before plotting, the program operator may take the time to make a hard copy of the

---

*CR This indicates carriage return. The characters to be typed in are underlined. This is the same convention as in the NOS manuals.

data and parameters to be associated with the plot if he is using the Tektronix terminal. One may also type in a message explaining what variable has been changed or what one is looking for, before making the hard copy. When it is desired to produce the plot, one enters 120,3 CR.

When it is desired to output to the microfiche or plot on the Zeta plotter, the procedure is slightly different. First of all, IPLOT on the data file must be set to its proper value for the desired output device. This is done as if one were changing any other variable on DATA, as described above. To send the output file to be processed on microfiche, one executes GRASER in the usual manner as described above, but the program will terminate without asking for the line speed and term type (as it does when outputting to the Tektronics terminal). The following control card must be executed in order to write the output to the microfiche processor: PFICHE, FR80,FR80.NØS FR-80 FORMAT TEST CR. The response from the computer is FICHE COMPLETE. This is described in Chap. 8 of the LASL Guide to NOS, PIM-4, p. 8-4 next to DISSPLA File.

When one is operating at a terminal which has a Zeta plotter, the output can be plotted by setting IPLOT equal to 2. The output will be on a file called ZETA and, after the program has stopped, one executes the following control cards to commence plotting on the Zeta:

REWIND,ZETA CR
TERM, TTYD CR
LIST,F=ZETA CR.

When the plotting on a Tektronics terminal is completed it is necessary to press the carriage return and enter STOP CR one or more times to rapidly terminate the program. In the other cases, the program terminates rapidly. To execute the program again, simply enter —GRASER CR, make any changes desired when it is indicated to do so, enter 0 to terminate GHANGE and execute TEST2 and so forth. To sign off, enter BYE CR.

The next several pages illustrate several examples of the use of GRASER.

SIGN-ON EXAMPLE (User inputs are underlined, carriage return follows input)

```
@WORKER STATUS: L=?? M=01 N=?? R=09 S=02
idm

 76/08/25. 12.48.26.
LASL 6600 -  0 NOS TIME SHAPING.          NOS1.1-419D 760802
USER NUMBER: 84952,
TERMINAL:      22,TTY
RECOVER /SYSTEM: BATCH
$RFL.50000.
/GET,GRASER
1./-GRASER

THIS IS THE PRESENT CONTENTS OF DATA

2. XDAT(1) = 2.0000000E+22    I = 1
   XDAT(2) = 5.0000000E+22    I = 2
   XDAT(3) = 2.0000000E-07    I = 3
   XDAT(4) = 2.0000000E-07    I = 4
   XDAT(5) = 0    I = 5
   XDAT(I) = 10    I = 6
   XDAT(I) = 7110    I = 7
   XDAT(I) = 9.200000    I = 8
   XDAT(I) = 3.500000    I = 9
   XDAT(I) = 4.500000    I = 10
   XDAT(I) = 1    I = 11
   XDAT(I) = .330000    I = 12
   XDAT(I) = 40    I = 13
   XDAT(I) = 200    I = 14
   XDAT(I) = 45    I = 15
   XDAT(I) = 450    I = 16
   XDAT(I) = 2.800000    I = 17
   XDAT(I) = 83    I = 18
   XDAT(I) = 10    I = 19
   XDAT(I) = 6    I = 20
   XDAT(I) = 18    I = 21
   XDAT(I) = 1.400000    I = 22
   XDAT(23) = 6.0000000E-08    I = 23
   IPLOT = 3    I = 24

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 24

ENTER THE NEW VALUE
? 1

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 1

ENTER THE NEW VALUE
? 5.0E+22

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
3.? 0

4.THE CONTENTS OF DATA IS NOW
```

```
XDAT(1) = 5.0000000E+22    I = 1
XDAT(2) = 5.0000000E+22    I = 2
XDAT(3) = 2.0000000E-07    I = 3
XDAT(I) = 2.0000000E-07    I = 4
XDAT(I) = 0    I = 5
XDAT(I) = 10    I = 6
XDAT(I) = 7110    I = 7
XDAT(I) = 9.200000    I = 8
XDAT(I) = 3.500000    I = 9
XDAT(I) = 4.500000    I = 10
XDAT(I) = 1    I = 11
XDAT(I) = .330000    I = 12
XDAT(I) = 40    I = 13
XDAT(I) = 200    I = 14
XDAT(I) = 45    I = 15
XDAT(I) = 450    I = 16
XDAT(I) = 2.800000    I = 17
XDAT(I) = 83    I = 18
XDAT(I) = 10    I = 19
XDAT(I) = 6    I = 20
XDAT(I) = 18    I = 21
XDAT(I) = 1.400000    I = 22
XDAT(23) = 6.0000000E-08    I = 23
IPLOT = 1    I = 24
5.INCREASED NUMBER OF FAST NEUTRONS          *DEL*
                                             *DEL*
```

1. This executes GRASER
2. Look to Table I for the meaning of each entry
3. This terminates GHANGE and causes TEST2 to be executed
4. GHANGE verifies the changes by outputting the data while writing to DATA
5. Message written by program operator. Be sure to hit the ESC key after writing each line!

SECOND EXAMPLE SHOWING THE OUTPUT FROM TEST2 (which is called GRASER in the message)

```
-GRASER

THIS IS THE PRESENT CONTENTS OF DATA

 XDAT(1) = 2.0000000E+21    I = 1
 XDAT(2) = 5.0000000E+22    I = 2
 XDAT(3) = 2.0000000E-07    I = 3
 XDAT(4) = 2.0000000E-07    I = 4
 XDAT(5) = 0    I = 5
 XDAT(I) = 10    I = 6
 XDAT(I) = 71.10000    I = 7
 XDAT(I) = 9.200000    I = 8
 XDAT(I) = 3.500000    I = 9
 XDAT(I) = 4.500000    I = 10
 XDAT(I) = 1    I = 11
 XDAT(I) = .330000    I = 12
 XDAT(I) = 40    I = 13
 XDAT(I) = 200    I = 14
 XDAT(I) = 45    I = 15
 XDAT(I) = 450    I = 16
 XDAT(I) = 2.800000    I = 17
 XDAT(I) = 83    I = 18
 XDAT(I) = 10    I = 19
 XDAT(I) = 0    I = 20
 XDAT(I) = 6    I = 21
 XDAT(I) = 1.400000    I = 22
 XDAT(23) = 6.0000000E-08    I = 23
 IPLOT = 1    I = 24

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 20

ENTER THE NEW VALUE
? 6

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 21

ENTER THE NEW VALUE
? 18

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 0

THE CONTENTS OF DATA IS NOW

 XDAT(1) = 2.0000000E+21    I = 1
 XDAT(2) = 5.0000000E+22    I = 2
 XDAT(3) = 2.0000000E-07    I = 3
 XDAT(I) = 2.0000000E-07    I = 4
 XDAT(I) = 0    I = 5
 XDAT(I) = 10    I = 6
 XDAT(I) = 71.10000    I = 7
 XDAT(I) = 9.200000    I = 8
 XDAT(I) = 3.500000    I = 9
 XDAT(I) = 4.500000    I = 10
 XDAT(I) = 1    I = 11
 XDAT(I) = .330000    I = 12
 XDAT(I) = 40    I = 13
 XDAT(I) = 200    I = 14
 XDAT(I) = 45    I = 15
 XDAT(I) = 450    I = 16
 XDAT(I) = 2.800000    I = 17
 XDAT(I) = 83    I = 18
 XDAT(I) = 10    I = 19
 XDAT(I) = 6    I = 20
 XDAT(I) = 18    I = 21
 XDAT(I) = 1.400000    I = 22
 XDAT(23) = 6.0000000E-03    I = 23
 IPLOT = 1    I = 24
EFFECT AT LARGER DEPTHS    *DEL*
```

Message written in by program operator

```
PROGRAM GRASER NOW BEING EXECUTED

THE DATA READ IN ARE

 XDAT(1) = 2.0000000E+21
 XDAT(2) = 5.0000000E+22
 XDAT(3) = 2.0000000E-07
 XDAT(4) = 2.0000000E-07
 XDAT(5) = 0
 XDAT(I) = 10
 XDAT(I) = 71.10000
 XDAT(I) = 9.200000
 XDAT(I) = 3.500000
 XDAT(I) = 4.500000
 XDAT(I) = 1
 XDAT(I) = .330000
 XDAT(I) = 40
 XDAT(I) = 200
 XDAT(I) = 45
 XDAT(I) = 450
 XDAT(I) = 2.800000
 XDAT(I) = 83
 XDAT(I) = 10
 XDAT(I) = 6
 XDAT(I) = 18
 XDAT(I) = 1.400000
 XDAT(23) = 6.0000000E-08
 IPLOT = 1
 S = 2.0000000E+21    ABSCO = 3.555000    G = .800000
 ETA = 1    XMAX = 5.063291    XNP = 5.0000000E+22    TAU = 2.0000000E
-07
 XMIN = 1.687764    DTH = 450    ALPHA = 2.800000
 SIGM = 2.9830945E-19    SU = 3.500000    SL = 4.500000
 FMX = .979039    TXSECT = 45    RESINT = 200    IPLOT = 1
 ES = 1.400000
     CALCULATING VECTOR POTENTIAL
1
                PLOTTING COMMENCING
                ...................

 NO. OF FIRST PLOT  0

ENTER LINE SPEED (CPS), TERM TYPE
? 120,3
```

This causes plotting to begin on the Tektronics

THIRD EXAMPLE SHOWING HOW TO OUTPUT TO MICROFICHE

```
PROGRAM GRASER NOW BEING EXECUTED

THE DATA READ IN ARE

 XDAT(1) = 5.0000000E+22
 XDAT(2) = 5.0000000E+22
 XDAT(3) = 2.0000000E-07
 XDAT(4) = 2.0000000E-07
 XDAT(5) = 0
 XDAT(I) = 10
 XDAT(I) = 7110
 XDAT(I) = 9.200000
 XDAT(I) = 3.500000
 XDAT(I) = 4.500000
 XDAT(I) = 1
 XDAT(I) = .330000
 XDAT(I) = 40
 XDAT(I) = 200
 XDAT(I) = 45
 XDAT(I) = 450
 XDAT(I) = 2.800000
 XDAT(I) = 83
 XDAT(I) = 10
 XDAT(I) = 6
 XDAT(I) = 12
 XDAT(I) = 1.400000
 XDAT(23) = 6.0000000E-08
 IPLOT = 3
 S = 5.0000000E+22   AESCO = 355.5000   G = .800000
 ETA = 1   XMAX = .0506329   XNP = 5.0000000E+22   TAU = 2.0000000E-07
 XMIN = .0168776   DTH = 450   ALPHA = 2.800000
 SIGM = 2.9830945E-13   SU = 3.500000   SL = 4.500000
 FMX = .379039   TXSECT = 45   RESINT = 200   IPLOT = 3
 ES = 1.400000
     CALCULATING VECTOR POTENTIAL
1
                        PLOTTING COMMENCING
                        ...................

 NO. OF FIRST PLOT  0

2
 THE SUB-
 PLOT NO.   1 WITH THE TITLE

 HAS BEEN COMPLETED. PLOT ID READS
 PLOT 1     14.03.53   WED 25 AUG. 1976    JOB-P27B022.  LASL,  DISSPLA VER 1.5

 DATA FOR PLOT

 NO. OF CURVES DRAWN  4
 ------------------------------
 HORIZ. AXIS LENGTH   5.0 INS.
 VERT. AXIS LENGTH    4.0 INS.
 ------------------------------
 HORIZ. ORIGIN 0.          VERT.  ORIGIN 0.
 ------------------------------
 HORIZ. AXIS LINEAR
 STEP SIZE 5.0000E-01 UNITS/INCH
 ------------------------------
 VERT.  AXIS LINEAR
 STEP SIZE 1.2500E+22 UNITS/INCH
 ------------------------------

   .....................................
   . LOCATION OF CURRENT PHYSICAL ORIGIN .
   . X=   1.00     Y=   1.00     INCHES .
   .    FROM LOWER LEFT CORNER OF PAGE  .
   .....................................

2
 THE SUB-
 PLOT NO.   2 WITH THE TITLE
      UP US T
 HAS BEEN COMPLETED. PLOT ID READS
 PLOT 2     14.04.02   WED 25 AUG. 1976    JOB-P27B022.  LASL,  DIS
SPLA VER 1.5

 DATA FOR PLOT

 NO. OF CURVES DRAWN  5
 ------------------------------
 HORIZ. AXIS LENGTH   5.0 INS.
 VERT. AXIS LENGTH    4.0 INS.
 ------------------------------
 HORIZ. ORIGIN 0.          VERT.  ORIGIN-3.0000E+00
 ------------------------------
 HORIZ. AXIS LINEAR
 STEP SIZE 5.0000E-01 UNITS/INCH
 ------------------------------
 VERT.  AXIS LINEAR
 STEP SIZE 1.0000E+00 UNITS/INCH
 ------------------------------

   .....................................
   . LOCATION OF CURRENT PHYSICAL ORIGIN .
   . X=   1.00     Y=   6.00     INCHES .
   .    FROM LOWER LEFT CORNER OF PAGE  .
   .....................................
```

STOP

```
 STOP
 PFICHE,FR80,FR80.NOS FR-80 FORMAT TEST
FICHE COMPLETE
```

This entry causes the output to be processed on microfiche

ANOTHER EXAMPLE OF EXECUTING GRASER AT A TEKTRONICS TERMINAL

```
 *INTERRUPTED*
STOP
 *TERMINATED*
/-GRASER

THIS IS THE PRESENT CONTENTS OF DATA

 XDAT(1) = 2.0000000E+22   I = 1
 XDAT(2) = 5.0000000E+20   I = 2
 XDAT(3) = 2.0000000E-07   I = 3
 XDAT(4) = 2.0000000E-07   I = 4
 XDAT(5) = 0    I = 5
 XDAT(I) = 10    I = 6
 XDAT(I) = 71.10000    I = 7
 XDAT(I) = 9.200000    I = 8
 XDAT(I) = 3.500000    I = 9
 XDAT(I) = 4.500000    I = 10
 XDAT(I) = 1    I = 11
 XDAT(I) = .330000    I = 12
 XDAT(I) = 4000    I = 13
 XDAT(I) = 200    I = 14
 XDAT(I) = 45    I = 15
 XDAT(I) = 450    I = 16
 XDAT(I) = 2.200000    I = 17
 XDAT(I) = 23    I = 18
 XDAT(I) = 10    I = 19
 XDAT(I) = 1.200000    I = 20
 XDAT(I) = 6    I = 21
 XDAT(I) = 1.400000    I = 22
 XDAT(23) = 6.0000000E-03    I = 23
 IPLOT = 1    I = 24

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 7

ENTER THE NEW VALUE
? 7110

ENTER THE NUMBER OF THE VARIABLE TO BE CHANGED
? 0

THE CONTENTS OF DATA IS NOW

 XDAT(1) = 2.0000000E+22   I = 1
 XDAT(2) = 5.0000000E+20   I = 2
 XDAT(3) = 2.0000000E-07   I = 3
 XDAT(I) = 2.0000000E-07   I = 4
 XDAT(I) = 0    I = 5
 XDAT(I) = 10    I = 6
 XDAT(I) = 7110    I = 7
 XDAT(I) = 9.200000    I = 8
 XDAT(I) = 3.500000    I = 9
 XDAT(I) = 4.500000    I = 10
 XDAT(I) = 1    I = 11
 XDAT(I) = .330000    I = 12
 XDAT(I) = 4000    I = 13
 XDAT(I) = 200    I = 14
 XDAT(I) = 45    I = 15
 XDAT(I) = 450    I = 16
 XDAT(I) = 2.800000    I = 17
 XDAT(I) = 83    I = 18
 XDAT(I) = 10    I = 19
 XDAT(I) = 1.200000    I = 20
 XDAT(I) = 6    I = 21
 XDAT(I) = 1.400000    I = 22
 XDAT(23) = 6.0000000E-03    I = 23
 IPLOT = 1    I = 24

PROGRAM GRASER NOW BEING EXECUTED

THE DATA READ IN ARE

 XDAT(1) = 2.0000000E+22
 XDAT(2) = 5.0000000E+20
 XDAT(3) = 2.0000000E-07
 XDAT(4) = 2.0000000E-07
 XDAT(5) = 0
 XDAT(I) = 10
 XDAT(I) = 7110
 XDAT(I) = 9.200000
 XDAT(I) = 3.500000
 XDAT(I) = 4.500000
 XDAT(I) = 1
 XDAT(I) = .330000
 XDAT(I) = 4000
 XDAT(I) = 200
 XDAT(I) = 45
 XDAT(I) = 450
 XDAT(I) = 2.800000
 XDAT(I) = 83
 XDAT(I) = 10
 XDAT(I) = 1.200000
 XDAT(I) = 6
 XDAT(I) = 1.400000
 XDAT(23) = 6.0000000E-03
 IPLOT = 1
 S = 2.0000000E+22   ABSCU = 3.555000   G = .300000
 ETA = 1   XMAX = 1.697764   XNP = 5.0000000E+20   TAU = 2.0000000E
-07
 XMIN = .337553   DTH = 450   ALPHA = 2.800000
 SIGM = 2.9830945E-19   SU = 3.520000   SL = 4.500000
 FMX = .979039   TXSECT = 45   RESINT = 200   IPLOT = 1
 ES = 1.400000
     CALCULATING VECTOR POTENTIAL
1
                 PLOTTING COMMENCING
                 ...................

 NO. OF FIRST PLOT  0

ENTER LINE SPEED (CPS), TERM TYPE
?
```

Trying to study absorption effect *DEL*

120,3 This program user input beings the plotting

```
 'STOP
STOP
/STOP
 $TERMINATED$
```

TO START GRASER AGAIN, ENTER:

-GRASER

THE COMPUTEP MUST INDICATE THAT IT HAS STOPPED OR TERMINATED BEFORE EXECUTING GRASER AGAIN.

SAMPLE PLOT

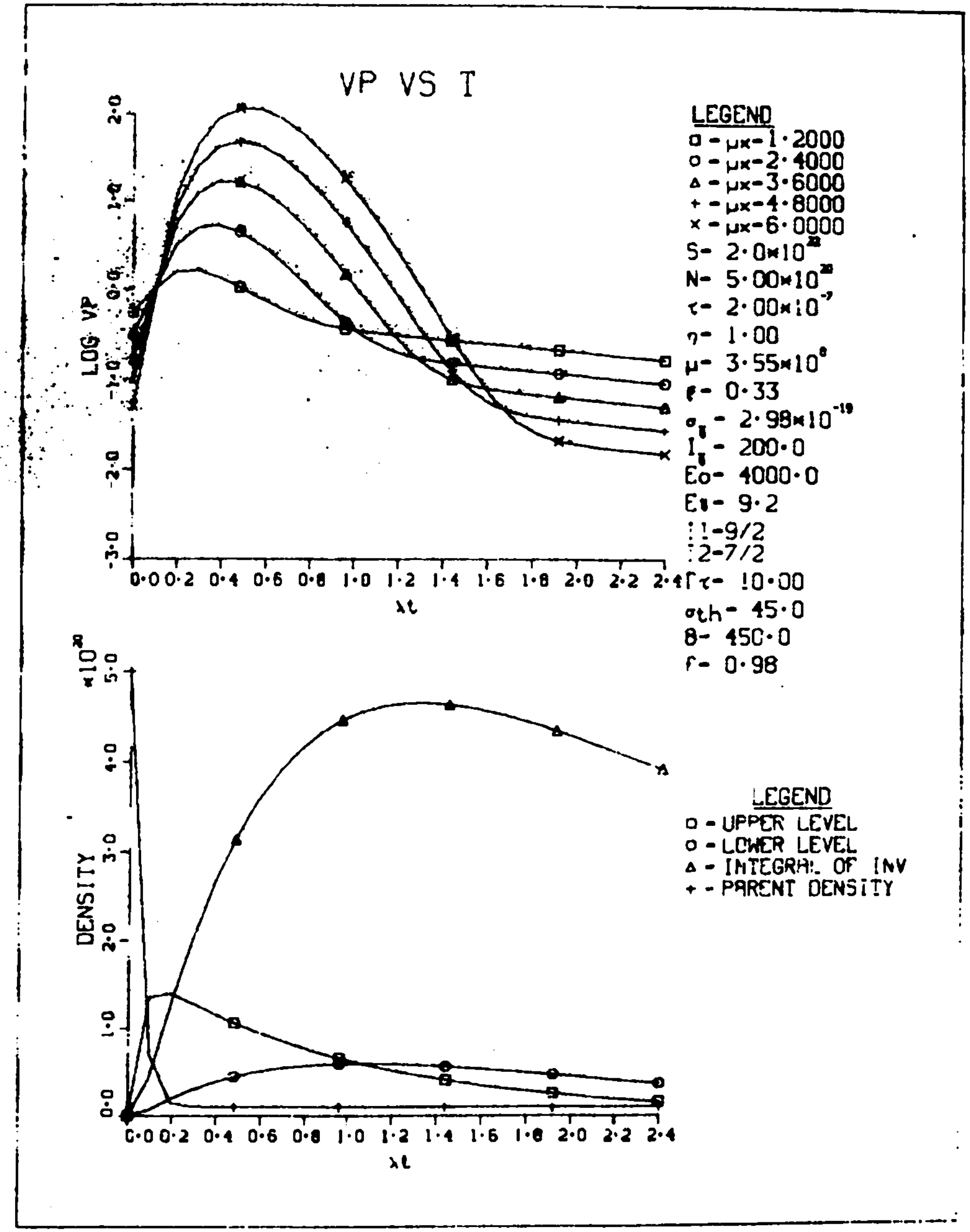


TABLE I

ORDER OF INPUT, DEFINITIONS, UNITS AND FORMAT FOR VARIABLES IN FILE DATA

INPUT ORDER		VARIABLE		FORMAT
XDAT(1)	→	S	(Number of fast neutrons injected at t=0, cm^{-3})	E7.1
XDAT(2)	→	N_p	(Number density of parent nuclei at t=0, cm^{-3})	E7.1
XDAT(3)	→	τ_2	(Lifetime of the upper graser level, seconds)	E7.1
XDAT(4)	→	τ_1	(Lifetime of the lower graser level, seconds)	E7.1
XDAT(5)	→	N_h	(Number density of the host, cm^{-3})	E7.1
XDAT(6)	→	μ_h	(Microscopic nonresonant absorption cross section of the host, barns)	F10.4
XDAT(7)	→	μ_p	(Microscopic nonresonant absorption cross section of the parent, barns)	F10.4
XDAT(8)	→	E_γ	(Energy of the gamma transition, keV)	F10.4
XDAT(9)	→	I_2	(Spin of the upper graser level, dimensionless)	F10.4
XDAT(10)	→	I_1	(Spin of the lower graser level, dimensionless)	F10.4
XDAT(11)	→	η	(Fraction of all upper graser levels formed by neutron capture in the parent, or isomer ratio, dimensionless)	F10.4
XDAT(12)	→	ξ	(Fraction of all captures which form either graser level, dimensionless)	F10.4
XDAT(13)	→	E_o	(Effective energy of a single neutron capture resonance in the parent nuclide, eV)	F10.4
XDAT(14)	→	I_γ	(Resonance integral of the capture resonance in the parent, barns)	F10.4
XDAT(15)	→	σ_{th}	(Thermal neutron capture cross section of the parent nuclide, barns)	F10.4
XDAT(16)		Θ	(Effective Debye temperature of the host-parent system, °K)	F10.4
XDAT(17)	→	α	(Internal conversion coefficient of the graser isomer, dimensionless)	F10.4
XDAT(18)	→	A	(Atomic mass number of the graser isomer, dimensionless)	F10.4
XDAT(19)	→	Γ	(Factor by which the graser gamma line is inhomogeneously broadened, dimensionless)	F10.4
XDAT(20)	→	X_{start}	(Distance in absorption lengths at which it is desired that GRASER start calculating the vector potential, dimensionless)	F10.4

TABLE I - Continued

INPUT ORDER		VARIABLE		FORMAT
XDAT(21)	→	X_{end}	(Distance in absorption lengths at which it is desired that GRASER finish calculating the vector potential, dimensionless)	F10.4
XDAT(22)	→	Σ_s	(Macroscopic neutron elastic scattering cross section of the neutron moderator, cm^{-1})	F10.4
XDAT(23)	→	t_n	(Neutron lifetime in the moderator, seconds)	E7.1
IPLOT	→	The plot option. IPLOT must be either 1, 2, 3 IPLOT = 1: Plot is done on Tektronics scope IPLOT = 2: Plot is done on Zeta plotter, output is on file ZETA IPLOT = 3: Plot is done by FR80 camera on microfiche		I1

COMPLETE PROGRAM LISTING

LASL Identification: LP-0712.

```
LIST,F=GRASER

GET(TEST2,DAT,DATA,GHANGE)
REWIND(DATA,DAT,GHANGE)
RETURN(LGO)
FUN(G,I=GHANGE)
REWIND(DATA,DAT)
REPLACE(DAT=DATA)
RFL(55000)
GET(DATA)
ATTACH(DISSPLA/UN=LIBRARY)
LIBRARY(DISSPLA)
REWIND(TEST2)
RETURN(LGO)
FUN(G,I=TEST2)
READY.

LIST,F=TEST2

      PROGRAM GRASER (INPUT,OUTPUT,DATA,TAPE2=DATA)
C
C     GRASER PROGRAM USING ODE
C
      EXTERNAL F
      COMMON X,XE,XNP,ETA,DEC1,G,RESINT,EO,TNL,ES,S,DEC,SI
      DIMENSION TC(26),X2(26),X1(26),QB(26),XDAT(23)
      DIMENSION Q(26),VECT(26,5),A(26),IPAK(100),XLC(5)
      DIMENSION Y(4),WORK(184),IWORK(5),XNPT(26)
C
C     PRINT MESSAGE THAT GRASER IS BEING EXECUTED
C
      PRINT 600
  600 FORMAT(//,*PROGRAM GRASER NOW BEING EXECUTED*,/)
      PRINT 627
  627 FORMAT(*THE DATA READ IN ARE*,//)
C
```

```
C      READ IN DATA
C
       READ(2,666) XDAT(1)
       READ(2,666) XDAT(2)
       READ(2,666) XDAT(3)
       READ(2,666) XDAT(4)
       READ(2,666) XDAT(5)
       OUTPUT,XDAT(1)
       OUTPUT,XDAT(2)
       OUTPUT,XDAT(3)
       OUTPUT,XDAT(4)
       OUTPUT,XDAT(5)
   666 FORMAT(E7.1)
       DO 13 I=6,22
       READ(2,667) XDAT(I)
   667 FORMAT(F10.4)
       OUTPUT,XDAT(I)
    13 CONTINUE
       READ(2,666) XDAT(23)
       OUTPUT,XDAT(23)
       READ(2,668) IPLOT
       OUTPUT,IPLOT
   668 FORMAT(I1)
       S=XDAT(1)
       XNP=XDAT(2)
       TAU=XDAT(3)
       TAU1=XDAT(4)
       XNH=XDAT(5)
       VH=XDAT(6)
       VP=XDAT(7)
       EG=XDAT(8)
       SU=XDAT(9)
       SL=XDAT(10)
       ETA=XDAT(11)
       SI=XDAT(12)
       EO=XDAT(13)
       RESINT=XDAT(14)
       TXSECT=XDAT(15)
       DTH=XDAT(16)
       ALPHA=XDAT(17)
       AMAS=XDAT(18)
       FLIN=XDAT(19)
       VST=XDAT(20)
       VND=XDAT(21)
       ES=XDAT(22)
       TNL=1.0/XDAT(23)
C
```

```
C     CALCULATE INITIAL VALUES
C
      NEQ=4
      IFLAG=1
      RELERR=0.000001
      ABSERR=0.000001
      T=0.0
      TOUT=0.0
      DEC=1.0/TAU
      DEC1=1.0/TAU1
      IS1=SL*2.0
      IS2=SU*2.0
      G=((2.0*SU)+1.0)/((2.0*SL)+1.0)
      SIG=2.45E-15/(EG**2)*G*(1.0/(1.0+ALPHA))
      RECOIL=5.37E-4*(EG**2)/AMAS
      FMX=EXP(-(3.0/2.0)*RECOIL/(8.617E-5*DTH))
      SIGM=SIG/FLIN*FMX*TAU1*(1.0/(TAU1+TAU))
      DELGAM=(DEC+DEC1)*(FLIN-1.0)/2.0
      ABH=UH*1.0E-24
      ABP=UP*1.0E-24
      XDIF=XNH-XNP
      IF(XDIF .LT. 0.0) XDIF=0.0
      ABSCO=((XDIF)*ABH)+(XNP*ABP)
      XMAX=UND/ABSCO
      XMIN=UST/ABSCO
      TMAX=2.4*TAU
      XE=TXSECT*1.0E-24*220000.0
      XNPT(1)=XNP
      X2(1)=0.0
      X1(1)=0.0
      TC(1)=0.0
      Q(1)=0.0
      QB(1)=0.0
      Y(1)=XNP
      Y(2)=0.0
      Y(3)=0.0
      Y(4)=0.0
      OUTPUT,S,ABSCO,G,ETA,XMAX,XNP,TAU,XMIN,DTH,ALPHA
      OUTPUT,SIGM,SU,SL,FMX,TXSECT,RESINT,IPLOT,ES
C
C     CALCULATE DENSITIES FOR TWENTY SIX VALUES OF TIME
C
      DO 10 I=2,26
      TOUT=TOUT+TMAX/25.0
      TC(I)=TOUT
      CALL ODE(F,NEQ,Y,T,TOUT,RELERR,ABSERR,IFLAG,WORK,IWORK)
      XNPT(I)=Y(1)
      X2(I)=Y(2)
      X1(I)=Y(3)
```

```
      Q(I)=Y(4)*SIGM/4.0*DELGAM
      QB(I)=Q(I)*1.0E+19
      T=TOUT
      IF(IFLAG .EQ. 2) GO TO 10
      OUTPUT,IFLAG
   10 CONTINUE
      PRINT 150
  150 FORMAT(5X,*CALCULATING VECTOR POTENTIAL*)
C
C     CALCULATE VECTOR POTENTIAL FOR  FIVE VALUES OF X AND
C     TWENTY SIX VALUES OF TIME
C
      DO 30 I=1,26
      DO 30 J=1,5
      XX=((XMAX-XMIN)/4.0*(J-1))+XMIN
      XLC(J)=XX*ABSCO
      TIM=TC(I)
      ARGA=Q(I)*XX
      IF(ARGA .LT. 0.0) GO TO 25
      ARG=2.0*SQRT(Q(I)*XX)
      BESSL=FI0(ARG)*EXP(ARG)
      GO TO 26
   25 ARGA=-Q(I)*XX
      ARG=2.0*SQRT(ARGA)
      BESSL=FJ0(ARG)
   26 VECT(I,J)=EXP(-0.5*((TIM/TAU)+(ABSCO*XX)))*
     1(1.0+(EXP(-DELGAM*TIM)*(BESSL-1.0)))
      VECT(I,J)=ABS(VECT(I,J))
   30 CONTINUE
C
C     SCALE TIME IN UNITS OF LAMDA*T
C
      DO 40 I=2,26
   40 TC(I)=TC(I)*DEC
C
C     PLOT RESULTS
C
      TOP=XNP
      YSTP=TOP/5.0
      GO TO (100,200,300),IPLOT
  100 CALL TEKTRN
      GO TO 400
  200 CALL ZETA
      GO TO 400
  300 CALL FR80(3)
  400 CONTINUE
      CALL BGNPL(0)
      CALL PHYSOR(1.0,1.0)
```

```
      CALL BASALF(8HSTANDARD)
      CALL MIXALF(6HL/CSTD)
      CALL MX3ALF(5HGREEK,1H&)
      CALL MX4ALF(8HL/CGREEK,1H+)
      CALL TITLE(1H ,1,5H+L(T),5,7HDENSITY,7,5.0,4.0)
      CALL GRAF(0.0,0.2,2.5,0.0,YSTP,TOP)
      CALL CURVE(TC,X2,26,5)
      CALL CURVE(TC,X1,26,5)
      CALL CURVE(TC,QB,26,5)
      CALL CURVE(TC,XNPT,26,5)
      CALL LINES(12HUPPER LEVEL$,IPAK,1)
      CALL LINES(12HLOWER LEVEL$,IPAK,2)
      CALL LINES(16HINTEGRAL OF INV$,IPAK,3)
      CALL LINES(15HPARENT DENSITY$,IPAK,4)
      CALL LEGEND(IPAK,5,5.0,2.0)
      CALL MESSAG(5H&G+T=,5,5.0,4.75)
      CALL REALNO(FLIN,2,4HABUT,4HABUT)
      CALL MESSAG(5H+S  =,5,5.0,4.5)
      CALL REALNO(TXSECT,1,4HABUT,4HABUT)
      CALL MESSAG(4H(TH),4,5.1,4.46)
      CALL MESSAG(3H&Q=,3,5.0,4.25)
      CALL REALNO(DTH,1,4HABUT,4HABUT)
      CALL MESSAG(4H(F)=,4,5.0,4.0)
      CALL REALNO(FMX,2,4HABUT,4HABUT)
      CALL ENDGR(1)
      CALL PHYSOR(1.0,6.0)
      CALL TITLE(7HVP VS T,7,5H+L(T),5,6HLOG VP,6,5.0,4.0)
      TOP=0.0
      DO 50 J=1,5
      DO 50 I=1,26
   50 TOP=AMAX1(VECT(I,J),TOP)
      TOP=ALOG10(TOP)
      TOP=TOP+0.5
      ITOP=TOP
      TOP=ITOP
      CALL GRAF(0.0,0.2,2.5,-3.0,1.0,TOP)
      DO 70 J=1,5
      DO 60 I=1,26
      IF(VECT(I,J) .LT. 0.001) VECT(I,J)=0.001
   60 A(I)=ALOG10(VECT(I,J))
      CALL CURVE(TC,A,26,5)
   70 CONTINUE
      JDUM=LINEST(IPAK,80,8)
      CALL LINES(7H+M(X= $,IPAK,1)
      CALL LINES(7H+M(X= $,IPAK,2)
      CALL LINES(7H+M(X= $,IPAK,3)
      CALL LINES(8H+M(X=  $,IPAK,4)
      CALL LINES(8H+M(X=  $,IPAK,5)
```

```
      CALL LEGEND(IPAK,5,5.0,2.95)
      CALL REALNO(XLC(1),4,5.6,3.75)
      CALL REALNO(XLC(2),4,5.6,3.55)
      CALL REALNO(XLC(3),4,5.6,3.35)
      CALL REALNO(XLC(4),4,5.6,3.15)
      CALL REALNO(XLC(5),4,5.6,2.95)
      CALL MESSAG(2HS=,2,5.0,2.7)
      CALL REALNO(S,-1,4HABUT,4HABUT)
      CALL MESSAG(2HN=,2,5.0,2.45)
      CALL REALNO(XNP,-2,4HABUT,4HABUT)
      CALL MESSAG(3H+T=,3,5.0,2.2)
      CALL REALNO(TAU,-2,4HABUT,4HABUT)
      CALL MESSAG(3H+C=,3,5.0,1.95)
      CALL REALNO(ETA,2,4HABUT,4HABUT)
      CALL MESSAG(3H+M=,3,5.0,1.7)
      CALL REALNO(ABSCO,-2,4HABUT,4HABUT)
      CALL MESSAG(3H+X=,3,5.0,1.45)
      CALL REALNO(SI,2,4HABUT,4HABUT)
      CALL MESSAG(4H+S =,4,5.0,1.2)
      CALL REALNO(SIGM,-2,4HABUT,4HABUT)
      CALL MESSAG(2H+G,2,5.1,1.1)
      CALL MESSAG(3HI =,3,5.0,0.95)
      CALL REALNO(RESINT,1,4HABUT,4HABUT)
      CALL MESSAG(2H+G,2,5.1,0.86)
      CALL MESSAG(4HE+D=,4,5.0,0.7)
      CALL REALNO(ED,1,4HABUT,4HABUT)
      CALL MESSAG(4HE+G=,4,5.0,0.45)
      CALL REALNO(EG,1,4HABUT,4HABUT)
      CALL MESSAG(3HI1=,3,5.0,0.2)
      CALL INTNO(IS1,4HABUT,4HABUT)
      CALL MESSAG(2H/2,2,4HABUT,4HABUT)
      CALL MESSAG(3HI2=,3,5.0,0.0)
      CALL INTNO(IS2,4HABUT,4HABUT)
      CALL MESSAG(2H/2,2,4HABUT,4HABUT)
      CALL ENDGR(2)
      CALL LASLOGO(2.0,6.5,1.0)
      CALL ENDPL(0)
      CALL DONEPL
      STOP $ END
C
      SUBROUTINE F(T,Y,YP)
      COMMON X,XE,XNP,ETA,DEC1,G,RESINT,ED,TNL,ES,S,DEC,SI
      DIMENSION Y(4),YP(4)
      YP(1)=-RATE(T)*Y(1)
      YP(2)=(ETA*RATE(T)*Y(1)*SI)-DEC*Y(2)
      YP(3)=(1.0-ETA)*RATE(T)*Y(1)*SI+DEC*Y(2)-DEC1*Y(3)
      YP(4)=Y(2)-G*Y(3)
      RETURN
      END
C
```

```
C
      FUNCTION RATE(TY)
      COMMON X,XE,XNP,ETA,DEC1,G,RESINT,ED,TNL,ES,S,DEC,SI
      F=RATFUN(TY)
      RATE=EXP(-TY*TNL)*(XE+F)*S
      RETURN $ END
C
      FUNCTION RATFUN(TV)
      COMMON X,XE,XNP,ETA,DEC1,G,RESINT,ED,TNL,ES,S,DEC,SI
      EX=ED*1.60209E-12
      XM=1.65976E-24
      V=SQRT(2.0*EX/XM)
      FLUX=((V*ES*TV)**2)*EXP(-V*ES*TV)
      AREA=RESINT*1.0E-24*V
      RATFUN=FLUX*AREA
      RETURN $ END
C
      FUNCTION  FIO(ZP)                                                 C342
*********************************************************************** C342
*     FIO(X) COMPUTES THE REAL BESSEL FUNCTION IO TIMES EXP(-X) OF THE * C342
*     ARGUMENT X .GT. 0                                                * C342
*********************************************************************** C342
      DIMENSION P(9,2),Q(8,2)                                           C342
      DIMENSION P0( 5,2),Q0(4,2)                                        C342
      EQUIVALENCE (EYIO,EYEIO,FIO)                                      C342
      DATA (P(J),J=1,9)/                                                C342
     * .3306413889260115E+22,                                           C342
     * .7929083985978161E+21,                                           C342
     * .4340939205795706E+20,                                           C342
     * .9506389969703280E+18,                                           C342
     * .1032250768917613E+17,                                           C342
     * .6124926562275096E+14,                                           C342
     * .2042613182306772E+12,                                           C342
     * .3651524854074652E+09,                                           C342
     * .2787285947599744E+06/                                           C342
      DATA (Q(J),J=1,8)/                                                C342
     * .3306413889260115E+22,                                           C342
     *-.3369507371721258E+20,                                           C342
     * .1704434675708268E+18,                                           C342
```

```
     +-.561815860040322 9E+15,                                          C342
     + .1326178904074606E+13,                                           C342
     +-.2305132440239370E+10,                                           C342
     + .2889856220077479E+07,                                           C342
     +-.2392135609522520E+04/                                           C342
      DATA (P(I),I=10,18)/                                              C342
     + .1590162083763072E+17,                                           C342
     + .5368395855470032E+16,                                           C342
     + .3756135827519507E+15,                                           C342
     + .9422080573862922E+13,                                           C342
     + .9718863641275663E+11,                                           C342
     + .4165980320912641E+09,                                           C342
     + .6879907643492760E+06,                                           C342
     + .3536566189687649E+03,                                           C342
     + .2963656768308448E-01/                                           C342
      DATA (Q(I),I=9,16)/                                               C342
     + .2606708705529648E+17,                                           C342
     + .2022142175869257E+17,                                           C342
     + .1980645892386831E+16,                                           C342
     + .6358669883569537E+14,                                           C342
     + .8204440573998663E+12,                                           C342
     + .4413044733863003E+10,                                           C342
     + .9394714900641401E+07,                                           C342
     + .6697052546975880E+04/                                           C342
      DATA (P0(J),J=1,5)/                                               C342
     + .7064621469033392E-13,                                           C342
     +-.1436279642653691E-08,                                           C342
     + .5286489105163732E-05,                                           C342
     +-.3830439714639074E-02,                                           C342
     + .2829406075894839E+00/                                           C342
      DATA (Q0(J),J=1,4)/                                               C342
     + .1770837992384417E-12,                                           C342
     +-.3606859805023185E-08,                                           C342
     + .1338520247147810E-04,                                           C342
     +-.1007723338776333E-01/                                           C342
      DATA (P0(I),I=6,10)/                                              C342
     + .2069343093741415E-12,                                           C342
     +-.4422252452216457E-08,                                           C342
     + .1366441906147109E-04,                                           C342
```

```
     *-.7387885439863839E-02,                                          C342
     * .3710722095990018E+00/                                          C342
      DATA (Q0(I),I=5,8)/                                              C342
     * .5187073908684608E-12,                                          C342
     *-.1108727721754080E-07,                                          C342
     * .3430127364212944E-04,                                          C342
     *-.1866802289017835E-01/                                          C342
      I=1                                                              C342
      X=ZP                                                             C342
      Z=X*X                                                            C342
      IF(X .GT. 16.)GO TO 1                                            C342
      IF(X .GT.  8.) I=2                                               C342
      EYEI0=(P(1,I)+Z*(P(2,I)+Z*(P(3,I)+Z*(P(4,I)+Z*(P(5,I)+Z*(P(6,I)+Z C342
     *     *(P(7,I)+Z*(P(8,I)+Z*P(9,I)))))))))/                        C342
     *      (Q(1,I)+Z*(Q(2,I)+Z*(Q(3,I)+Z*(Q(4,I)+Z*(Q(5,I)+Z*(Q(6,I)+Z C342
     *     *(Q(7,I)+Z*(Q(8,I)+Z))))))))                                C342
      IF(I .EQ. 1) EYEI0=EXP(-X)*EYEI0                                 C342
      RETURN                                                           C342
    1 Z=1./(64*Z)                                                      C342
      Q01=(P0(1,1)+Z*(P0(2,1)+Z*(P0(3,1)+Z*(P0(4,1)+Z*P0(5,1)))))/     C342
     *   (Q0(1,1)+Z*(Q0(2,1)+Z*(Q0(3,1)+Z*(Q0(4,1)+Z))))               C342
      Q01=Q01/(8*X)                                                    C342
      P01=(P0(1,2)+Z*(P0(2,2)+Z*(P0(3,2)+Z*(P0(4,2)+Z*P0(5,2)))))/     C342
     *   (Q0(1,2)+Z*(Q0(2,2)+Z*(Q0(3,2)+Z*(Q0(4,2)+Z))))               C342
      EYEI0=(P01+Q01)/SQRT(X)                                          C342
      RETURN                                                           C342
      END                                                              C342
      FUNCTION  FJ0(ZP)                                                C340A
*********************************************************************** C340A
*     FJ0(X) COMPUTES THE REAL BESSEL FUNCTION J0 OF THE REAL ARGUMENT * C340A
*     X .GT. 0                                                        * C340A
*********************************************************************** C340A
      DIMENSION P(9,2),Q(8,2)                                          C340A
      DIMENSION P0( 5,2),Q0(4,2)                                       C340A
      EQUIVALENCE (EYJ0,EYEJ0,FJ0)                                     C340A
      DATA (P(J),J=1,9)/                                               C340A
     * .1947930043950693E+22,                                          C340A
     *-.1030790592800841E+21,                                          C340A
     * .2305050589779804E+19,                                          C340A
     *-.2576446288321790E+17,                                          C340A
```

```
     + .1608225312767766E+15,                                      C340A
     +-.5864983055305724E+12,                                      C340A
     + .1247014316352538E+10,                                      C340A
     +-.1441160259923904E+07,                                      C340A
     + .7057781986067161E+03/                                      C340A
      DATA (Q(J),J=1,8)/                                           C340A
     + .1947930043950693E+22,                                      C340A
     + .1866656846683426E+20,                                      C340A
     + .8988812598699360E+17,                                      C340A
     + .2872923345367091E+15,                                      C340A
     + .6757621507401545E+12,                                      C340A
     + .1218915111066339E+10,                                      C340A
     + .1685448234198584E+07,                                      C340A
     + .1690305128256799E+04/                                      C340A
      DATA (P(I),I=10,18)/                                         C340A
     + .1915735987569625E+21,                                      C340A
     +-.1001781172709201E+20,                                      C340A
     + .2205602059805950E+18,                                      C340A
     +-.2402516341734763E+16,                                      C340A
     + .1445553026998197E+14,                                      C340A
     +-.5012886478578140E+11,                                      C340A
     + .9974512187863315E+08,                                      C340A
     +-.1058363049975556E+06,                                      C340A
     + .4653169559540574E+02/                                      C340A
      DATA (Q(I),I=9,16)/                                          C340A
     + .1915744022256219E+21,                                      C340A
     + .1955478121458678E+19,                                      C340A
     + .1019029152817659E+17,                                      C340A
     + .3594650569271279E+14,                                      C340A
     + .9693226280485690E+11,                                      C340A
     + .2020934854546515E+09,                                      C340A
     + .3977520908299590E+06,                                      C340A
     + .3951153319317984E+03/                                      C340A
      DATA (P0(J),J=1,5)/                                          C340A
     +-.2669230888166635E-12,                                      C340A
     + .3487301310346338E-08,                                      C340A
     + .1420614795808099E-04,                                      C340A
     + .8789741021455747E-02,                                      C340A
     + .5423439570936309E+00/                                      C340A
      DATA (Q0(J),J=1,4)/                                          C340A
     +-.3345384807898387E-12,                                      C340A
```

```
     ◆ .4358138840301680E-08,                                             C340A
     ◆ .1797068587349931E-04,                                             C340A
     ◆ .1165656249458522E-01/                                             C340A
      DATA (P0(I),I=6,10)/                                                C340A
     ◆-.1616796790064454E-11,                                             C340A
     ◆ .1538702516195051E-07,                                             C340A
     ◆ .4373398193154472E-04,                                             C340A
     ◆ .1812080215474507E-01,                                             C340A
     ◆ .7331365490421742E+00/                                             C340A
      DATA (Q0(I),I=5,8)/                                                 C340A
     ◆-.2026354274154102E-11,                                             C340A
     ◆ .1927565757246875E-07,                                             C340A
     ◆ .5490008915137954E-04,                                             C340A
     ◆ .2294894906253812E-01/                                             C340A
C                                                                         C340A
C     DATA T=PI/4.                                                        C340A
C                                                                         C340A
      DATA T/.7853981633974831/                                           C340A
      I=1                                                                 C340A
      X=ZP                                                                C340A
      Z=X◆X                                                               C340A
      IF(X .GT. 14.)GO TO 1                                               C340A
      IF(X .GT. 9.5) I=2                                                  C340A
      EYEJ0=(P(1,I)+Z◆(P(2,I)+Z◆(P(3,I)+Z◆(P(4,I)+Z◆(P(5,I)+Z◆(P(6,I)+Z   C340A
     ◆     ◆(P(7,I)+Z◆(P(8,I)+Z◆P(9,I)))))))))/                           C340A
     ◆      (Q(1,I)+Z◆(Q(2,I)+Z◆(Q(3,I)+Z◆(Q(4,I)+Z◆(Q(5,I)+Z◆(Q(6,I)+Z   C340A
     ◆     ◆(Q(7,I)+Z◆(Q(8,I)+Z))))))))                                   C340A
      EYEJ0=(1.-EYEJ0◆Z/4.)                                               C340A
      RETURN                                                              C340A
    1 Z=1./(64◆Z)                                                         C340A
      Q01=(P0(1,1)+Z◆(P0(2,1)+Z◆(P0(3,1)+Z◆(P0(4,1)+Z◆P0(5,1)))))/        C340A
     ◆   (Q0(1,1)+Z◆(Q0(2,1)+Z◆(Q0(3,1)+Z◆(Q0(4,1)+Z))))                  C340A
      Q01=Q01/(8◆X)                                                       C340A
      P01=(P0(1,2)+Z◆(P0(2,2)+Z◆(P0(3,2)+Z◆(P0(4,2)+Z◆P0(5,2)))))/        C340A
     ◆   (Q0(1,2)+Z◆(Q0(2,2)+Z◆(Q0(3,2)+Z◆(Q0(4,2)+Z))))                  C340A
      EYEJ0= (P01◆COS(X-T)+Q01◆SIN(X-T))/SQRT(X)                          C340A
      RETURN                                                              C340A
      END                                                                 C340A
 -END OF FILE-
? END
 END TEXT EDITING.
READY.
```