PORTIONS OF THIS DOCUMENT ARE ILLEGIBLE
EBT Time-Dependent Point Model Code: Description and User's Guide

Jane F. Roberts  Nermin A. Uckan

NOTICE

PORTIONS OF THIS REPORT ARE ILL legible. It has been reproduced from the best available copy to permit the broadest possible availability.

OAK RIDGE NATIONAL LABORATORY
OPERATED BY UNION CARBIDE CORPORATION FOR THE ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the Energy Research and Development Administration, United States Nuclear Regulatory Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe private-owned rights.
ORNL/CSD/TM-27

Contract No. W-7405-eng-26

COMPUTER SCIENCES DIVISION

EBT TIME-DEPENDENT POINT MODEL CODE: DESCRIPTION AND USER'S GUIDE

Jane F. Roberts
Computer Sciences Division

and

Nermin A. Uckan
Fusion Energy Division

Date Published: July 1977

NOTICE

This document contains information of a preliminary nature. It is subject to revision or correction and therefore does not represent a final report.

UNION CARBIDE CORPORATION, NUCLEAR DIVISION
operating the
Oak Ridge Gaseous Diffusion Plant • Oak Ridge National Laboratory
Oak Ridge Y-12 Plant • Paducah Gaseous Diffusion Plant

for the
ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>v</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>1</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2. PHYSICS MODEL</td>
<td>2</td>
</tr>
<tr>
<td>3. PROGRAM LOGIC AND ORGANIZATION</td>
<td>4</td>
</tr>
<tr>
<td>3.1. Description of Routines</td>
<td>4</td>
</tr>
<tr>
<td>3.2. Integration</td>
<td>10</td>
</tr>
<tr>
<td>3.3. Input</td>
<td>11</td>
</tr>
<tr>
<td>3.4. Execution</td>
<td>13</td>
</tr>
<tr>
<td>3.5. Output: Printed</td>
<td>14</td>
</tr>
<tr>
<td>3.6. Output: Plotting</td>
<td>15</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>17</td>
</tr>
<tr>
<td>APPENDIX A</td>
<td>23</td>
</tr>
<tr>
<td>APPENDIX B</td>
<td>25</td>
</tr>
<tr>
<td>APPENDIX C</td>
<td>37</td>
</tr>
<tr>
<td>APPENDIX D</td>
<td>81</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

This work was supported by the Fusion Energy Division at Oak Ridge National Laboratory.

We would like to specifically acknowledge and thank C. L. Hedrick of the Fusion Energy Division and S. E. Attenberger of the Computer Sciences Division for their help in distinguishing, interpreting, and resolving physical and numerical difficulties encountered during code development.
ABSTRACT

A D-T time-dependent point model has been developed to assess the energy balance in an EBT reactor plasma. Flexibility is retained in the model to permit more recent data to be incorporated as they become available from the theoretical and experimental studies. This report includes the physics models involved, the program logic, and a description of the variables and routines used. All the files necessary for execution are listed, and the code, including a post-execution plotting routine, is discussed.

1. INTRODUCTION

As a first step in assessing the energy balance in an EBT reactor plasma, a simulation model (point model) has been developed in which the bulk plasma is characterized by simple energy and particle containment times, with appropriate density and temperature scaling. In view of uncertainties in the theoretical coefficients and the present lack of experimental evidence in this area, flexibility is retained in the simulation model to permit more recent data to be incorporated as they become available.

The full time-dependent point model for D-T fusion in EBT consists of nine coupled nonlinear ordinary differential equations (discussed in Sect. 2). The program logic and organization is outlined in Sect. 3. A description of the variables, a listing of the files, and the execution routine are given in appendixes.
2. PHYSICS MODEL

The particle and energy balance equations used in the code are

\[
\frac{dn_D}{dt} = S_{D,COLD} + S_{D,BEAM}(1 - f_D) - n_D n_T <\sigma v> DT
\]
\[+ n_e n_{OD} <\sigma v> \text{ioniz} - \frac{n_D}{\tau_{p,D}}, \quad (1)
\]

\[
\frac{dn_T}{dt} = S_{T,COLD} - S_{D,BEAM f_D} - n_D n_T <\sigma v> DT + n_e n_{OT} <\sigma v> \text{ioniz} - \frac{n_T}{\tau_{p,D}}, \quad (2)
\]

\[
\frac{dn_\alpha}{dt} = S_{D,BEAM f_D} + n_D n_T <\sigma v> DT - \frac{n_\alpha}{\tau_{p,\alpha}}, \quad (3)
\]

\[
\frac{dn_I}{dt} = (n_D n_T <\sigma v> DT + S_{D,BEAM f_D}) I_N + \frac{n_D}{\tau_{p,D}} I_D + \frac{n_T}{\tau_{p,T}} I_T + \frac{n_\alpha}{\tau_{p,\alpha}} I_\alpha
\]
\[+ \frac{n_I}{\tau_{p,I}} (I_I - 1), \quad (4)
\]

\[
\frac{dn_{HI}}{dt} = (n_D n_T <\sigma v> DT + S_{D,BEAM f_D}) I_N + \frac{n_D}{\tau_{p,D}} I_D + \frac{n_T}{\tau_{p,T}} I_T + \frac{n_\alpha}{\tau_{p,\alpha}} I_\alpha
\]
\[+ \frac{n_{HI}}{\tau_{p,H_I}} (I_{HI} - 1), \quad (5)
\]

\[
\frac{dn_{OD}}{dt} = \frac{n_D}{\tau_{p,D}} f_{RD} - n_e n_{OD} <\sigma v> \text{ioniz}, \quad (6)
\]

\[
\frac{dn_{OT}}{dt} = \frac{n_T}{\tau_{p,T}} f_{RT} - n_e n_{OT} <\sigma v> \text{ioniz}, \quad (7)
\]
\[ \frac{d(p_e)}{dt} = p_{\mu} + S_{\text{DBEAM}}(f_{\text{DB},e} + f_{\alpha U,\beta,e}) + n_D n_T C_0 <\sigma v>_{\text{DT}}(U \alpha f_{\alpha,e} - 3T_e) \]

\[ - \frac{p_e}{\tau_{E,e}} - p_{ei} - p_{\text{RAD}} , \]

\[ \frac{d(p_i)}{dt} = n_D n_T C_0 <\sigma v>_{\text{DT}}(U \alpha f_{\alpha,i} - 3T_i) + S_{\text{DBEAM}}(U \beta \alpha f_{\text{DB},i} + f_{U,\beta,\alpha,i}) \]

\[ + p_{ei} - \frac{p_i}{\tau_{E,i}} , \]

where

\[ n_e = n_D + n_T + 2n_{\alpha} + z_1 n_{\alpha} + z_{HI} n_{HI} , \]

\[ n_i = n_D + n_T , \]

\[ p_e = \frac{3}{2} n_e T_e , \]

\[ p_i = \frac{3}{2} n_i T_i . \]

As a first step, a simple model has been analyzed. The equations implemented in the current version of the EBT point model code (see Appendix A) are a subset of Eqs. 1-9. The parameters used in Eqs. 1-9 are listed in Appendix B.

The code was developed using the full set of equations. The modifications were accomplished by changing expressions in various subroutines (maintaining the full expressions in comment statements) and by specific choices of input parameters. This will be discussed in detail later in this report.

The units of both the model and the code are mks ("temperatures," representing energy, are in keV). The only numerical problem this has caused so far is in expressions like \( n_D n_T C_0 <\sigma v>_{\text{DT}} \). Since densities are \( \sim 10^{20} \) and cross-sections \( \sim 10^{-20} \), it is necessary in coding to force the
3. PROGRAM LOGIC AND ORGANIZATION

Figure 1 shows how control flows from one routine to another. Specifically, MAIN controls the run. INIT and PRINT1 are called at the very beginning for initialization of variables and parameters and initial printout; the rest of the subroutines are called repeatedly as integration progresses in time.

A simplified flow chart appears in Fig. 2. There are only two main loops: the larger loop represents integration of the equations in time until some final, specified time (controlled in MAIN), and the smaller loop represents repeated integration over a specific time-step until the convergence criterion is met for all equations (controlled by EXTINT).

Except for EXTINT, ERROR, F, PRINT2, and GETVAR, the subroutines have no arguments—all variables, parameters, and combinational terms are passed to routines through common blocks. Table 1 shows which common blocks are currently used in which routines. All commons appear to their full extent in MAIN (for initial allocation of core) and nearly all commons appear in PRINT2, to facilitate output (for debugging, for example).

3.1. Description of Routines

A listing of the EBT time-dependent point model code is in Appendix C. A descriptive listing of all the variables used in the code appears in Appendix B. The routines operate as follows:
MAIN

MAIN controls the integration of the equations, calculates global variables for output, controls the frequency of output (in integration time) and checks feedbacks on cold fueling rates. The cold fueling rates are decreased or increased in order to keep \( n_D \) and \( n_T \) near the operating value \( n_{DTf} \), after the neutrals reach steady state. The feedback of \( \dot{S}_{DCOLD}^* \) if \( X > TISTAR \) (the time for \( n_{Qi} \) to reach steady state), is as follows for \( i = D, T \):

if \( n_i > n_{DTf} \) and \( \frac{dn_i}{dt} > 0 \), \( S_{iCOLD} = \alpha S_{iCOLD} \cdot \beta < 1 ; \)

if \( n_i < n_{DTf} \) and \( \frac{dn_i}{dt} < 0 \), \( S_{iCOLD} = \beta \mid iLOSS \), \( \beta > 1 \).

MAIN also controls the turning off of microwave power after ignition.

INIT

INIT does all the initialization necessary for the run: makes initial assignments, reads input data from EBT7EQ.IN, sets up a Y-array of dependent variables for EXTINT, and opens the output files specified by PRTFIL and PLTFIL.

PRINT1

PRINT1 does the initial printout into the files PRTFIL and PLTFIL.

PRINT2(X, HÔ)

PRINT2 does the printout at time \( X \) during integration of the equations.
GETVAR(Y)

GETVAR takes the dependent variables out of the array Y (used by EXTINT) and gives them mnemonic names in the common block VARBLE.

F(X, Y, DY)

F calculates the right-hand sides of the differential equations for EXTINT (see the list of equations in Appendix A and the descriptive list of variables in Appendix B).

FCOUNT counts how many times F has been called by EXTINT; it normally takes on values like 7, 13, 21, 53, etc. FCOUNT > 200 indicates problems with the equations converging smoothly.

SIGDT

SIGDT reads <\sigma>\_DT as a function of T\_i from SIGVDT.DAT (the first time it is called). Then <\sigma>\_DT is calculated through linear interpolation of the values corresponding to the temperatures that bracket T\_i (found by a binary search of the array TEMP).

SIGION

SIGION calculates <\sigma>\_ioniz as a function of T\_e, and <\sigma>\_cx as a function of T\_i.

\[
<\sigma>_{ioniz} = 10^{-6} \exp \left\{ \sum_{i=0}^{6} a_i \ln(10^3 T_e) \right\} ,
\]

\[
<\sigma>_{cx} = 10^{-6} \exp \left\{ \sum_{i=0}^{8} b_i \ln(10^3 T_i) \right\} .
\]
COLMB

COLMB calculates $P_{e,i}$, the power transferred to the electrons from the ions during Coulomb collisions:

$$P_{e,i} = 1.5 \times 10^{-19} \frac{n_e}{T_e} \frac{Z_j^2 n_j}{A_j} \sum_{j=D,T} (T_e - T_i) (\ln \Lambda).$$

where

$$\ln \Lambda = 39.4 - \ln \left( \frac{\sqrt{n_e}}{T_e} \right), \text{ in the full model; }$$

$$\ln \Lambda = 17.0, \text{ currently.}$$

TAUS

Taus calculates all the particle and energy confinement times, using Kovrizhnykh's results. ²

$$\tau_{p,j} = C_j \frac{1}{v_j} \left( 1 + \frac{v_j^2}{\Omega_j^2} \right), \quad j = D, T, \alpha, I, HI,$$

$$v_j = 2.218 \times 10^{-18} \ln \Lambda \frac{n_j Z_j^2}{\sqrt{A_j} T_i^{3/2}},$$

$$C_j = 3 \left[ \frac{R_T}{R_c} \left( 1 + 10^{-3} \frac{Z_j E_R C}{T_i} \right) \right]^2,$$

$$\Omega_j = 10^3 \frac{T_i}{Z_j R_c a} \left( 1 + 10^{-3} \frac{Z_j E_R C}{T_i} \right),$$

$$\frac{1}{\tau_{p,i}} = \frac{n_D}{n_i} \frac{1}{\tau_{p,D}} + \frac{n_T}{n_i} \frac{1}{\tau_{p,T}}.$$
\[
\frac{1}{\tau_{\text{cx}}} = \langle n_o \rangle \frac{n_D^n + n_T^n}{n_i}
\]

\[
\tau_{E,\text{COND}} = \frac{3}{7} \tau_{p,i} \quad (\text{currently, } \tau_{E,\text{COND}} = 0),
\]

\[
\tau_S = 0.24 \times 10^{18} \frac{T_e^{7/2}}{n_e} \quad (\text{currently, } \tau_S = 0),
\]

\[
\frac{1}{\tau_{E,i}} = \frac{1}{\tau_{p,i}} + \frac{1}{\tau_{\text{cx}} + \tau_{E,\text{COND}}} \quad (\text{currently, } \frac{1}{\tau_{E,i}} = \frac{1}{\tau_{p,i}} + \frac{1}{\tau_{\text{cx}}}),
\]

\[
\tau_{E,e} = \tau_{p,i}.
\]

**FRACT**

FRACT calculates the fractions of energy transferred by various mechanisms. (See ref. 1.)

\[
f_{\alpha i} = \frac{T_e}{50} - 0.37 \left( \frac{T_e}{50} \right)^{7/4},
\]

\[
f_{\alpha e} = 1 - f_{\alpha i},
\]

\[
f_{DB,i} = 0.5 \exp \left[ -0.0916 \frac{U_{\text{BEAM}}}{U_{CR}} \left( 1 + 0.5 \frac{\tau_S}{\tau_{\text{cx}}} \right) \right] \]

\[
+ 0.5 \exp \left[ -0.635 \frac{U_{\text{BEAM}}}{U_{CR}} \left( 1 + 0.5 \frac{\tau_S}{\tau_{\text{cx}}} \right) \right],
\]

\[
f_{DB,e} = 1 - f_{DB,i} \quad (\text{currently, } f_{DB,i} = f_{DB,e} = 0);
\]

\[
f_{RD} = f_{RT} = 0; \quad f_H = 0.
\]
POWRAD

POWRAD calculates the power losses due to various types of radiation: Bremsstrahlung, line radiation, and synchrotron radiation.

\[ P_{\text{BREM}} = 3 \times 10^{-21} Z^2 n_e^2 T_e^{1/2} \]
\[ Z_{\text{eff}} = \frac{\sum Z_j^2 n_j}{\sum Z_j n_j} = \frac{\sum Z_j^2 n_j}{n_e} \text{, for } j = D, T, \alpha, I, \text{HI} \]
\[ P_{\text{LINE}} = 2 \times 10^{-17} n_{\text{HI}} n_e (\text{currently, } P_{\text{LINE}} = 0) \]
\[ P_{\text{SYNC}} = 1.5 \times 10^5 \sqrt{\frac{n_e B_s}{a}} T_e^{1/4} (1 - R_e)(1 + \frac{T_e}{204}) \]

\[ P_{\text{RAD}} = P_{\text{BREM}} + P_{\text{SYNC}} - F_{\text{HSYNC}} + P_{\text{LINE}} \]

SPUTR

SPUTR will calculate the sputtering rates of impurities off the walls. Currently, this routine is not even called (no impurities are implemented in the code).

FUFRAC

FUFRAC will calculate \( f_D \), the fraction of the deuterium beam which undergoes suprathermal fusion.

\[ f_D = \frac{U_{\text{DBEAM}} n_T F(U_{\text{DBEAM}}, T_e)}{1.76 \times 10^4 n_e} \]

Currently FUFRAC is not even called, as there is no deuterium beam.
As the physics is understood better and as more things are implemented in the code, the form of the subroutines will change, particularly TAUS, FRACI, POWRAD, SPUTR, FUFRAC, and of course F. And as the expressions become more complicated, the calling sequence of some of the subroutines by F may need to be changed to keep everything consistent in time.

3.2 Integration

Integration of the differential equations is done by EXTINT, a deferred-limit integrator written by J. P. Boris and N. K. Windsor, using the extrapolation method developed by R. Bulirsch and J. Stoer.\(^3\)

The arguments are described in the initial comments in the listing of EXTINT in Appendix C.

EXTINT calls a subroutine, ERROR (also contained in the file EXTINT.F4), which determines whether or not all the equations have met the convergence criterion (EPS) for integration over the time-step (H0). If not, after HMAX extrapolation attempts, H0 is halved. The original ERROR did a relative error check,

\[
\frac{|dY_i/dX|}{Y_{imax}} < \text{EPS} ,
\]

for convergence of the ith equation, \( S(I) = Y_{imax} \) = the maximum value yet reached by \( Y_i \). But this has been changed to give a strict relative error check,

\[
\frac{|dY_i/dX|}{Y_i} < \text{EPS} ,
\]

for convergence of the ith equation.
3.3. Input

The input file, EBT7EQ.IN, is listed in Appendix C. The input formats in INIT (except for the first record) are such that the first eight spaces are skipped. Thus the variable name is typed in those first eight spaces, next to each input value, simplifying modification of the input file for the user.

The input variables are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LABEL (1st record)</td>
<td>An 80-character descriptive title of the run, to appear on both printer output and DISSPLA plots.</td>
</tr>
<tr>
<td>PRTFIL</td>
<td>A 10-character file name for the printer output.</td>
</tr>
<tr>
<td>PLTFIL</td>
<td>A 10-character file name for the (intermediate) plot file.</td>
</tr>
<tr>
<td>EPS</td>
<td>The convergence criteria for EXTINT and ERROR (EPS &lt; 10^-6 would require double precision).</td>
</tr>
<tr>
<td>NMAX</td>
<td>The maximum number of extrapolation attempts EXTINT makes before decreasing the time-step.</td>
</tr>
<tr>
<td>XSTART</td>
<td>The starting time in seconds (usually 0.0).</td>
</tr>
<tr>
<td>H0</td>
<td>The initial time-step for EXTINT in seconds.</td>
</tr>
<tr>
<td>XFINAL</td>
<td>The final time in seconds.</td>
</tr>
<tr>
<td>PRTFRQ</td>
<td>The frequency in seconds with which printer and plotting output is required, for example, every 0.5 sec.</td>
</tr>
<tr>
<td>EPSO</td>
<td>&quot;Error condition&quot; for neutral equations. Neutrals reach steady state when dn_{OD}/dt and dn_{OT}/dt are less than EPSO.</td>
</tr>
<tr>
<td>CIGNIT</td>
<td>Ignition criterion. When CPALP = CPLOSS + CIGNIT, ignition has occurred.</td>
</tr>
<tr>
<td>ZI</td>
<td>Nuclear charge of the low-Z impurity ions.</td>
</tr>
<tr>
<td>AI</td>
<td>Mass number of the low-Z impurity ions.</td>
</tr>
</tbody>
</table>
ZHI
Nuclear charge of the high-Z impurity ions.

AHl
Mass number of the high-Z impurity ions.

PMICRO
Microwave power, input as MW/m³ (converted to keV/m³/sec for computation).

UDBEAM
Deuterium beam energy.

UALPHA
The energy of the alpha particles = 3.5 MeV for D-T fusion (in keV).

PBEAM
Total beam power (0.0 for no beams).

RCURVA
VB
= the radius of curvature of the B-field.

RMAJOR
Major radius of the EBT, in meters.

RMINOR
Minor radius of the EBT, in meters.

ERATIO
10⁻³EᵣRᵣ
Tᵣ
, the ratio of the radial electric field to the ion temperature.

BFIELD
Magnitude of the B-field at the center of the midplane, in tesla.

DENOTF
Final deuterium density desired = final tritium density desired, in m⁻³.

DEN0DF
Final neutral deuteron density desired, in m⁻³.

DEN0TF
Final neutral triton density desired, in m⁻³.

TEF
The final electron temperature desired, in keV.

TIF
The final ion temperature desired, in keV.

DEND
The initial deuteron density, in m⁻³.

DENT
The initial triton density, in m⁻³.

DENA
The initial alpha particle density, in m⁻³.

DENIM
The initial low-Z impurity density, in m⁻³.

DENHIM
The initial high-Z density, in m⁻³.
DENØD  The initial neutral deuterium density, in $m^{-3}$.

DENØT  The initial neutral tritium density, in $m^{-3}$.

TE      The initial electron temperature, in keV.

TI      The initial ion temperature, in keV.

3.4. Execution

The point model code has been developed for the DECsystem-10 (PDP-10). Execution can be either in time-sharing mode or batch mode.

Figure 3 shows the execution flow. The files necessary are as follows (listings are in Appendix C).

**EBT7EQ.IN**  Input data

**EBT7EQ.F4**  Source file for point model code, containing routines MAIN, INIT, PRINT1, GETVAR, PRINT2, F, SIGDT, SIGION, COLMB, TAUSS, FUFRAZ, SPUTR, FRACT, and POWRAD.

**EXTINT.F4**  Source file containing EXTINT and ERROR.

**SIGVDT.DAT**  Resident disk file containing $<ov>DT$ as a function of $Ti$ (used by SIGDT).

**DIS7EQ.F4**  Source file of DISSPLA plotting program.

**DISVEC.CMD**  Command files of all DISSPLA modules to be linked in to execute DISSPLA program.

The output consists of two files:

**PRINT.7EQ**  Printer output of variables and parameters in time (see example in Appendix D).

**PLOT.7EQ**  Intermediate plot file - contains data which is input to plotting program, DIS7EQ.F4 (example in Appendix D).

The names of these two output files are character input values of the variables PRTFIL and PLTFIL read in from EBT7EQ.IN, so that successive
runs can be done without overwriting the output files, merely by changing the output file names.

To run under Batch, one needs an additional file, a Batch command file such as 7EQ.CMD, listed in Appendix C. With error-checking at every other step, this file should enable the user to trace aborted runs through the messages printed in the user's Batch log.

The monitor command

```
.SUBMIT 7EQ.CMD/TIME:10:00/CORE:50K/AFTER:18:00
```

will submit the Batch job to run after 6 p.m. with 50K core (necessary for the plots), for a maximum time of 10 minutes. It will create a BATCH log file named 7EQ.LOG in the user's disk space. A /RESTART switch will force the Batch job to completely restart after a system crash (normally the Batch jobs continue from the point of interruption, unless the crash is severe).

3.5. Output: Printed

The printed output will be in the user's disk area in a file with the name the user chose to be the value of the input variable PRTFIL, which, in our example, is PRINT.7EQ.

To have this output printed, the user needs to give the monitor print command:

```
.PPRINT PRINT.7EQ
```

The /DISPOSE:RENAME switch, if used, will remove the file from the user's disk area for temporary storage in a system area until it is printed, after which the file disappears.
3.6. Output: Plotting

The DISSPLA plotting routine, DIS7EQ.F4, makes five semi-log plots (listing in Appendix C):

1. electron, ion, and alpha densities vs time,
2. electron and ion temperatures vs time,
3. total alpha power and total power losses vs time,
4. deuterium and tritium cold fueling rates vs time,
5. time-step (H0) vs time.

Simple execution of this program may require two changes by the user:

1. The input to the plotting routine is the outfile designated by the input variable PLTFIL, in our example PLOT7EQ. If a different file name is used, the OPEN and CLOSE statements in the plotting program must be changed to the new value of PLTFIL.

2. A device-initialization routine must be the first DISSPLA subroutine called. So depending on whether a Tektronix CRT terminal is being used or Versatek printer plots are desired, either the CALL TKTRN (ICPS, IMODE) or the CALL VRSTEC instruction must be implemented, and the other one commented out (see listing in Appendix C). Also, the value of ICPS should be 30 or 120 depending on the line-speed of the Tektronix terminal.

Depending on which plotting device is being used, the user must link in the routines listed either in DISVEC.CMD or DISTEK.CMD (listed in Appendix C), for example,

\texttt{.EX DIS7EQ.F4,DISVEC.CMD}.

If the Tektronix terminal is being used (handy for time-sharing), DISSPLA rings the bell upon completion of a plot and waits for a carriage return from the user before generating the next plot.

If the Versatek printer is being used to produce plots, the user needs to run the post-processor utility program VECVER:

\texttt{.R VECVER}
\texttt{FILE: }
and the plots will automatically be queued to the Versatek printer (with the /DISPOSE:RENAME switch). An example of the Versatek plots is in Appendix D.
References


Fig. 1. Subroutine hierarchy.
Fig. 2. Simplified flow chart.
Fig. 3. Execution of seven-equation point model.
Table 1. Location of common blocks in routines

<table>
<thead>
<tr>
<th>Common blocks</th>
<th>MAIN</th>
<th>INIT</th>
<th>PRINT1</th>
<th>GETVAR</th>
<th>PRINT2</th>
<th>F</th>
<th>SIGD</th>
<th>SIGM</th>
<th>COLMB</th>
<th>TAU4</th>
<th>FUR4L</th>
<th>SPUT</th>
<th>FRAC</th>
<th>P/NRAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>VARIABLE</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>INPUT</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>SIGMAS</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>TIMEP</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>TIMEE</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>TIMEX</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>FRAC</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>POWER</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>INRATE</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>HATE</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>DIFLOS</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>SPUT</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>SNATES</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>RECYCL</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>POWTRN</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>ALF</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>GLOBAL</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>'IGNITE</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>CLOG</td>
<td>x</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>PARM</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>D/FRT</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>ERRCOM</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
<tr>
<td>PRTL4</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
<td>x/</td>
</tr>
</tbody>
</table>

\(\times\) = common appears.
\(\checkmark\) = variables are used.
Appendix A

EQUATIONS IMPLEMENTED IN THE CODE

\[
\frac{dn_D}{dt} = S_{DCOLD} - n_D n_T \langle \alpha \rangle_D + n_e n_{OD} \langle \alpha \rangle_{ioniz} - \frac{n_D}{\tau_{p,D}}. \tag{1}
\]

\[
\frac{dn_T}{dt} = S_{TCOLD} - n_D n_T \langle \alpha \rangle_D + n_e n_{OT} \langle \alpha \rangle_{ioniz} - \frac{n_T}{\tau_{p,T}}. \tag{2}
\]

\[
\frac{dn_\alpha}{dt} = n_D n_T \langle \alpha \rangle_D - \frac{n_\alpha}{\tau_{p,\alpha}}. \tag{3}
\]

\[
\frac{dn_I}{dt} = 0. \tag{4}
\]

\[
\frac{dn_{HI}}{dt} = 0. \tag{5}
\]

\[
\frac{dn_{OD}}{dt} = \begin{cases} S_{DCOLD} - n_e n_{OD} \langle \alpha \rangle_{ioniz} & \text{if } \left| \frac{dn_{OD}}{dt} \right| < \epsilon_0, \\ 0 & \text{otherwise}. \end{cases} \tag{6}
\]

\[
\frac{dn_{OT}}{dt} = \begin{cases} S_{TCOLD} - n_e n_{OT} \langle \alpha \rangle_{ioniz} & \text{if } \left| \frac{dn_{OT}}{dt} \right| < \epsilon_0, \\ 0 & \text{otherwise}. \end{cases} \tag{7}
\]

\[
\frac{dp_e}{dt} = p_\mu + n_D n_T \langle \alpha \rangle_D (U_f a_e - eT_e) - \frac{p_e}{\tau_{E,e}} - p_{ei} - P_{RAD}. \tag{8}
\]

\[
\frac{dp_i}{dt} = n_D n_T \langle \alpha \rangle_D (U_f a_e - 3T_i) + p_{ei} - \frac{p_i}{\tau_{E,i}}, \tag{9}
\]
where

\[ n_e = n_D + n_T + 2n_\alpha, \]

\[ n_i = n_D + n_T, \]

\[ p_e = \frac{3}{2} n_e T_e, \]

\[ p_i = \frac{3}{2} n_i T_i. \]
### Appendix B

#### VARIABLES IN CODE

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AHI</td>
<td>$A_{HI}$</td>
<td>Atomic number of high-Z impurity ions</td>
</tr>
<tr>
<td>AI</td>
<td>$A_I$</td>
<td>Atomic number of low-Z impurity ion</td>
</tr>
<tr>
<td>ALFAE</td>
<td>$n_Dn_T &lt;\alpha v&gt; D_T \alpha ae$</td>
<td>Power delivered from $\alpha$ particles to electrons</td>
</tr>
<tr>
<td>ALFAI</td>
<td>$n_Dn_T &lt;\alpha v&gt; D_T \alpha ai$</td>
<td>Power delivered from $\alpha$ particles to ions</td>
</tr>
<tr>
<td>ALPHA</td>
<td>$10^{-3} Z_j \frac{E_{RC}}{T_1}$</td>
<td>The product of the charge of the particle $\times$ ERATIO</td>
</tr>
<tr>
<td>AREA</td>
<td>$S$</td>
<td>Surface area of bumpy torus</td>
</tr>
<tr>
<td>BDFUSN</td>
<td>$S_{DBEAM} f_D$</td>
<td>Rate of fusion of deuterium beam with plasma tritons</td>
</tr>
<tr>
<td>BFIELD</td>
<td>$B$</td>
<td>Magnetic field strength at center of midplane</td>
</tr>
<tr>
<td>BFUSNE</td>
<td>$S_{DBEAM} f_D U \alpha ai$</td>
<td>Specific power delivered to electrons by 3.5 MeV alphas from suprathermal fusion</td>
</tr>
<tr>
<td>BFUSNI</td>
<td>$S_{DBEAM} f_D U \alpha ai$</td>
<td>Specific power delivered to ions by 3.5 MeV alphas from suprathermal fusion</td>
</tr>
<tr>
<td>C</td>
<td>$C_j$</td>
<td>Power conversion factor from keV/sec/m$^3$ to MW (in MAIN)</td>
</tr>
<tr>
<td>C</td>
<td>$C_j$</td>
<td>Coefficient for $\tau_{p,j}$ (in TAUS)</td>
</tr>
</tbody>
</table>
Appendix B (continued)

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIGNIT</td>
<td></td>
<td>Ignition criterion (ignition: CPALP - CPLOSS &gt; CIGNIT), MW</td>
</tr>
<tr>
<td>COLFRJ</td>
<td>$v_j$</td>
<td>Collision frequency for the jth particle ($j = D, T, \alpha, I, HI$)</td>
</tr>
<tr>
<td>COLOM</td>
<td>$\frac{\nu_j}{n_j}$</td>
<td>The ratio of the collision frequency to the precision for particle j</td>
</tr>
<tr>
<td>COULOG</td>
<td>$\ln \Lambda$</td>
<td>Coulomb logarithm</td>
</tr>
<tr>
<td>CPALP</td>
<td></td>
<td>Total alpha power produced (in whole device), MW</td>
</tr>
<tr>
<td>CPLOSS</td>
<td></td>
<td>Total power losses (in whole device), MW</td>
</tr>
<tr>
<td>DBEAME</td>
<td>$S_{DBEAMU}DBEAM^fDB,e$</td>
<td>Specific power directly delivered to electrons from deuterium beam (keV/sec/m^3)</td>
</tr>
<tr>
<td>DBEAMI</td>
<td>$S_{DBEAMU}DBEAM^fDB,i$</td>
<td>Specific power directly delivered to ions from deuterium beam (keV/sec/m^3)</td>
</tr>
<tr>
<td>DELTAX</td>
<td>$\Delta t$</td>
<td>Change in time between successive printouts</td>
</tr>
<tr>
<td>DEN0</td>
<td>$n_0$</td>
<td>Total neutral density (m^-3) = $n_{OD} + n_{OT}$</td>
</tr>
<tr>
<td>DENOD</td>
<td>$n_{OD}$</td>
<td>Neutral deuterium density (m^-3)</td>
</tr>
<tr>
<td>DENOOF</td>
<td>$n_{ODf}$</td>
<td>Final neutral deuterium density desired (m^-3)</td>
</tr>
<tr>
<td>DENTOT</td>
<td>$n_{OT}$</td>
<td>Neutral tritium density (m^-3)</td>
</tr>
<tr>
<td>Code variable</td>
<td>Analytical expression (if any)</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>DENOTF</td>
<td>$n_{OTf}$</td>
<td>Final neutral tritium density ($m^{-3}$)</td>
</tr>
<tr>
<td>DELA</td>
<td>$n_{\alpha}$</td>
<td>Alpha particle density ($m^{-3}$)</td>
</tr>
<tr>
<td>DEND</td>
<td>$n_{D}$</td>
<td>Deuterion density ($m^{-3}$)</td>
</tr>
<tr>
<td>DENDTF</td>
<td>$n_{DTf}$</td>
<td>Final deuterion density = final triton density desired</td>
</tr>
<tr>
<td>DENE</td>
<td>$n_{e}$</td>
<td>Total electron density ($m^{-3}$)</td>
</tr>
<tr>
<td>DENHIM</td>
<td>$n_{HI}$</td>
<td>High-Z impurity density ($m^{-3}$)</td>
</tr>
<tr>
<td>DENTI</td>
<td>$n_{i}$</td>
<td>Total ion density ($m^{-3}$)</td>
</tr>
<tr>
<td>DENT</td>
<td>$n_{T}$</td>
<td>Triton density ($m^{-3}$)</td>
</tr>
<tr>
<td>DERIV</td>
<td>$\frac{d}{dt}$ $n_{\alpha}$/\tau_{p,\alpha}$,$n_{D}/\tau_{p,D}$,$n_{HI}/\tau_{p,HI}$</td>
<td>Array of right-hand sides of equations (for printout)</td>
</tr>
<tr>
<td>DIFA</td>
<td>$\frac{d}{dt}$ $n_{\alpha}$/\tau_{p,\alpha}$</td>
<td>Alpha particle diffusion loss rate</td>
</tr>
<tr>
<td>DIFD</td>
<td>$\frac{d}{dt}$ $n_{D}/\tau_{p,D}$</td>
<td>Deuterium particle diffusion loss rate</td>
</tr>
<tr>
<td>DIFHIM</td>
<td>$\frac{d}{dt}$ $n_{HI}/\tau_{p,HI}$</td>
<td>High-Z impurity particle diffusion loss rate</td>
</tr>
</tbody>
</table>
### Appendix B (continued)

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFIM</td>
<td>$\frac{n_I}{\tau_{p,I}}$</td>
<td>Low-Z impurity particle diffusion loss rate</td>
</tr>
<tr>
<td>DIFT</td>
<td>$\frac{n_T}{\tau_{p,T}}$</td>
<td>Tritium particle diffusion loss rate</td>
</tr>
<tr>
<td>DLOSS</td>
<td></td>
<td>Total losses in $n_D$ equation</td>
</tr>
<tr>
<td>DY</td>
<td>$\frac{d}{dt}$</td>
<td>Array of right-hand sides</td>
</tr>
<tr>
<td>ENTAU</td>
<td></td>
<td>Global $n\tau$ product</td>
</tr>
<tr>
<td>EPS</td>
<td>$\epsilon$</td>
<td>Convergence criterion</td>
</tr>
<tr>
<td>EPS0</td>
<td>$\epsilon_0$ $10^{-4}E_R R_C$</td>
<td>&quot;ERROR condition&quot; for neutral equations (steady state)</td>
</tr>
<tr>
<td>ERATIO</td>
<td>$\frac{10^{-4}E_R R_C}{T_i}$</td>
<td>The ratio of the radial electric field $\times$ the radius of curvature to the ion temperature</td>
</tr>
<tr>
<td>FAE</td>
<td>$f_{a,e}$</td>
<td>Fraction of 3.52 MeV alpha energy given to electrons</td>
</tr>
<tr>
<td>FAI</td>
<td>$f_{a,i}$</td>
<td>Fraction of 3.52 MeV alpha energy given to electrons</td>
</tr>
<tr>
<td>FCOUNT</td>
<td></td>
<td>Number of times EXTINT calls F during integration over one time-step</td>
</tr>
<tr>
<td>FD</td>
<td>$f_D$</td>
<td>Fraction of deuterium beam which undergoes suprathermal fusion</td>
</tr>
<tr>
<td>FDBE</td>
<td>$f_{DB,e}$</td>
<td>Fraction of injected deuterium beam energy given directly to electrons</td>
</tr>
</tbody>
</table>
### Appendix B (continued)

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDB1</td>
<td>$f_{\text{DB},i}$</td>
<td>Fraction of injected deuterion beam energy given directly to ions</td>
</tr>
<tr>
<td>FH</td>
<td>$f_H$</td>
<td>Fraction of hot-electron annuli synchrotron radiation energy transferred to background plasma</td>
</tr>
<tr>
<td>FHSYNC</td>
<td>$f_{H}^e\cdot HSYNC$</td>
<td>Power delivered to background electrons due to synchrotron radiation from hot-electron annuli</td>
</tr>
<tr>
<td>FRD</td>
<td>$f_{RD}$</td>
<td>Fraction of deuterions which are recycled</td>
</tr>
<tr>
<td>FRT</td>
<td>$f_{RT}$</td>
<td>Fraction of tritons which are recycled</td>
</tr>
<tr>
<td>HØ</td>
<td></td>
<td>The integration time-step (modified by EXTINT)</td>
</tr>
<tr>
<td>LABEL</td>
<td></td>
<td>An 80-character label for plots and printer output</td>
</tr>
<tr>
<td>MMAX</td>
<td></td>
<td>The minimum number of extrapolations before having HØ.</td>
</tr>
<tr>
<td>NMAX</td>
<td></td>
<td>Number of equations = 9</td>
</tr>
<tr>
<td>OKEGAJ</td>
<td>$\Omega_j$</td>
<td>Precession frequency for the $j^{\text{th}}$ type particle $j = D, T, \alpha, I, HI$</td>
</tr>
<tr>
<td>PALPHA</td>
<td></td>
<td>$\text{ALFAE} + \text{ALFAI} = \text{total power delivered to electrons + ions by } \alpha \text{ particles}$</td>
</tr>
<tr>
<td>PBEAM</td>
<td>$P_{\text{BEAM}}$</td>
<td>(Deuterium) beam power</td>
</tr>
<tr>
<td>Code variable</td>
<td>Analytical expression (if any)</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PBREM</td>
<td>$P_{BREM}$</td>
<td>Power loss due to Bremsstrahlung</td>
</tr>
<tr>
<td>PCOLMB</td>
<td>$P_{e,i}$</td>
<td>Power transfer from electrons to ions by Coulomb collisions</td>
</tr>
<tr>
<td>PE</td>
<td>$P_e$</td>
<td>Electron &quot;pressure&quot; $= \frac{3}{2} n_e T_e$</td>
</tr>
<tr>
<td>PHSYNC</td>
<td>$P_{HSYNC}$</td>
<td>Power loss due to synchrotron radiation (from hot-electron annuli)</td>
</tr>
<tr>
<td>PI</td>
<td>$p_i$</td>
<td>Ion &quot;pressure&quot; $= \frac{3}{2} n_i T_i$</td>
</tr>
<tr>
<td>PLINE</td>
<td>$P_{LINE}$</td>
<td>Power loss due to line radiation</td>
</tr>
<tr>
<td>PLOSS</td>
<td></td>
<td>Total power losses (keV)</td>
</tr>
<tr>
<td>PLTFIL</td>
<td></td>
<td>The plot file name</td>
</tr>
<tr>
<td>PMICRO</td>
<td>$P_\mu$</td>
<td>Specific power delivered to background electrons from microwaves (keV/sec/m$^3$)</td>
</tr>
<tr>
<td>PRAD</td>
<td>$P_{RAD}$</td>
<td>Total power losses due to various radiation processes</td>
</tr>
<tr>
<td>PRTFIL</td>
<td></td>
<td>The print file name</td>
</tr>
<tr>
<td>PRTFRQ</td>
<td></td>
<td>Printer and plotting output frequency in seconds (integration time)</td>
</tr>
<tr>
<td>PSYNC</td>
<td>$P_{SYNC}$</td>
<td>Power loss due to synchrotron radiation (from background electrons)</td>
</tr>
<tr>
<td>Code variable</td>
<td>Analytical expression (if any)</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PT</td>
<td>$p_e + p_i$</td>
<td>Total &quot;pressure&quot; for electrons and ions</td>
</tr>
<tr>
<td>RATE0D</td>
<td>$n_e n_{OD}^{&lt;ov&gt;}_{ioniz}$</td>
<td>Deuterium – neutral ionization rate</td>
</tr>
<tr>
<td>RATE0T</td>
<td>$n_e n_{OT}^{&lt;ov&gt;}_{ioniz}$</td>
<td>Tritium – neutral ionization rate</td>
</tr>
<tr>
<td>RATE0T</td>
<td>$n_{Dn}^{&lt;ov&gt;}_{OT}$</td>
<td>Deuterium – tritium fusion rate</td>
</tr>
<tr>
<td>RCRVRA</td>
<td>$R_C$</td>
<td>Radius of curvature of field = $&lt;\frac{v_B}{B}&gt;$</td>
</tr>
<tr>
<td>RE</td>
<td>$R_e$</td>
<td>Reflection coefficient (used in calculating $P_{SYNC}$)</td>
</tr>
<tr>
<td>RECYLD</td>
<td>$\left(\frac{n_D}{T_{p,D}}\right)^{f_{RD}}$</td>
<td>Deuteron recycling rate</td>
</tr>
<tr>
<td>RECYLT</td>
<td>$\left(\frac{n_T}{T_{p,T}}\right)^{f_{RT}}$</td>
<td>Triton recycling rate</td>
</tr>
<tr>
<td>RMJOR</td>
<td>$R_T$</td>
<td>Major radius (m)</td>
</tr>
<tr>
<td>RMINOR</td>
<td>$a$</td>
<td>Minor radius (m)</td>
</tr>
<tr>
<td>S</td>
<td>Array (in error) of magnitudes of dependent variables for relative error check</td>
<td></td>
</tr>
<tr>
<td>SOBEAM</td>
<td>$S_{DBEAM}$</td>
<td>Deuterium injection rate for fast atoms (beam) $(m^{-3}sec^{-1})$</td>
</tr>
<tr>
<td>SOCOLD</td>
<td>$S_{DCOLD}$</td>
<td>Deuterium injection rate for fast cold fuel $(m^{-3}sec^{-1})$</td>
</tr>
</tbody>
</table>
### Appendix B (continued)

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIGVCX</td>
<td>( &lt;\sigma&gt;_{\text{CX}} )</td>
<td>Maxwellian-averaged charge-exchange cross-section</td>
</tr>
<tr>
<td>SIGVDT</td>
<td>( &lt;\sigma&gt;_{\text{DT}} )</td>
<td>Maxwellian-averaged D-T fusion cross-section</td>
</tr>
<tr>
<td>SIGVIO</td>
<td>( &lt;\sigma&gt;_{\text{ioniz}} )</td>
<td>Maxwellian-averaged ionization cross-section</td>
</tr>
<tr>
<td>SIGVXD</td>
<td>( &lt;\sigma&gt;_{\text{CX-D}} )</td>
<td>Maxwellian-averaged charge-exchange cross-section for D</td>
</tr>
<tr>
<td>SIGVXT</td>
<td>( &lt;\sigma&gt;_{\text{CX-T}} )</td>
<td>Maxwellian-averaged charge-exchange cross-section for T</td>
</tr>
<tr>
<td>SPUTA</td>
<td>( I_{\alpha} )</td>
<td>Sputtering coefficient for alpha particles</td>
</tr>
<tr>
<td>SPUTD</td>
<td>( I_{\text{D}} )</td>
<td>Sputtering coefficient for deuterons</td>
</tr>
<tr>
<td>SPUTHI</td>
<td>( I_{\text{HI}} )</td>
<td>Sputtering coefficient for low-Z impurities (self-sputtering)</td>
</tr>
<tr>
<td>SPUTI</td>
<td>( I_{\text{I}} )</td>
<td>Sputtering coefficient for high-Z impurities (self-sputtering)</td>
</tr>
<tr>
<td>SPUTN</td>
<td>( I_{\text{N}} )</td>
<td>Sputtering coefficient for neutrons</td>
</tr>
<tr>
<td>SPUTT</td>
<td>( I_{\text{T}} )</td>
<td>Sputtering coefficient for tritons</td>
</tr>
<tr>
<td>SRATEA</td>
<td>( \left( \frac{n_{\alpha}}{n_{\text{D},\gamma}} \right) I_{\alpha} )</td>
<td>Sputtering rate for alpha particles</td>
</tr>
<tr>
<td>SRATED</td>
<td>( \left( \frac{n_{\text{D}}}{n_{\text{D},\gamma}} \right) I_{\text{D}} )</td>
<td>Sputtering rate for deuterons</td>
</tr>
<tr>
<td>Code \ variable</td>
<td>Analytical expression (if any)</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SRATEN</td>
<td>$(n_D n_T &lt;\alpha v&gt;<em>{DT} + S</em>{DBEAM} f_D) I_n$</td>
<td>Sputtering rate for neutrons</td>
</tr>
<tr>
<td>SRATET</td>
<td>$(n_T) I_T$</td>
<td>Sputtering rate for tritons</td>
</tr>
<tr>
<td>SSRHIM</td>
<td>$(n_{HI}) I_{HI}$</td>
<td>Self-sputtering rate for high-Z impurities</td>
</tr>
<tr>
<td>SSRIM</td>
<td>$(n_{i}) I_{i}$</td>
<td>Self-sputtering rate for low-Z impurities</td>
</tr>
<tr>
<td>STCOLD</td>
<td>$S_{STCOLD}$</td>
<td>Tritium injection rate for cold fuel ($m^{-3}sec^{-1}$)</td>
</tr>
<tr>
<td>TACOND</td>
<td>$\tau_{E,COND}$</td>
<td>Energy confinement time for conduction</td>
</tr>
<tr>
<td>TACX</td>
<td>$\tau_{CX}$</td>
<td>Charge-exchange time $= \frac{1}{n_{0&lt;\alpha v&gt;CX}}$</td>
</tr>
<tr>
<td>TAE</td>
<td>$\tau_{E,e}$</td>
<td>Energy confinement time for electron</td>
</tr>
<tr>
<td>TAI</td>
<td>$\tau_{E,i}$</td>
<td>Energy confinement time for ion</td>
</tr>
<tr>
<td>TAGLOB</td>
<td>$\frac{p_e + p_i}{\rhoLOSS}$</td>
<td>&quot;Global&quot; confinement time</td>
</tr>
<tr>
<td>TALFAE</td>
<td>$n_D n_T &lt;\alpha v&gt;<em>{DT} (U</em>{\alpha e} - 3T_e)$</td>
<td>Net power delivered from $\alpha$-particles to electrons</td>
</tr>
<tr>
<td>TALFAI</td>
<td>$n_D n_T &lt;\alpha v&gt;<em>{DT} (U</em>{\alpha i} - 3T_i)$</td>
<td>Net power delivered from $\alpha$-particles to ions</td>
</tr>
</tbody>
</table>
## Appendix B (continued)

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAPA</td>
<td>( \tau_{p,\alpha} )</td>
<td>Particle confinement time for alpha particles</td>
</tr>
<tr>
<td>TAPD</td>
<td>( \tau_{p,D} )</td>
<td>Particle confinement time for deuterons</td>
</tr>
<tr>
<td>TAPHIM</td>
<td>( \tau_{p,HI} )</td>
<td>Particle confinement time for high-Z impurities</td>
</tr>
<tr>
<td>TAPIM</td>
<td>( \tau_{p,I} )</td>
<td>Particle confinement time for low-Z impurities</td>
</tr>
<tr>
<td>TAPION</td>
<td>( \tau_{p,i} )</td>
<td>Particle confinement time for ions (weighted average for D and T)</td>
</tr>
<tr>
<td>TAPT</td>
<td>( \tau_{p,i} )</td>
<td>Particle confinement time for tritons</td>
</tr>
<tr>
<td>TAS</td>
<td>( \tau_s )</td>
<td>Spitzer ion-electron momentum exchange time</td>
</tr>
<tr>
<td>TBEAME</td>
<td>( S_{\text{DBEAM}}(U_{\text{DBEAM}}f_{DB,e} + f_{D\alpha}f_{\alpha e}) )</td>
<td>Total power delivered to electrons from D-beam (including ( \alpha )'s)</td>
</tr>
<tr>
<td>TBEAMI</td>
<td>( S_{\text{DBEAM}}(U_{\text{DBEAM}}f_{DB,i} + f_{D\alpha}f_{\alpha i}) )</td>
<td>Total power delivered to ions from D-beam (including ( \alpha )'s)</td>
</tr>
<tr>
<td>TDSTAR</td>
<td>( t^*_D )</td>
<td>Time (seconds) that ( n_{OD} ) takes to reach steady state</td>
</tr>
<tr>
<td>TE</td>
<td>( T_e )</td>
<td>Electron temperature (keV)</td>
</tr>
<tr>
<td>TEF</td>
<td>( T_{ef} )</td>
<td>Final electron temperature desired (keV)</td>
</tr>
<tr>
<td>TI</td>
<td>( T_i )</td>
<td>Ion temperature (keV)</td>
</tr>
<tr>
<td>TIF</td>
<td>( T_{if} )</td>
<td>Final ion temperature desired (keV)</td>
</tr>
</tbody>
</table>
## Appendix B (continued)

<table>
<thead>
<tr>
<th>Code variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLOSS</td>
<td>$3 \frac{n_e T_e}{\tau_{E,e}} = \frac{p_e}{\tau_{E,e}}$</td>
<td>Total losses in $n_T$ equation</td>
</tr>
<tr>
<td>TRANE</td>
<td>$\frac{3}{2} \frac{n_e T_e}{\tau_{E,e}} = \frac{p_e}{\tau_{E,e}}$</td>
<td>Specific power loss from electrons due to transport processes</td>
</tr>
<tr>
<td>TRANI</td>
<td>$\frac{3}{2} \frac{n_i T_i}{\tau_{E,i}} = \frac{p_i}{\tau_{E,i}}$</td>
<td>Specific power loss from ions due to transport processes</td>
</tr>
<tr>
<td>TTSTAR</td>
<td>$t^*_{t}$</td>
<td>Time (seconds) that $n_{0T}$ takes to reach steady state</td>
</tr>
<tr>
<td>UALPHA</td>
<td>$U_\alpha$</td>
<td>Energy of alpha particles = $3.52 \times 10^3$ keV from D-T fusion</td>
</tr>
<tr>
<td>UCR</td>
<td>$U_{CR}$</td>
<td>Critical energy during the slowing down process, at which energy is being transferred equally to plasma ions and electrons</td>
</tr>
<tr>
<td>UDUBEAM</td>
<td>$U_{DBEAM}$</td>
<td>Energy of injected deuterons (keV)</td>
</tr>
<tr>
<td>VOLUME</td>
<td>$V$</td>
<td>Volume of &quot;torus&quot; = $2\pi^2 R_T a^2$</td>
</tr>
<tr>
<td>X</td>
<td>$t$</td>
<td>Integration time (seconds)</td>
</tr>
<tr>
<td>XFINAL</td>
<td>$t_{final}$</td>
<td>Final integration time (seconds)</td>
</tr>
<tr>
<td>XOLD</td>
<td></td>
<td>Previous printout time; do printout if $X$-$XOLD \geq PRTFRQ$</td>
</tr>
<tr>
<td>XSTART</td>
<td>$t_{start}$</td>
<td>Initial integration time (seconds)</td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td>Array of dependent variables for EXTINT</td>
</tr>
</tbody>
</table>
## Appendix B (continued)

<table>
<thead>
<tr>
<th>Code Variable</th>
<th>Analytical expression (if any)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZEFF</td>
<td>$Z_{\text{eff}}$</td>
<td>Effective charge = $\Sigma Z_j^2 n_j / \Sigma Z_j n_j$ $j = D, T, \alpha, I, HI$</td>
</tr>
<tr>
<td>ZHI</td>
<td>$Z_{Hi}$</td>
<td>Charge number of high-Z impurity</td>
</tr>
<tr>
<td>ZI</td>
<td>$Z_I$</td>
<td>Charge number of low-Z impurity</td>
</tr>
</tbody>
</table>
Appendix C

LISTINGS OF FILES

EBT7EQ.IN
EBT7EQ.F4
EXTINT.F4
SIGVDT.DAT
DIS7EQ.F4
DISVEC.CMD
DISTEK.CMD
7EQ.CMD
ETB7EQ.IN - Sample Input File

**35** REACTOR(48 COILS) - PMICRO SHUTOFF - 12/10/76
PRTFIL PRINT.7EQ
PLTFIL PLOT.7EQ
EPS 1.000000E-06
NMAX 4
XSTART 0.000000E+00
H0 1.000000E-06
XFINAL 1.000000E+02
PRTFRO 0.500000E+00
EPS0 1.000000E-05
CIGNIT 0.000000E+00
J1 0.000000E+00
AI 0.000000E+00
ZHI 0.000000E+00
ANI 0.000000E+00
PMICRC 0.200000E+00
UDBEAM 1.000000E+02
UALPHA 3.500000E+03
PBEAM 0.000000E+00
RCURVA 2.700000E+00
RMAJOR 6.000000E+01
RMNOR 1.000000E+00
ERATIO 0.100000E+00
BFIELD 2.500000E+00
DENDTF 0.750000E+20
DENDBF 1.000000E+12
DENATF 1.000000E+12
TEF 1.500000E+01
TIF 1.000000E+01
DEND 0.500000E+20
DENT 0.500000E+20
DENA 1.000000E+15
DENIM 1.000000E+00
DENIM3 1.000000E+00
DENAD 1.000000E+14
DENOT 1.000000E+14
TE 1.000000E+00
TI 1.000000E+00
EBT7EQ.F4 - Point Model Code Source Listing

*****************************************************************************

REACTOR TIME-DEPENDENT POINT MODEL FOR EBT
7 EQUATIONS -- PLOTTING

PHYSICIST: N.A. ICKIN, FUSION ENERGY DIVISION
PROGRAMMER: J.F. ROBERTS, COMPUTER SCIENCES DIVISION
OAK RIDGE NATIONAL LABORATORY
OAK RIDGE, TENNESSEE 37830

*****************************************************************************

MKS UNITS ARE USED:
TIME IS IN SECONDS
DENSITIES ARE IN *xtt**3
POWERS ARE IN MEGAWATTS/M**3
TEMPERATURES ARE IN KEV
ENERGIES ARE IN KEV
PRESSURES ARE IN ******/**3

INTEGRATION IS DONE BY A "DEFERRED-LIMIT INTEGRATOR", CALLED
EXTINT, WRITTEN BY J.P. BORIS AND N.K. WINSOR

EXTINT CALLS A SUBROUTINE CALLED ERROR, WHICH DEFINES
CONVERGENCE CRITERIA. OUR ERROR ROUTINE IS FROM FRED JAEGER.

EXTINT ALSO CALLS A SOURCE SUBROUTINE, F., WHICH SETS UP THE
RIGHT-HAND SIDES OF THE DIFFERENTIAL EQUATIONS.

THE DEPENDENT VARIABLES FOR THE D.E.'S ARE AS FOLLOWS:
Y(1) = DEND = DEUTERON DENSITY
Y(2) = DENT = TRITON DENSITY
Y(3) = DENA = ALPHA PARTICLE DENSITY
Y(4) = DENIM = LOW-Z IMPURITY DENSITY
Y(5) = DENIM = HIGH-A IMPURITY DENSITY
Y(6) = DENBD = DEUTERON NEUTRAL DENSITY
Y(7) = DENOT = TRITON NEUTRAL DENSITY
Y(8) = PE = ELECTRON PRESSURE
Y(9) = PI = ION PRESSURE

OTHER DEPENDENT VARIABLES ARE ALGEBRAIC COMBINATIONS OF
THE DEPENDENT VARIABLES DEFINED BY THE DIFFERENTIAL EQUATIONS
DENE = DEND + DENT + 2*DENAM + 2*IMI + 2*IMH
 = ELECTRON DENSITY
DENI = DEND + DENT = ION DENSITY
DEN0 = DEND + DEN0 = TOTAL NEUTRAL DENSITY
TE = (2/3)*(PE/DENE) = ELECTRON PRESSURE
TI = (2/3)*(PI0/DENI) = ION PRESSURE

IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)
REAL*8 PRTF, TF, IL
EXTERNAL F,ERROR,EXTINT
INTEGER FCOUNT

*****************************************************************************

-------COMMON BLOCKS FOR VARIABLES, INPUT VALUES, % DERIVED EXPRESSIONS

*****************************************************************************
CALL INIT

CALL PRINT1

C------CONVERT PMICRO TO KEV/SEC/T1**3 FOR CALCULATIONS
C
PMICRO=6.242197E+21*PMICRO

C********** MAIN INTEGRATION LOOP **********
C
X=XSTART
XOLD=XSTART
150 FCOUNT=0
C
CALL EXTINT(NMAX,X,Y,F,H0,MMAX,ERROR)
C
CALL GETVAR(Y)
C
C*** IF (TE.LT.1.0) H0=1.0E-5
C*** IF (TI.LT.1.0) H0=1.0E-5
-----PT IS TOTAL "PRESSURE"
TAGLOB IS A GLOBAL CONFINEMENT TIME
ENTAU IS A GLOBAL (ICN) DENSITY*CONFINEMENT TIME

PT=PE+P1
PALPHA=ALFAE+ALFAI
PLLOSS=TRANE+TRAIN+3.0*RATEDT*(TE+TI)*PRA
TAGLOB=PT/PLLOSS
ENTAU=DEN1/TAGLOB

-----CPALP IS TOTAL ALPHA POWER (MEGAWATTS)
CPLOSS IS TOTAL POWER LOST (MEGAWATTS)

CPALP=C*PALPHA
CPLOSS=C*PLOSS

-----REGULATE FREQUENCY OF OUTPUT WITH DELTAX

30 DELTAX=X-XOLD
   IF (DELTAX.LT.PRFRO) GO TO 1000

   CALL PRINT2(X,H0)
   XOLD=X

-----FEEDBACK ON SDCOLD AND STCOLD FOR NEXT TIME-STEP

1000 IF(X.LE.TDSTAR) GO TO 1020

   IF(DENT.GT.DENDTF.AND.DERIV(1).GT.0.0) SDCOLD=0.9*SDCOLD
   IF(DENT.GT.DENDTF.AND.DERIV(1).GT.0.0) H0=1.0E-02
   IF(DENT.LT.D-ENDTF.AND.DERIV(1).LT.0.0) SDCOLD=1.1*ABS(TLOSS)
   IF(DENT.LT.DENDTF.AND.DERIV(1).LT.0.0) H0=1.0E-02

   DEN9D=SDCOLD/(DEN1*SIGVIO)
   Y(6)=DEN9D

1020 IF(X.LE.TTSTAR) GO TO 1030

   IF(DENT.GT.DENDTF.AND.DERIV(2).GT.0.0) STCOLD=0.9*STCOLD
   IF(DENT.GT.DENDTF.AND.DERIV(2).GT.0.0) H0=1.0E-02
   IF(DENT.LT.DENDTF.AND.DERIV(2).LT.0.0) STCOLD=1.1*ABS(TLOSS)
   IF(DENT.LT.DENDTF.AND.DERIV(2).LT.0.0) H0=1.0E-02

   DENAT=STCOLD/(DEN1*SIGVIO)
   Y(7)=DENAT

-----SHUT OFF MICROWAVE POWER AFTER IGNITION

1030 IF(PMICRO.LT.1.0E-06) PMICRO=0.0
IF(TI.GT.TIF.AND.DENI.GT.(DEMTF+DEMTF).AND.
   (ALP-CPLOSS).GT.CIGNIT) PMICRO=0.1*PMICRO

IF(x.LT.xFINAL) GO TO 150

CLOSE FILES

CLOSE(UNIT=22, FILE="EBT7EQ.IN")
CLOSE(UNIT=23, FILE=PRTFIL)
CLOSE(UNIT=28, FILE=PLTFIL)

STOP
END
SUBROUTINE INIT

REAL*3 PRTFIL, PLTFIL

COMMON/VARIABLE/DEEND, DENT, DEMA, DETA, DENIM, DEHIM, DENTD, DENTT,
     PE, PI, DENV, DETA, TE, TI

& COMMON/INPUT/ZI, AI, ZHI, AHI, PMICRO, UDEARN, UALPHA, PBEAM,
     VOLUME, AREA, RCURVA, RMAJOR, RMINOR, ERATIO, BFIELD

& COMMON/SIGMAS, SIGTXT, SIGVIO, SIGVX, SIGVY, SIGVZ

& COMMON/INRATE, SDBETA, SDOLD, STL, DENDT, DENTD, DENTDF, DENTTF, TEF,
     TDSTAR, TSTART, TLOSS, TDSS

& COMMON/IGMAS, SIGV0T, SIGV0, SIGVX, SIGVY, SIGVZ

& COMMON/SIGT, SIGT, TSTOP, TSTART, TSTOP, TSTOP, TSTOP, TDSS, TDSS

& COMMON/INPUT/PE, PI, DENV, DETA, TE, TI

& COMMON/CTRL/DEEND, DENDT, DENTD, DENTT, PE, PI, DENV, DETA, TE, TI

& COMMON/PRTLAB/LABEL(20), PRTFIL, PLTFIL

C-----SET UP PARAMETERS FOR EXTINT AND ERROR ROUTINES

NMAX = 9
DO 5 I = 1, 100
S(I) = 6.1E8
5 Y(I) = 0.0

C-----OPEN INPUT FILE AND READ CONTROL PARAMETERS

OPEN (UNIT=22, FILE='EBT7E9.IN')

READ (22, 40) (LABELCI), (I=1, 20)
READ (22, 20) PRTFIL
READ (22, 20) PLTFIL
READ (22, 20) EPS
READ (22, 20) MMAX
READ (22, 20) EPS
READ (22, 20) XSTART
READ (22, 20) H0
READ (22, 20) XFINAL
READ (22, 20) PRTFRQ

EPS0 IS ERROR CONDITION FOR NEUTRAL EQUATIONS
CIGNIT IS THE CRITERION FOR WHEN (PALPHA - PLOSS) MEANS IGNITION

READ (22, 20) EPS0
READ (22, 20) CIGNIT

20 FORMAT (5X, A10)
40 FORMAT (20A4)
30 FORMAT (9X, E13.6)
60 FORMAT (9X, 12)
---READ INPUT PARAMETERS---

ZI is Low-Z Impurity Charge Number
AI is Low-Z Impurity Atomic Number
ZH is High-Z Impurity Charge Number
AH is High-Z Impurity Atomic Number
PM is The Micro-Wave Power, Input as Megawatts/3
UD is The Deuterium Beam Energy
UAI is The Alpha Particle Beam Energy
PBEAM is The Total Beam Power
VOLUME is The Plasma Volume in the EBT
AREA is The Surface Area
RCURVA is the Radius of Curvature (GRAD(BFIELD)/BFIELD)
RMajor is The Major Radius
RMinor is The Minor Radius
ERAT is The Ratio of the Charge*Radial Electric Field/Area to The Electron Temperature

READ (22.50) ZI
READ (22.50) AI
READ (22.50) ZH
READ (22.50) AH
READ (22.50) PM
READ (22.50) UD
READ (22.50) UAI
READ (22.50) PBEAM
READ (22.50) RCURVA
READ (22.50) RMajor
READ (22.50) RMinor
READ (22.50) ERAT
READ (22.50) BFIELD

READ (22.50) DENDTF
READ (22.50) DENODF
READ (22.50) DENOTF
READ (22.50) TEF
READ (22.50) TIF

---READ INITIAL CONDITIONS---

READ (22.50) DEND
READ (22.50) DENT
READ (22.50) DINA
READ (22.50) DENIM
READ (22.50) DENHIM
READ (22.50) DENOD
READ (22.50) DENOT
READ (22.50) TE
READ (22.50) TI

---CALCULATE OTHER NEEDED QUANTITIES---

DENE = DEND + DENT + 2.0 * DINA + ZI * DENIM + ZH * DENHIM
DEN1 = DEN0D + DEN0T
DEN2 = DEN0D + DEN0T
PE = 1.5 * TE * DEN
PI = 1.5 * TI * DEN

C C------------- TDSTAR, TTSTAR ARE TIMES (SEC) FOR NEUTRALS (D,T RESPECTIVELY)
C TO REACH EQUILIBRIUM --> SET TO INFINITY
C (USED AS CHECK FOR NEUTRAL EQUATIONS IN F)
C
C TDSTAR = 1.00E+05
TTSTAR = 1.00E+05
C
C VOLUME = 2.0 * 9.869684 * MAJOR * MINOR * MINOR
C
C SDBEAM = 6.25E21 * PBEAM / (VOLUME * UDPAM)
SDBEAM = 0.88
C
C C------------- CALL SIGION WITH TE=TEF TO GET SIGVIO TO
C CALCULATE SDCOLD, STCOLD
C
TE = TEF
CALL SIGION
TE = (2.0 * PE) / (3.0 * DEN)

C SDCOLD = 2.0 * DENDTF * (DEN0DF * SIGVIO)
STCOLD = 2.0 * DENDTF * (DEN0TF * SIGVIO)

C C------------ C IS POWER CONVERSION FACTOR FROM KEV/SEC/MM3 TO MEGAWATTS
C (TOTAL POWER, NOT NORMALIZED TO VOLUME)
C
C = 1.602E-22 * VOLUME

C C------------- SET UP Y-ARRAY OF DEPENDENT VARIABLES
C
Y(1) = DEND
Y(2) = DENT
Y(3) = DENA
Y(4) = DENDM
Y(5) = DENHIM
Y(6) = DEN0D
Y(7) = DEN0T
Y(8) = PE
Y(9) = PI

C C------------- OPEN PRINT AND PLOT FILES
C
OPEN(UNIT=23, FILE=PRTFIL)
OPEN(UNIT=2d, FILE=PLTFIL)
C
RETURN
END
SUBROUTINE PRINT1

C WRITE INPUT PARAMETERS AND INITIAL CONDITIONS

COMMON/VARIABLE/DENT. DENT. DENT. DENH. DENH. DENH. DENH.
& PI. PI. PI. PI. PI. PI. PI. PI.
COMMON/INPUT/ZI. ZH. AH. PMICRO. UDBEAM. UALPHA. PBEAM.
& VOLUME. AREA. RCRV. RMAJOR. R. IMOR. ERATIO. BFIELD
COMMON/PARAM/FCOUNT. EPS0
COMMON/ERRCOM/EPS. S(100). Y(100). NMAX. NMAX
COMMON/CONTROL/XSTART. HB. XFINAL. PRTFRQ
COMMON/PRTLAB/LABEL(20). PRTFIL. PLTFIL

C 100 FORMAT(' . 30X,*EBT REACTOR STUDIES ,//)
WRITE('**(23,106)
100 FORMAT(' '16X,'INPUT PARAMETERS:',//)
WRITE(23,106) ZI. ZH. AH. PMICRO
106 FORMAT('.20(AE11.4))'), 20(A(AE11.4))
& WRITE(23,106) UDBEAM. UALPHA. PBEAM. VOLUME. AREA
107 FORMAT(' '12X,'INPUT PARAMETERS:',//)
WRITE(23,107) UDBEAM. UALPHA. PBEAM. VOLUME. AREA
108 FORMAT(' '12X,'INITIAL CONDITIONS:',//)
WRITE(23,120) DENH. DENH. DENH. DENH. DENH. DENH. DENH.
& WRITE(23,120) PI. PI. PI. PI. PI. PI. PI. PI.
120 FORMAT('.14X,'DENT.',E11.4.8X),'DENT.',E11.4.8X.
& WRITE(23,125) DENH. DENH. DENH. DENH. DENH. DENH. DENH.
& WRITE(23,125) PI. PI. PI. PI. PI. PI. PI. PI.
125 FORMAT('.14X,'DENT.',E11.4.8X),'DENT.',E11.4.8X.
& WRITE(23,130) TE. TI
130 FORMAT('.16X,'TE.',E11.4.0X),'TE.',E11.4.0X)
WRITE(23,135)
135 FORMAT(' 'CONTROL PARAMETERS:',//)
WRITE(23,140) XSTART. HB. XFINAL. EPS. NMAX
140 FORMAT('.12X,'XSTART.',E11.4.8X),'XSTART.',E11.4.8X.
& WRITE(23,145) HB. XFINAL. EPS. NMAX
145 FORMAT('.12X,'EPS.',E11.4.0X),'EPS.',E11.4.0X.
& RETURN
END
SUBROUTINE PRINT2(X,HO)

C========PRINT VARIABLES, DERIVATIVES, AND VARIOUS EXPRESSIONS
AS A FUNCTION OF TIME, DURING INTEGRATION

COMMON/VARIE/DEND, DENT, DEMA, DENIM, DENHIM, DENBD, DENBT,
PE, P1, DENE, DEN1, DEN0, TE, TI
COMMON/INPUT/21, A1, ZHI, AHI, PMICRO, UDREAM, UALPHA, PBEAM,
VOLUME, AREA, RCURVA, MAJOR, RMNOR, ERATIO, BFIELD
COMMON/SIGMAT/SIGVOT, SIGVIO, SIGVX, SIGVD, SIGVT
COMMON/TMPE/TAP, TAP1, TAP2, TEPHIM, TAP2ON
COMMON/TIME/TAE, TAE1, TACOND
COMMON/TIME/TAS, TAX
COMMON/FRA/CFB, FDB, FAE, FAI, FRD, FRT, FHL, FD
COMMON/POWER/PSYCH, PHSYCH, PSYCH, PLINE, PBREM, PRAD, PDCLMB
COMMON/INRATE/SDREAM, SDCLM, TDSTAR, TDSTAR, TDCLM, DENDT, DENOBT, DENTBT, TE,
TDSTAR, TDSTAR, DLOSS, DLOSS
COMMON/RATES/RATE1, RATE0, RATE6, DBFUS
COMMON/DIFLOS/DIFD, DIFT, DIP1, DIFM, DIFHIM
COMMON/SPUT/SPUT1, SPUT2, SPUT3, SPUT4, SPUT5
COMMON/ERATES/ERAT1, ERAT0, ERAT6, EARAT1, EARAT0
COMMON/RECYC/RECYC1, RECYC0, RECYC6, RECYC7
COMMON/RATES/RATES1, RATES0, RATES6, SRATES1, SRATES0
COMMON/ARATE/ARATE1, ARATE0, ARATE6, SRATES1, SRATES0
COMMON/ALFA/ALFAM, ALFAN, ALFAS, ALFA1, ALFA0
COMMON/GLOBAL/AZ, AHR, AHI, PMICRO, UDREAM, UALPHA, PBEAM,
VOLUME, AREA, RCURVA, MAJOR, RMNOR, ERATIO, BFIELD
COMMON/CLGG/CCLGG
COMMON/EPS8/EPS8
COMMON/PRTFIL, PLTFIL
COMMON/LAB6/LABEL(28), PRTFIL, PLTFIL

WRITE(28,600) X, DENE, DENT, DEMA, DEN0T, TE, TI, CPALP, CPLS, TDSTAR, TDSTAR, SDREAM
600 FORMAT(92X,'TIME',E11.4,9X)
WRITE(28,800) X, FCOUNT
800 FORMAT(92X,'COUNT',E11.4,9X)
WRITE(28,900) DERIV1, DERIV2, DERIV3, DERIV4, DERIV5
900 FORMAT(92X,'DY(1)',E11.4,9X,'DY(2)',E11.4,9X,'DY(3)',E11.4,9X,
E11.4,9X,'DY(4)',E11.4,9X,'DY(5)',E11.4,9X)
WRITE(28,905) DERIV6, DERIV7, DERIV8, DERIV9
905 FORMAT(92X,'DY(6)',E11.4,9X,'DY(7)',E11.4,9X,'DY(8)',E11.4,9X,
E11.4,9X,'DY(9)',E11.4,9X)
WRITE(28,920) DEND, DENT, DEMA, DEN0T, DENIM, DENBHIM
920 FORMAT(92X,'DEN',E11.4,9X,'DEN',E11.4,9X,'DEN',E11.4,9X,
E11.4,9X,'DEN',E11.4,9X)
WRITE(28,925) DENBD, DENBT, DENE, DEN1, DEN6
925 FORMAT(92X,'DEN',E11.4,9X,'DEN',E11.4,9X,'DEN',E11.4,9X,
E11.4,9X)
WRITE(28,930) TE, TI, PE, PI
930 FORMAT(92X,'TE',E11.4,9X,'TI',E11.4,9X,'PE',E11.4,9X,
'PI',E11.4)
WRITE(28,935) SDREAM, SDCLM, TDSTAR, TTSTAR
935 FORMAT(92X,'SDREAM',E11.4,9X,'SDCLM',E11.4,9X,'TDSTAR',E11.4,9X,
'TTSTAR',E11.4)
WRITE(23.940) RATEDT, RATEBD, RATEBT, BDFUSN
        FORMAT( '12X,'RATEDT'=',E11.4,3X,'RATEBD'=',E11.4,6X,'BDFUSN'=',E11.4)
WRITE(23.945) SIGVDT, SIGVIO, SIGVCX, SIGVX0, SIGVXT
        FORMAT( '12X,'SIGVDT'=',E11.4,6X,'SIGVIO'=',E11.4,6X,'SIGVCX'=',E11.4,6X,'SIGVX0'=',E11.4,6X,'SIGVXT'=',E11.4)
WRITE(23.950) TAPD, TAPT, TAPA, TAPIM, TAPHIM
        FORMAT( '14X,'TAPD'=',E11.4,8X,'TAPT'=',E11.4,8X,'TAPA'=',
        & E11.4,7X,'TAPIM'=',E11.4,6X,'TAPHIM'=',E11.4)
WRITE(23.955) TAPION, TAE, TAEI, TACOND
        FORMAT( '12X,'TAPION'=',E11.4,8X,'TAE'=',E11.4,8X,'TAEI'=',
        & E11.4,6X,'TACOND'=',E11.4)
WRITE(23.960) TAS, TACX
        FORMAT( '15X,'TAS'=',E11.4,8X,'TACX'=',E11.4)
WRITE(23.965) PSYNC, PHSYNC, FHSYNC, PLINE, PBREM, PCOLMB, PRAD, PMICRO
        FORMAT( '13X,'PSYNC'=',E11.4,6X,'PHSYNC'=',E11.4,6X,'FHSYNC'=',E11.4,7X,'PLINE'=',E11.4,7X,'PBREM'=',E11.4,7X,'PCOLMB'=',E11.4,8X,'PRAD'=',E11.4,6X,'PMICRO'=',E11.4)
WRITE(23.970) DBEAME, BFUSNE, TBEAME, TRANE, DBEAMI, BFUSNI, TBEAMI, TPHI
        FORMAT( '12X,'DBEAME'=',E11.4,6X,'BFUSNE'=',E11.4,6X,'TBEAME'=',E11.4,7X,'TRANE'=',E11.4,7X,'DBEAMI'=',E11.4,6X,'BFUSNI'=',E11.4,6X,'TBEAMI'=',E11.4,7X,'TRANI'=',E11.4)
WRITE(23.975) ALFAE, ALFAI
        FORMAT( '13X,'ALFAE'=',E11.4,7X,'ALFAI'=',E11.4)
WRITE(23.980) PALPHA, PLOSS, TAGLOB, ENTAU
        FORMAT( '12X,'PALPHA'=',E11.4,7X,'PLOSS'=',E11.4,6X,'TAGLOB'=',E11.4,7X,' ENTAU'=',E11.4)
WRITE(23.985) CPALP, CPLOSS
        FORMAT( '10X,'PALPHA'=',E11.4,5X,'PLOSS'=',E11.4,5X)
RETURN
END
SUBROUTINE GETVAR(Y)

C DIMENSION Y(1)

COMMON/VARBLE/DEND, DENT, DENA, DENIM, DENHIM, DENBD, DEN0T.
   & PE, PI, DENE, DEN1, DEN2, TE, TI
 COMMON/INPUT/ZI, AI, ZHI, AMI, PHICR, UDBEAM, UALPHA, PBEAM.
   & VOLUME, AREA, RCURVA, RMAJOR, RMINOR, ERATIO, BFIELD

DEND = Y(1)
DENT = Y(2)
DENA = Y(3)
DENIM = Y(4)
DENHIM = Y(5)
DENBD = Y(6)
DEN0T = Y(7)
PE = Y(8)
PI = Y(9)

DENE = DEND + DENT + 2.0 * DENA + 2 * ZI * DENIM + 2 * ZHI * DENHIM
DENI = DEND + DENT
DEN0 = DENBD + DEN0T
TE = (2.0 * PE) / (3.0 * DENE)
TI = (2.0 * PI) / (3.0 * DENI)

RETURN
END
SUBROUTINE F(X,Y,DY)

--- SOURCE SUBROUTINE TO DEFINE THE RIGHT-HAND SIDES OF THE
DIFFERENTIAL EQUATIONS FOR THE PARTICLE-BALANCE MODEL
FOR EBT REACTOR STUDIES ---

X IS THE INDEPENDENT VARIABLE (TIME)
Y(100) IS ARRAY OF DEPENDENT VARIABLES
DY(100) IS ARRAY TO RETURN RIGHT-HAND-SIDES OF
DIFFERENTIAL EQUATIONS TO EXTINT

TIME DERIVATIVES ARE AS FOLLOWS:
DY(1) = D/DT(DEND)
DY(2) = D/DT(DENT)
DY(3) = D/DT(DENA)
DY(4) = D/DT(DENIM)
DY(5) = D/DT(DENHIM)
DY(6) = D/DT(DENBD)
DY(7) = D/DT(DENBT)
DY(8) = D/DT(PE)
DY(9) = D/DT(PI)

DEPENDENT VARIABLES ARE IN Y ARRAY
DEND = Y(1)
DENT = Y(2)
DENA = Y(3)
DENIM = Y(4)
DENHIM = Y(5)
DENBD = Y(6)
DENBT = Y(7)
PE = Y(8)
PI = Y(9)

OTHER DEPENDENT VARIABLES ARE ALGEBRAIC COMBINATIONS OF THE
DEPENDENT VARIABLES DEFINED BY THE DIFFERENTIAL EQUATIONS

IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)
INTEGER FCOUNT

COMMON/VARIABLE/DEND, DENT, DENA, DENIM, DENHIM, DENBD, DENBT,
PE, PI, DENE, DENI, DENO, TE, TI
COMMON/INPUT/ZI, A, ZI, AHI, PMICRO, UDREAM, UALPHA, PBEAM,
VOLUME, AREA, RMAJOR, RMINOR, ERATIO, BFIELD
COMMON/SIGMAS, SIGVT, SIGVI, SIGVCX, SIGVX, SIGVTX
COMMON/TIMEP, TAPD, TAPT, TAPA, TAPIM, TAPHIM, TAPION
COMMON/TIMEE, TAEI, TACOND
COMMON/TIMEX, TAS, TACX
COMMON/FRAC, DBE, DBI, FAI, FRD, FRT, FM, FD
COMMON/POWER, Psync, PSYCN, PSYN, PLINE, PBREM, PRAD, PCOLM
COMMON/INRATE, SDBREAM, SDCOLD, SCDOLD, DENDTF, DENBDF, DENBTF, TEF,
TDSTAR, TTSTAR, DLOSS, TLOSS
CALL GETVAR(y)

CALCULATE NEEDED FUNCTIONAL QUANTITIES (SUBROUTINES)

CALL SIGDT
CALL SIGION
CALL COLOR
CALL TAUS
CALL FUFRAC
CALL SPUTR
CALL FRACT
CALL POURAD

CALCULATE D/DT(DEND)

BDFUSN IS THE RATE OF FUSION OF THE DEUTERIUM BEAM WITH PLASMA TRITONS (/M**3/SEC)
RATEDT IS THE DEUTERIUM-TRITIUM FUSION RATE
RATEBD IS THE DEUTERIUM-NEUTRAL IONIZATION RATE
DIFD IS THE DEUTERIUM PARTICLE DIFFUSION (CONVECTION) LOSS
SDCOLD IS THE DEUTERIUM INJECTION RATE FROM COLD FUEL

BDFUSN=SDBEAM*FD
BDFUSN=0.0
RATEDT=DEND*(DEND*SIGVDT)
RATEDT=DEF=DEF*SIGVIO)
IF (X.GT.TDSTAR) RATEBD=SDCOLD
DIFD=DEND/TAPD
DLOSS=SDBEAM*BDFUSN*RATEDT-DIFD

CALCULATE D/DT(DENT)

RATEBD IS THE TRITIUM-NEUTRAL IONIZATION RATE
DIFT IS THE TRITIUM-PARTICLE DIFFUSION (CONVECTION) LOSS
SDCOLD IS THE TRITIUM-INJECTION RATE FROM COLD FUEL
C
-----CALCULATE D/DT(DENA)
C
DIFA IS THE ALPHA PARTICLE DIFFUSION (CONVECTION) LOSS
300  DIFA=DENA/TAPA
309  DY(2)=RATE8T+TLOSS
C
-----CALCULATE D/DT(DEN8T)
C
SRATEN IS THE SPUTTERING RATE FOR NEUTRONS
SRATED IS THE SPUTTERING RATE FOR DEUTERONS
SRATET IS THE SPUTTERING RATE FOR TRITONS
SRATEA IS THE SPUTTERING RATE FOR ALPHA PARTICLES
DIFIM IS THE LOW-Z IMPURITY PARTICLE DIFFUSION (CONVECTION) LOSS
SSRIM IS THE SELF-SPUTTERING RATE FOR LOW-Z IMPURITIES
C
C**** SRATEN=(RATE8T+BDFUSN)*SPUTN
C**** SRATED=DIDT*SPUTD
C**** SRATET=DIFT*SPUTT
C**** SRATEA=DIFA*SPUTA
C**** DIFIM=DEN8T/TAP8T
C**** SSRIM=DIFIM*SPUTI
C**** 400  DY(4)=SRATEN+SRATED+SRATET+SRATEA+SSRIM-DIFIM
400  DY(4)=0.8
C
-----CALCULATE D/DT(DEN8D)
C
DIFHIM IS THE HI-Z IMPURITY PARTICLE DIFFUSION (CONVECTION) LOSS
SSRHIM IS THE SELF-SPUTTERING RATE FOR HIGH-Z IMPURITIES
C
C**** DIFHIM=DEN8D/TAP8D
C**** SSRHIM=DIFHIM*SPUTHI
C**** 500  DY(5)=SRATEN+SRATED+SRATET+SRATEA+SSRHIM-DIFHIM
500  DY(5)=0.8
C
-----CALCULATE D/DT(DEN80)
C
RECYLD IS THE RATE OF RECYCLED DEUTERONS
C
C**** RECYLD=DIFD*FRD
C**** 600  DY(6)=RECYLD-RATE8D
600  IF (X.GT.TDSTAR) GO TO 630
53

610  DY(6) = STCOLD-RATE0
615  IF (ABS(DY(6)).LT.EPS8) TSTAR = X
620  GO TO 700
630  DY(6) = 0.0

C------CALCULATE D/DT(ENE)

RECPLT IS THE RATE OF RECYCLED TRITONS

C

C RECYLT=DIFFFRT
C
C 700  DY(7) = RECPLT-RATE0
700  IF (X.GT.TTSTAR) GO TO 730
710  DY(7) = STCOLD-RATE0
715  IF (ABS(DY(6)).LT.EPS8) TTSTAR = X
720  GO TO 800
730  DY(7) = 0.0

C------CALCULATE D/DT(PE)

DBEAME IS THE SPECIFIC POWER DELIVERED DIRECTLY TO THE ELECTRONS
   FROM THE DEUTERIUM BEAM
BFUSNE IS THE SPECIFIC POWER DELIVERED DIRECTLY TO THE ELECTRONS
   BY THE 3.5 MEV ALPHA PARTICLES FROM SUPRATHERMAL FUSION
TBEAME IS THE TOTAL POWER DELIVERED TO THE ELECTRONS FROM THE
   DEUTERIUM BEAM AND ALPHA PARTICLES
ALFAE IS POWER DELIVERED FROM THE ALPHA PARTICLES TO ELECTRONS
TALFAE IS NET POWER DELIVERED FROM ALPHA PARTICLES TO ELECTRONS
TRANE IS THE SPECIFIC POWER LOSS FROM THE ELECTRONS DUE TO
   DIFFERENT TRANSPORT PROCESSES

800  DBEAME = SDBEAM*UDBEAM*FDBE
     BFUSNE = BDFUSN*UALPHA*FAE
     TBEAME = DBEAME + BFUSNE
     ALFAE = RATE0*UALPHA*FAE
     TALFAE = ALFAE - 3.8*TTRATEDT
     TRANE = PE/TAEE

890  DY(8) = PREDR + TBEAME + TALFAE - TRANE - P忙碌LY - PRAD

C------CALCULATE D/DT(PI)

DBEAMI IS THE SPECIFIC POWER DELIVERED DIRECTLY TO THE IONS
   FROM THE DEUTERIUM BEAM
BFUSNI IS THE SPECIFIC POWER DELIVERED DIRECTLY TO THE IONS
   BY THE 3.5 MEV ALPHA PARTICLES FROM SUPRATHERMAL FUSION
TBEAMI IS THE TOTAL POWER DELIVERED TO THE IONS FROM THE
   DEUTERIUM BEAM AND ALPHA PARTICLES
ALFAI IS POWER DELIVERED FROM THE ALPHA PARTICLES TO THE IONS
TALFAI IS NET POWER DELIVERED FROM ALPHA PARTICLES TO IONS
TRANI IS THE SPECIFIC POWER LOSS FROM THE IONS DUE TO
   DIFFERENT TRANSPORT PROCESSES
DDBEAM=SDDBEAM*UDDBEAM*FDBI
BFUSN=BDFUSSH*UDALPHA*FAI
TBEAMI=DDBEAMI+BDFUSTI
ALFAI=RATEDT*UDALPHA*FAI
TALFAI=ALFAI-3.0*T1*RATEDT
TRAI=PI/TAEI

DY(9)=TALFAI+TBEAMI+FDCOLMB-TRAI

FCOUNT=FCOUNT+1

DO 950 I=1,9
DEIV(I) = DY(I)

RETURN
END
SUBROUTINE SICDT

--- FIND SIGVDT, THE MAXWELLIAN-AVERAGED DEUTERIUM-TRITIUM
FUSION CROSS-SECTION, BY INTERPOLATING FROM ARRAYS WHERE
CROSS-SECTION VALUES ARE STORED AS A FUNCTION OF THE
ION TEMPERATURE, TI.

COMMON/VARIABLE/DEND,DENT, DEMA, DENIM, DENHIM, DENBD, DENBTD, PE, PI, DENE, DENI, DENO, TE, TI
COMMON/SIGMA/SIGVDT, SIGVIO, SIGVCO, SIGVAD, SIGVXT

DIMENSION TEMP(150), SV(150)
DATA ICALL/1/

IF (TI.GT.1.0) GO TO 5
SIGVDT=2.88E-29*EXP(5.4*TI)
RETURN

GO TO (10,20), ICALL

10--- READ IN ARRAYS OF SIGVDT VERSUS TEMPERATURE FROM
SIGVEQ.DAT THE FIRST TIME ROUTINE IS CALLED
OPEN(UNIT=21, FILE='SIGVDT.DAT')
READ(21,1) DUM
READ(21,2) (TEP(I), SV(I), I=1,141)
1 FORMAT(A4)
2 FORMAT(F25.3, E25.16)
ICALL=2
CLOSE(UNIT=21, FILE='SIGVDT.DAT')

20--- BINARY SEARCH TO FIND TEMP(I) < TI < TEMP(I+1)
     OR TEMP(I+1) - TI
INDEX=0
INC=1
100 MODINC=MOD(INC,2)
INC=INC/2
INC=INC+MODINC
INDEX=INDEX+INC
IF (I.GT.141) I=141
IF (TEMP(I)-TI) 200, 500, 125
125 IF (TEMP(I-1)-TI) 200, 500, 125
IF (INC.EQ.1) GO TO 1000
GO TO 100

200--- CHANGE INDEX TO SEARCH 2ND HALF OF REMAINING TEMPERATURES
IF (TEMP(I+1).GT.TI) GO TO 400
INDEX=I
IF(INC.EQ.1) GO TO 1000
GO TO 100

C TEMP(I-1) < TI < TEMP(I)
C 300 I=I-1
C TEMP(I) < TI < TEMP(I+1)
C 400 SIGVDT=SV(I)+(SV(I+1)-SV(I))*(TI-TEMP(I))/(TEMP(I+1)-TEMP(I))
RETURN
C TEMP(I)=TI
C 500 SIGVDT=SV(I)
RETURN
C IF GET HERE TI IS OUT OF RANGE OF TABLED VALUES
C 1000 WRITE(6,5000) TI
5000 FORMAT(' TEMPERATURE ',E14.7, ' IS NOT IN RANGE FOR SIGVDT')
SIGVDT=0.0
RETURN
END
SUBROUTINE SIGIOH

C-----CALCULATE THE CROSS-SECTION OF IONIZATION, SIGVIO,
C AS A FUNCTION OF TE
C AND THE CROSS-SECTION OF CHARGE-EXCHANGE, SIGVCX,
C AS A FUNCTION OF TI

DIMENSION A(7).C(9)

COMMON/VARBLS/DEND.DENT.DENM.DENIN.DENG0.DENOT.
& PE.PI.DENI.DENO.TE.TI
COMMON/INPUT/Z1.AI.ZHI.AMI.PMICRO.UDEAL.UALPHA.PBEAM
& VOLUME.AREA.RCURVA.RMAJOR.RMINOR.ERATIO.BFIELD
COMMON/SIGMAS/SIGVDT.SIGVIO.SIGVCX.SIGVXD.SIGVXT

DATA A/-8.3173858E+02,8.1143818E+02,-8.3833998E+01,
& 7846692E+00,-7431406E-01,0.0.4153749E-02,-9486967E-04/
DATA C/-8.1841757E+02,8.528295E+01,-2.2291197E+01,
& 8.9758192E-01,-0.179133E-01,8.4954298E-03,
& 0.2174910E-03,-0.2538285E-04,0.0.823051E-06/

B=ALOG(1000.0*TE);
SUM=0.0
DO 100 I=1,7
SUM=SUM+A(I)*(B**(I-1))
100 CONTINUE
SIGVIO=1.88E-6*EXP(SUM)
C

D=ALOG(1000.0*TI)
SUM2=0.0
DO 200 I=1,9
SUM2=SUM2+C(I)*(D**(I-1))
200 CONTINUE
SIGVCX=1.88E-06*EXP(SUM2)
C
RETURN
END
SUBROUTINE COLMB

CALCULATE PCOLMB, THE POWER TRANSFERRED FROM THE ELECTRONS TO THE IONS BY COULOMB COLLISIONS

COMMON/VARIABLE/DEND,DENI,DENA,DEI,DEN0,TE,TI

COMMON/INPUT/ZI,ZHI,AHI,PMicro,UIDBEAM,UALPHA,PBEAM,

VOLUME,AREA,RCURVA.RMAJOR,RMINOR,ERATIO,BFIELD

COMMON/POWER/PSYNC,FPSYNC,FHSYNC,PLINE,PBREM,PRAD,PCOLMB

COMMON/CLOG/COULOG

COULOG IS THE COULOMB LOGARITHM

COULOG=38.4-LOG(SORT(DENE)/TE)

COULOG=17.000

PCOLMB=1.5e-19*DENE*COULOG*(TE-TI)*(0.5*DEND+DENT/3.0)/
      (SORT(TE**3))

RETURN
END
SUBROUTINE TAUS

CALCULATE THE PARTICLE AND ENERGY CONFINEMENT TIMES
(USING KURZHIKHXH'S RESULTS)
AND CHARGE-EXCHANGE AND MOMENTUM-EXCHANGE TIMES

COMMON/VARIABLE/DEND, DENT, DENA, DENIM, DENBD, DENBT,
& PE, PI, DENE, DENT, DENB, TE, TI
COMMON/INPUT/ZI, AI, ZHI, AHI, MICRO, UDBEAH, VALPHA, PBEAM,
& VOLUME, AREA, RCURVA, RMAJOR, RMINOR, ERATIO, BFIELD
COMMON/SISMAS/SIGVDT, SIGVIO, SIGVCX, SIGVAD, SIGVXT
COMMON/TIMEP/TAPD, TAPT, TAPA, TAPIM, TAPHIM, TAPION
COMMON/TIMEE/TAEE, TAEI, TACOND
COMMON/TIMEX/TAX, TACX
COMMON/CLOG/COULOG

SET UP ARRAYS TO CALCULATE PARTICLE CONFINEMENT TIMES
I=1 FOR DEUTERON
I=2 FOR TRITON
I=3 FOR ALPHA PARTICLE
I=4 FOR LOW-Z IMPURITY NUCLEUS
I=5 FOR HIGH-Z IMPURITY NUCLEUS

DIMENSION Z(5), A(5), DEN(5), TAU(5)
EQUIVALENCE (DEND, DEN(1))
EQUIVALENCE (TAPD, TAU(1))
DATA Z/1.0, 1.8, 2.0, 2.8, 4.0/
DATA A/2.0, 3.0, 4.0, 5.0, 6.0/
DATA IFIRST/1/

GO TO (4,5), IFIRST

SET IMPURITY CHARGE NUMBERS AND ATOMIC NUMBERS
THE FIRST TIME THIS ROUTINE IS CALLED

Z(4)=ZI
Z(5)=ZHI
A(4)=AI
A(5)=AHI
IFIRST=2

LOOP CALCULATES PARTICLE CONFINEMENT TIMES
IMAX IS NUMBER OF PARTICLE CONFINEMENT TIMES TO CALCULATE
IMAX=3 FOR 7 EQUATIONS (D, T, A, NEUTRALS - NO IMPURITIES)

IMAX=3

DO 50 I=1, IMAX
COFR=2.218E-18*COULOG*DEN(I)*Z(I)*Z(I)/SQRT(A(I)*TI**3)
ALPHA-Z(I)*ERATIO
C=7.5*(RMAJOR/RCURVA)**2*(1.0+(2.0+ALPHA)*ALPHA)
OMEGAJ=1.8E3*TI*(1.0+ALPHA)/(Z(I)*BFIELD*RCURVA*RMN)
COLOM=COFRJ/OMEGAJ

50
TAU(I) = (C/COLFRJ) * (1.0 + COLOM*COLOM)

50 CONTINUE

C----- TAPION IS THE ION PARTICLE CONFINEMENT TIME
C
TPION1 = DEND/(DENI*TAPD) + DENT/(DENI*TAPT)
TAPION = 1.0/TPION1

C----- TACX IS THE CHARGE EXCHANGE TIME
C
TACXI = (SIGVCX/DENI) * (DEND*DENDI+DENT*DENDT)
TACX = 1.0/TACXI

C----- TACOND IS THE ENERGY CONFINEMENT TIME
C
TACOND = 3.0*TAPION/7.0
TACOND = 0.0

C----- TAS IS THE SPITZER ION-ELECTRON MOMENTUM EXCHANGE TIME
C (NOTE: THIS EXPRESSION FOR DEUTERIUM BEAM)
C
TAS = (0.24E19/DENE)**SQR(TE**3)
TAS = 8.8

C----- TAEI IS THE ION ENERGY CONFINEMENT TIME
C
TAEI1 = TPION1 + TACXI + 1.0/TACOND
TAEI1 = TPION1 + TACXI
TAEI = 1.0/TAEI1

C----- TAEI, THE ELECTRON ENERGY CONFINEMENT TIME, IS TEMPORARILY SET
C EQUAL TO THE ION PARTICLE CONFINEMENT TIME
C
200 TAEI = TAPION

RETURN
END
SUBROUTINE FRACT

--CALCULATE THE FRACTIONS OF ENERGY TRANSFER, PARTICLES RECYCLED, AND RADIATION TRANSFER

COMMON/VARLBE/DEND, DENT, DENA, DENH, DENTM, DENTD, DENBD, DENBT,
& PE, PI, DENE, DENI, DEND, TE, TI
COMMON/INPUT/ZI, AI, ZHI, PHI, PHICRO, UDDEAM, ALPHA, PBEAM,
& VOLUME, AREA, RCURVA, RMAJOR, RMINOR, ERATIO, BFIELD
COMMON/FRAC/FD8E, FD8I, FAE, FAI, FRD, FRT, FH, FD
COMMON/TIMEX/TAS, TACX

FAI IS THE FRACTION OF 3.52 MEV ALPHA ENERGY GIVEN TO THE IONS
FAI = TE/58.0 - 0.37*(TE/58.0)**1.75

FAE IS THE FRACTION OF 3.52 MEV ALPHA ENERGY GIVEN TO THE ELECTRONS
FAE = 1.0 - FAI

UCR IS THE CRITICAL ENERGY DURING THE SLOWING DOWN PROCESS AT WHICH ENERGY IS BEING TRANSFERRED EQUALLY TO THE IONS AND ELECTRONS
UCR = 29.6*TE*(1.0/DENE)*(0.5*DEND+0.3333*DENT+DENA)**(2.0/3.0)

TEMP = (UDDEAM/UCR)*(1.0+0.5*TIMEX/TACX)

FD8I IS THE FRACTION OF INJECTED DEUTERON ENERGY GIVEN DIRECTLY TO THE IONS
FD8I = 0.5*EXP((-0.0916)*TEMP)+0.5*EXP((-0.635)*TEMP)
FD8I = 0.0

FD8E IS THE FRACTION OF INJECTED DEUTERON ENERGY GIVEN DIRECTLY TO THE ELECTRONS
FD8E = 1.0 - FD8I
FD8E = 0.0

FRD IS THE FRACTION OF THE DEUTERONS WHICH ARE RECYCLED (TEMPORARILY SET TO ZERO)
FRD = 0.0

FRT IS THE FRACTION OF THE TRITONS WHICH ARE RECYCLED (TEMPORARILY SET TO ZERO)
FRT = 0.0

FH IS THE FRACTION OF HOT ELECTRON ANOMAL SYNCHROTRON
RADIATION TRANSFERRED TO BACKGROUND ELECTRONS
(TEMPORARILY SET TO ZERO)

FH=0.0
RETURN
END
SUBROUTINE POURAD

---CALCULATE TOTAL POWER LOSS FROM RADIATION, PRAD

IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)

COMMON/VARIABLE/DEND, DENT, DENA, DENIM, DENHM, DENBD, DENBT,
PE, PI, DENE, DEN1, DENB, TE, TI
COMMON/INPUT/ZI, AI, ZHI, AHI, PHI, UDEAM, UDEAM, VALPHA, PBEAM,
& VOLUME, AREA, CURVA, MAJOR, MINOR, ERATIO, BFIELD
COMMON/FRA/FDE, FDB, FAE, FALPHA, FRT, FD, FRD
COMMON/POWER/PSYNC, PHSYNC, FHSYNC, PLINE, PBREM, PRAD, PCOLMB

---ZEFF IS THE EFFECTIVE CHARGE

ZEFF=(DEND+DENT+4.B*DEMA+2I*ZI*ZHI*DENIM+ZHI*ZEMI)/DENE

---PBREM IS THE POWER LOSS DUE TO RAMSTRAHLUNG RADIATION

PBREM=3.BE-2I*ZEFF*DENE*DENE*SRT(TE)

---PLINE IS THE POWER LOSS DUE TO LINE RADIATION

PLINE=2.BD-17*DENHIM/DENE
   PLINE=0.B

---PSYNC IS THE POWER LOSS DUE TO SYNCHROTRON RADIATION
   FROM THE BACKGROUND Plasma
   (RE IS THE REFLECTION COEFFICIENT:
   RE=0.B

   100 PSYNC=1.BE+5*SRT(DENE*(BFIELD**5)/MINOR)*(TE**2.75)*(1.B-RE)*
       (1.B+TE/2B4.8)

---PHSYNC IS THE POWER LOSS DUE TO SYNCHROTRON RADIATION
   FROM THE HOT-ELECTRON ANNULI
   (TEMPORARILY ZERO)

PHSYNC=0.B

---ISYNC IS THE POWER GAIN DUE TO THE SYNCHTRON RADIATION
   FROM HOT-ELECTRON ANNULI TRANSFERRED TO THE
   BACKGROUND ELECTRONS

FHSYNC=FHSYNC+PSYNC
   FHSYNC=0.B

PRAD=PBREM+PSYN,-FHSYNC+PLINE

200 RETURN
END
SUBROUTINE SPUTR

CALCULATE SPUTN, SPUTD, SPUTA, SPUTI, AND SPUTHI, THE SPUTTERING RATES FOR NEUTRON, DEUTERON, TRITON, ALPHA, AND LOW-Z AND HIGH-Z IMPURITIES, RESPECTIVELY

IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)

COMMON/VARBLE/DEND,DENT,DEHA,DEHM,DEHD,DEHD,PE,PI,DEHE,DENI,DENB,TE,DI
COMMON/INPUT/ZI, AI, ZHI, AH, PHI, MICRO, U, BEAM, UALPHA, PBEAM,
     VOLUME, AREA, RMAJOR, RMINOR, ERATIO, BFIELD
COMMON/SPUT/SPUTN, SPUTD, SPUTT, SPUTA, SPUTI, SPUTHI

--TEMPORARILY ALL SPUTTERING RATES ARE SET TO ZERO

SPUTN=0.0
SPUTD=0.0
SPUTT=0.0
SPUTA=0.0
SPUTI=0.0
SPUTHI=0.0

RETURN
END
SUBROUTINE FUFRAC

CALCULATE FD, THE FRACTION OF THE DEUTERIUM BEAM WHICH UNDERGOES SUPRATHERMAL FUSION

IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)

COMMON/VARIABLE/DEND, DENT, DENA, DENIM, DENHIM, DENO, DENOIT, PE, PI, DENE, DENI, DENO, TE, TI
COMMON/INPUT/ZI, AI, ZHI, AMI, PMICRO, UDDEAM, UALPHA, PBEAM
& COMMON/FRA Not A-FDBE, FDBI, FAE, FAI, FRB, FRT, FH, FD

IF (ABS(UDDEAM-150.0).LT.5.0) GO TO 150
IF (ABS(UDDEAM-200.0).LT.5.0) GO TO 200
IF (ABS(UDDEAM-225.0).LT.5.0) GO TO 225

150  FUN=-0.13075+0.27828*TE-0.00926*TE*TE
    IF (TE.GT.15.0) FUN=2.0
    GO TO 500

200  FUN=-0.12098+0.2747*TE-0.00752*TE*TE
    IF (TE.GT.18.0) FUN=2.3
    GO TO 500

225  FUN=-0.13612+0.28177*TE-0.0074*TE*TE
    IF (TE.GT.19.0) FUN=2.5

500  IF (FUN.LT.0.0) FUN=0.0

FD=UDDEAM*DENT*FUN/(1.76E+04*DENE)

RETURN

END
SUBROUTINE EXTINT (NMAX, X, Y, F, HB, M MAX, ERROR)

IMPLICIT REAL (A-H, O-Z), INTEGER (I-H)
INTEGER N MAX, M MAX
REAL X, HB, Y(NMAX)

A DEFERRED-LIMIT INTEGRATOR (J.P. BORIS AND N.K. WIMSOR)

THIS SUBROUTINE INTEGRATES UP TO 100 SIMULTANEOUS FIRST ORDER
ORDINARY DIFFERENTIAL EQUATIONS FROM X TO X + HB BY REPEATED EX-
TRAPOLATIONS ON A MIDPOINT RULE. UP TO 10 EXTRAPOLATIONS MAY BE
REQUESTED BEFORE REDUCTION OF THE INITIAL STEPSIZE IS CARRIED OUT.
(REF. R. BULIRSCH AND J. STDER - NUMERISCHE MATHEMATIK 8.1 (1966)

NMAX THE TOTAL NUMBER OF DEPENDENT VARIABLES BEING INTE-
GRATED, THERE WILL BE ONE FIRST ORDER EQUATION FOR
EACH DEPENDENT VARIABLE.

X THE INDEPENDENT VARIABLE, X IS TREATED AS REAL AND
MONOTONIC DURING THE INTEGRATION STEP. THE VALUE OF
X ON RETURN FROM "EXT INT" CONTAINS THE VALUE WHICH
APPROXIMATELY CORRESPONDS TO THE LENGTH OF THE INTEGRATION ACTU-
ALLY PERFORMED. IF CONVERGENCE HAS BEEN OBSERVED IN
M MAX OF FEWER EXTRAPOLATIONS, X (AT EXIT) = X (AT
ENTRY) + HB. ON ENTRY X MUST BE SET TO THE INITIAL
VALUE OF THE INDEPENDENT VARIABLE FOR THE INTEGRATION
STEP BEING CONTEMPLATED.

Y THE DEPENDENT VARIABLES, EACH DEPENDENT VARIABLE Y(N)
(FOR N = 1, 2, ..., NMAX) IS INTEGRATED FROM X TO X+HB
IF ADEQUATE CONVERGENCE, AS DEFINED BY THE "ERROR"
SUBROUTINE, OCCURS IN M MAX OR FEWER EXTRAPOLATIONS.

F THE DERIVATIVE SUBROUTINE SUPPLIED BY THE USER FOR
HIS PARTICULAR PROBLEM. IT MUST BE OF THE FORM
F (X, Y, DY) WHERE THE ARRAY DY(N) (FOR N = 1, 2, ..., NMAX)
IS RETURNED CONTAINING THE NMAX DERIVATIVES
(DY(N)/DX)(X,Y).

HB IS THE BASIC STEPSIZE OF THE INTEGRATION. IN CONVER-
GENCE OCCURS WITHIN M MAX EXTRAPOLATIONS, X RETURNS
FROM EXT INT WITH THE VALUE X + HB. THE VALUES IN THE
ARRAY Y ARE THE VALUES OF THE DEPENDENT VARIABLES AT
THIS VALUE OF X. IF CONVERGENCE DOES NOT OCCUR, HB IS
HALVED AND THE ENTIRE EXTRAPOLATION PROCEDURE IS
REPEATED AND REPEATED AGAIN UNTIL CONVERGENCE OCCURS.
AN ATTEMPT HAS BEEN MADE TO UTILIZE AS MUCH PREVIOUS-
LY COMPUTED INFORMATION AS POSSIBLE WHEN HB MUST BE
HALVED FOR CONVERGENCE.

M MAX CONTAINS THE NUMBER OF TIMES EXTRAPOLATION IS ATTEM-
PTED BEFORE HB IS HALVED. THIS VALUE WILL VARY WITH
COMPUTER ROUND-OFF ERROR AND WITH THE TYPE AND NUMBER
OF EQUATIONS BEING INTEGRATED. IN ALL CASES, HOWEVER,
ONE SHOULD SPECIFY AN M MAX 'GO' 2. THE STEPSIZE FOR
FUTURE ITERATIONS IS SELECTED AND RETURNED IN HB.
THIS NEW VALUE IS CHOSEN SO THAT CONVERGENCE WILL BE
OBSERVED AT ABOUT THE 0.66*M MAX-TH EXTRAPOLATION.
AS WRITTEN, M MAX.LE.10.
ERROR

IS A SUBROUTINE WHICH IS USED TO DETERMINE THE SATISFACTORY CONVERGENCE OF EACH INDIVIDUAL EQUATION BEING INTEGRATED. AN EXAMPLE IS GIVEN WHICH CORRESPONDS TO THE ERROR CRITERION USED BY BULIRSCH AND STOER.

THE CALLING SEQUENCE IS

ERROR (M, DY, CONV, FINISH) WHERE M IS THE ORDER OF EXTRAPOLATION, DY IS THE VECTOR OF INCREMENTS TO THE DEPENDENT VARIABLES Y, CONV IS A VECTOR OF LOGICALS WHICH ARE TRUE COMPONENTWISE WHEN THE CORRESPONDING DEPENDENT VARIABLE HAS CONVERGED, AND FINISH IS RETURNED TRUE WHEN THE ENTIRE SYSTEM OF EQUATIONS HAS SATISFIED THE CONVERGENCE CRITERION.

LOGICAL LATERL, CONV(100), PREVIN, FINISH
REAL STEPFC, X0, U, SUM, YH, BETA, H, DEN, SQRT2, Y2
INTEGER J, K, L, M, N, LMAX, K ASIDE, PTS, MM, MMAXP
INTEGER KMIN
REAL HM(11), S(11), P(11), YBAR(100,11), YH(100)
REAL Y NEW(100), Y OLD(100), DY(100), DYZ(100), DYOH(100), Y HOLD(7,10,100)

C
C INITIALIZE..100
MMAXP = MMAX + 1
SORTZ = SORT(2.00)
FINISH = .FALSE.
LATERL • .FALSE.
X0 = X
DO 100 N = 1, MMAX
100 YB(N) = Y(N)
LMAX = (MMAX + 1)/2 + 1
CALL F (X0, Y, DYZ)

C
C START A NEW LEVEL..200
204 X = X0 + H0
KMIN = 1
STEPFC = 2.0**(MMAX/3.0+.5)
DO 205 H = 1, MMAX
205 CONV(N) = .FALSE.
IF (.NOT.LATERL .OR. (MMAX.LT.D)) GO TO 203
ELSE BEGIN SHIFTING OLD INFORMATION INTO POSITION..
DO 202 N = 1, MMAX
Y(N) = Y HOLD(2,1,N)
YBAR(N,1) = Y(N)
DO 202 L = 1, LMAX
M2 = MMAX - 2
MMIN = IABS(2*M-3)
DO 282 M2 = MMIN, MMAX
282 Y HOLD(L,M,N) = Y HOLD(L+1, M+2, N)
C
C COMPUTING BETA AND ASSOCIATED QUANTITIES..300
203 H = H0/2
HM(1) = H0/2.0
HM(2) = H0/4.0
HM(3) = H0/6.0
BETA = 0.25/(H0*H0)

C
C EXTRAPOLA TIONS OF HIGHER ORDER EXPANSIONS..400
DO 400 MM = 1, MMAXP
BEGINNING THE LOOP OVER MMAX EXTRAPOLATIONS IN THIS LEVEL..
M = MM - 1
STEPFC = STEPFC/SQRT2
PREVIN = .FALSE.
IF (LATERL.AND.(M.LT.MMAX - 1)) PREVIN = .TRUE.
KASIDE = 2
IF (2*(M/2).EQ.MD KASIDE = 3
L = (M+1)/2 + 1
IF (M.GT.2) HMM(MM) = HM(MM-2)/2
H = HM(MM)
S(MM) = 1 - EXP(-BETA*H
IF (PREVIN) GO TO 563
ELSE GENERATE THE M-TH MIDPOINT INTEGRAL...
DO 484 N = 1, NMIMAX
   Y OLD(N) = YB(N)
   Y NEW(N) = YB(N) + H*DYB(N)
   CALL F (XBM. YNEW, DY)
   FTS = (H*1.000001)/M
   DO 405 K = 2, PTS
     C BEGIN THE MIDPOINT INTEGRATION...
     DO 486 N = 1, NMIX
       U = Y OLD(N) + 2*H*DY(N)
       Y OLD(N) = Y NEW(N)
       CALL F (X8 + K*H, YNEW, DY)
       IF (K.NE.KASIDE).OR.(L.LT.2)) GO TO 465
     ELSE BEGIN PUTTING INFORMATION ASIDE...
     DO 468 N = 1, NMAX
       468 Y HOLD (U, M, N) = (YNEW(N) + YOLD(N) + H*DY(N))/2.8
       L = L - 1
       KASIDE = 2*KASIDE
     C END OF PUTTING INFORMATION ASIDE
     465 CONTINUE
     C END OF THE MIDPOINT INTEGRATION
     C
     C NOW ADVANCE THE DEPENDENT VARIABLES...
     IF (M.GT.0) GO TO 504
     DO 505 N = 1, NMIX
       YBAR(N,1) = (YNEW(N) + YOLD(N) + H*DY(N))/2
     505 Y(N) = YBAR(N,1)
     IF (MMX.EQ.0) GO TO 788
     GO TO 468
     C NOW DETERMINE THE INTERPOLATIONAL POLYNOMIALS...
     IF (STEPFC.LT.1.1) KMIN = KMIN + 1
     DEN = 1
     DO 401 K = KMIN, M
       P(K) = ((H/HM(K))**2)
       DO 410 J = KMIN, M
         IF (J.NE.K) P(K) = P(K)*(S(J) - S(MM))/(S(J) - S(K))
     410 CONTINUE
     401 DEN = DEN - P(K)
     C END DETERMINATION OF THE INTERPOLATIONAL POLYNOMIALS
     DO 500 N = 1, NMAX
     IF (CONV(N)) GO TO 500
     YP = Y(N)
     YM = Y HOLD (I, M, N)
     IF (.NOT.PREVIN) YM = (YNEW(N) + YOLD(N) + H*DY(N))/2.8
     SUM = 0.0
     IF (M.LT.2) GO TO 501
     DO 502 J = KMIN, M
       502 SUM = SUM + (YBAR(N,J) - YP)*P(J)
501   DY(N) = 0.0
      IF (DEN.NE.0.0) DY(N) = ((YM - YP) - SUM)/DEN
      Y(N) = Y* + DY(N)
      YBAR(N, MH) = YM
    500 CONTINUE
    CALL ERROR (M, DY, CONV, FINISH)
    IF (FINISH) GO TO 700
    400 CONTINUE
C
C PREPARE FOR THE NEXT LEVEL IF NECESSARY..600
      LATERL = .TRUE.
      HB = HB/2
      GO TO 204
C
C RETURN..700
   700 HB = HB * STEPFC
        RETURN
END
SUBROUTINE ERROR (M, DY, CONV, FINISH)

THIS VERSION OF THE ERROR ROUTINE CORRESPONDS CLOSELY TO THE
VERSION EMPLOYED BY BULIRSCH AND STÖER. THE DIFFERENCES, HOWEVER,
ARE NOTEWORTHY. THIS VERSION IS CALLED ONLY AFTER ALL UNCONVERGED
DEPENDENT VARIABLES HAVE BEEN EXTRAPOLATED AT ORDER M. THUS AN
EXCESSIVE NUMBER OF SUBROUTINE REFERENCES ARE MADE UNNECESSARY.
NEW VALUES FOR ANY PARTICULAR DEPENDENT VARIABLE ARE NOT CALC-
ULATED AFTER CONVERGENCE, AS DEFINED BY THIS SUBROUTINE. HAS BEEN
OBSERVED FOR THAT VARIABLE. THIS POSSIBLE INSTABILITIES ASSOCIATED
WITH ATTEMPTS AT OVERCONVERGENCE CAN BE ELIMINATED. ONE OF THESE
INSTABILITIES, ARISING FROM RESETTING THE MAGNITUDE VECTOR S AT
EVERY EXTRAPOLATION AS DOES THE B - S PROGRAM, IS ELIMINATED BY
RESETTING S ONLY AFTER THE CORRESPONDING DEPENDENT VARIABLE HAS
CONVERGED ACCORDING TO THE CRITERIA CHOSEN.

IF \( |ABS(Y(N))| > S(N) \) S(N) = \( |ABS(Y(N))| \)

IF (\( M \neq 1 \)) GO TO 1

DO 3 N = 1, NMAX

3 NTIMES(N) = 0

MCONV = 0

1 DO 2 N = 1, NMAX

IF \( (Y(N) \neq 0.0) \) S(N) = \( |ABS(Y(N))| \)

IF (.NOT. (.ABS(DY(N))/S(N) .LT. EPS) .OR. CONV(N)) GO TO 2

NTIMES(N) = NTIMES(N) + 1

IF (NTIMES(N) .EQ. 1) MCONV = MCONV + 1

IF (NTIMES(N) .EQ. 2) CONV(N) = .TRUE.

C****

2 CONTINUE

IF (MCONV.EQ.NMAX) FINISH = .TRUE.

RETURN

END
<table>
<thead>
<tr>
<th>TEMPERATURE. KEV</th>
<th>SIGMAV FOR DT. N=3/SEC (MAR. 24.75)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>0.626999999999999E-26</td>
</tr>
<tr>
<td>1.200</td>
<td>0.195888999999999E-23</td>
</tr>
<tr>
<td>1.400</td>
<td>0.443999999999999E-23</td>
</tr>
<tr>
<td>1.600</td>
<td>0.911000000000000E-25</td>
</tr>
<tr>
<td>1.800</td>
<td>0.167000000000000E-24</td>
</tr>
<tr>
<td>2.000</td>
<td>0.263000000000000E-24</td>
</tr>
<tr>
<td>2.200</td>
<td>0.447000000000000E-24</td>
</tr>
<tr>
<td>2.400</td>
<td>0.672000000000000E-24</td>
</tr>
<tr>
<td>2.600</td>
<td>0.966999999999999E-24</td>
</tr>
<tr>
<td>2.800</td>
<td>0.133999999999999E-23</td>
</tr>
<tr>
<td>3.000</td>
<td>0.181999999999999E-23</td>
</tr>
<tr>
<td>3.200</td>
<td>0.230999999999999E-23</td>
</tr>
<tr>
<td>3.400</td>
<td>0.306999999999999E-23</td>
</tr>
<tr>
<td>3.600</td>
<td>0.396999999999999E-23</td>
</tr>
<tr>
<td>3.800</td>
<td>0.476999999999999E-23</td>
</tr>
<tr>
<td>4.000</td>
<td>0.569999999999999E-23</td>
</tr>
<tr>
<td>4.200</td>
<td>0.706999999999999E-23</td>
</tr>
<tr>
<td>4.400</td>
<td>0.842999999999999E-23</td>
</tr>
<tr>
<td>4.600</td>
<td>0.995999999999999E-23</td>
</tr>
<tr>
<td>4.800</td>
<td>0.116999999999999E-22</td>
</tr>
<tr>
<td>5.000</td>
<td>0.134999999999999E-22</td>
</tr>
<tr>
<td>5.200</td>
<td>0.155999999999999E-22</td>
</tr>
<tr>
<td>5.400</td>
<td>0.176999999999999E-22</td>
</tr>
<tr>
<td>5.600</td>
<td>0.200000000000000E-22</td>
</tr>
<tr>
<td>5.800</td>
<td>0.226000000000000E-22</td>
</tr>
<tr>
<td>6.000</td>
<td>0.253000000000000E-22</td>
</tr>
<tr>
<td>6.200</td>
<td>0.281000000000000E-22</td>
</tr>
<tr>
<td>6.400</td>
<td>0.311999999999999E-22</td>
</tr>
<tr>
<td>6.600</td>
<td>0.343999999999999E-22</td>
</tr>
<tr>
<td>6.800</td>
<td>0.377999999999999E-22</td>
</tr>
<tr>
<td>7.000</td>
<td>0.414999999999999E-22</td>
</tr>
<tr>
<td>7.200</td>
<td>0.451999999999999E-22</td>
</tr>
<tr>
<td>7.400</td>
<td>0.491000000000000E-22</td>
</tr>
<tr>
<td>7.600</td>
<td>0.531000000000000E-22</td>
</tr>
<tr>
<td>7.800</td>
<td>0.573999999999999E-22</td>
</tr>
<tr>
<td>8.000</td>
<td>0.617999999999999E-22</td>
</tr>
<tr>
<td>8.200</td>
<td>0.663999999999999E-22</td>
</tr>
<tr>
<td>8.400</td>
<td>0.709999999999999E-22</td>
</tr>
<tr>
<td>8.600</td>
<td>0.756999999999999E-22</td>
</tr>
<tr>
<td>8.800</td>
<td>0.806999999999999E-22</td>
</tr>
<tr>
<td>9.000</td>
<td>0.856999999999999E-22</td>
</tr>
<tr>
<td>9.200</td>
<td>0.909999999999999E-22</td>
</tr>
<tr>
<td>9.400</td>
<td>0.962999999999999E-22</td>
</tr>
<tr>
<td>9.600</td>
<td>0.102999999999999E-21</td>
</tr>
<tr>
<td>9.800</td>
<td>0.107999999999999E-21</td>
</tr>
<tr>
<td>10.000</td>
<td>0.113999999999999E-21</td>
</tr>
<tr>
<td>12.000</td>
<td>0.174999999999999E-21</td>
</tr>
<tr>
<td>14.000</td>
<td>0.238999999999999E-21</td>
</tr>
<tr>
<td>16.000</td>
<td>0.306999999999999E-21</td>
</tr>
<tr>
<td>18.000</td>
<td>0.376999999999999E-21</td>
</tr>
<tr>
<td>20.000</td>
<td>0.438999999999999E-21</td>
</tr>
<tr>
<td>22.000</td>
<td>0.488999999999999E-21</td>
</tr>
<tr>
<td>24.000</td>
<td>0.538999999999999E-21</td>
</tr>
<tr>
<td>26.000</td>
<td>0.585999999999999E-21</td>
</tr>
<tr>
<td>28.000</td>
<td>0.627999999999999E-21</td>
</tr>
<tr>
<td>30.000</td>
<td>0.664999999999999E-21</td>
</tr>
<tr>
<td>32.000</td>
<td>0.698999999999999E-21</td>
</tr>
<tr>
<td>34.000</td>
<td>0.726999999999999E-21</td>
</tr>
<tr>
<td>Value</td>
<td>Result</td>
</tr>
<tr>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>660.000</td>
<td>0.31100000000000000E-21</td>
</tr>
<tr>
<td>680.000</td>
<td>0.30700000000000000E-21</td>
</tr>
<tr>
<td>700.000</td>
<td>0.30199999999999999E-21</td>
</tr>
<tr>
<td>720.000</td>
<td>0.29800000000000000E-21</td>
</tr>
<tr>
<td>740.000</td>
<td>0.29400000000000000E-21</td>
</tr>
<tr>
<td>760.000</td>
<td>0.29000000000000000E-21</td>
</tr>
<tr>
<td>780.000</td>
<td>0.28600000000000000E-21</td>
</tr>
<tr>
<td>800.000</td>
<td>0.28300000000000000E-21</td>
</tr>
<tr>
<td>820.000</td>
<td>0.27899999999999999E-21</td>
</tr>
<tr>
<td>840.000</td>
<td>0.27599999999999999E-21</td>
</tr>
<tr>
<td>860.000</td>
<td>0.27299999999999999E-21</td>
</tr>
<tr>
<td>880.000</td>
<td>0.26999999999999999E-21</td>
</tr>
<tr>
<td>900.000</td>
<td>0.26799999999999999E-21</td>
</tr>
<tr>
<td>920.000</td>
<td>0.26499999999999999E-21</td>
</tr>
<tr>
<td>940.000</td>
<td>0.26199999999999999E-21</td>
</tr>
<tr>
<td>960.000</td>
<td>0.25999999999999999E-21</td>
</tr>
<tr>
<td>980.000</td>
<td>0.25800000000000000E-21</td>
</tr>
<tr>
<td>1000.000</td>
<td>0.25500000000000000E-21</td>
</tr>
<tr>
<td>1020.000</td>
<td>0.25499999999999999E-21</td>
</tr>
<tr>
<td>1040.000</td>
<td>0.25499999999999999E-21</td>
</tr>
<tr>
<td>1060.000</td>
<td>0.25499999999999999E-21</td>
</tr>
<tr>
<td>1080.000</td>
<td>0.25499999999999999E-21</td>
</tr>
</tbody>
</table>
DIS7EQ.F4 - DISSPLA Plotting Program Source Listing

C PROGRAM TO PLOT VARIABLES VS. TIME (SEMI-LOG)
FROM PLOT DATA FILE FOR A SINGLE RUN OF EBT7EQ.F4
USING DISSPLA -- ON VERSATEK PLOTTER ON PDP-10
(USER NEEDS TO EXECUTE POST-PROCESSOR VECVER)
OR ON TEKTRONIX CRT TERMINAL

DIMENSION LABEL(16)
DIMENSION TIME(58), DENE(58), DENO(58), DENI(58), DEN0(58),
& TE(58), TI(58), CPALPH(58), CPLOSS(58), TAGLO(58),
& ENTAU(58), H0(58), SDCL0D(58), STCOLD(58)
DIMENSION IPAR(48)

DATA IDIM/58/

C------SET PLOTTING PARAMETERS

XINCH=10.0
YINCH=6.0
IMARK=2

C------SET DEFAULT PLOT AXIS LIMITS

DENMIN=1.0E+18
DENMAX=1.0E+21
TMIN=1.0
TMAX=1.0E+18
POUMIN=1.0
POUMAX=1.0E+19
SCMIN=0.0
SCMAX=1.0E+18

C------GET DATA FROM PLOT FILE AND CHECK PLOT AXIS LIMITS

OPEN(UNIT=20, FILE='PLOT.7EQ')
READ(20, 100) (LABEL(I), I=1, 16)
100 FORMAT(16A5)
READ FILE
NPTS=0
DO 250 I=1, IDIM
READ(20, 200, END=260) TIME(I), DENE(I), DENO(I), DENI(I), DEN0(I),
& TE(I), TI(I), CPALPH(I), CPLOSS(I), TAGLO(I),
& ENTAU(I), H0(I), SDCOLD(I), STCOLD(I)
200 FORMAT(1, 1E11.4, ' ', 3E11.4)
NPTS=NPTS+1
IF (DENE(I) LT DENMIN) DENMIN=DENE(I)
IF (DENO(I) LT DENO(I)) DENO(I)=DENE(I)
IF (TE(I) LT TMIN) TMIN=TE(I)
IF (TI(I) LT TMIN) TMIN=TI(I)
IF (CPALPH(I) GT POUMAX) POUMAX=CPALPH(I)
IF (CPLOSS(I) GT POUMAX) POUMAX=CPLOSS(I)
IF (SDCOLD(I) LT SDCMIN) SDCMIN=SDCOLD(I)
IF (STCOLD(I) LT STCOLD(I)) STCOLD(I)=STCOLD(I)
RANGE CHECK
IF(SDCOLD(I).GT.SDCMAX) SDCMAX=SDCOLD(I)

250 CONTINUE
260 CONTINUE

C--------PUT '*' IN LABEL FOR SELF-COUNTING TITLE

DU 300 I=1,16
IF(LABEL(I).EQ."") GO TO 310

300 CONTINUE
310 LABEL(I)="*

C

C********** TEKTRONIX OR VERSATEK? *******************************************

C

ICPS=128
IMODE=0

C*** CALL TKTRN(ICPS,IMODE)
CALL VASTEC

C

C--------PLOT DENSITIES VS. TIME

C

CALL BGNPL(1)
J=LINEST(IPAK,.49,.25)
CALL AXSPLT(TIME(I),TIME(NPTS),XINCH,YOR,XSTEP,XAXIS)
CALL ALGPL(T(ENVMIN,ENVMAX,YINCH,YOR,YCYCLE)
CALL CARTOG
CALL TITLE(LABEL,-100,TIME (SEC),10,DENSITIES (e/M**3),
18,XAXIS,YINCH)
CALL YLOG(XOR,XSTEP,YOR,YCYCLE)
CALL RESET('CARTOG')
BLNKL=XAXIS-1.9
BLNKR=BLNKL+1.5
BLNKB=1.0
BLNKT=BLNKB+1.9
CALL BLNKI(BLNKL,BLNKR,BLNKB,28)
CALL CURVE(TIME,DENC,NPTS,MARK)
CALL CURVE(TIME,DENC,NPTS,MARK)
CALL CURVE(TIME,DENC,NPTS,MARK)
CALL GRID(3)
CALL RESET('BLNKI')
CALL LINES('ELECTRON DENSITY*',IPAK,1)
CALL LINES('ION DENSITY*',IPAK,2)
CALL LINES('ALPHA DENSITY*',IPAK,3)
XLEGND=BLNKL+.8
YLEGND=BLNKB+.8
CALL LEGEND(IPAK,3,XLEGND,YLEGND)
CALL ENDP(1)

C

C--------PLOT TEMPERATURES VS. TIME

C

CALL BGNPL(2)
J=LINEST(IPAK,.49,.25)
CALL ALGPL(T(HIMIN,TMAX,YINCH,YOR,YCYCLE)
CALL CARTOG
CALL TITLE(LABEL,-100,TIME (SEC),10,TEMPERATURES (KEV),18,
XAXIS,YINCH)
CALL RESET('CARTOG')
CALL YLOG(XOR,XSTEP,YOR,YCYCLE)
CALL CURVE(TIME,TI,NPTS,IMARK)
CALL GRID(2,1)
CALL RESET('BLNKB')
CALL LINES('TOTAL ALPHA POWERS',IPAK,1)
CALL CURVE(TIME,SCOLD,NPTS,IMARK)
CALL LINES('TOTAL POWER LOSSES',IPAK,2)
CALL LEGEND(IPAK,2,XLEGND,YLEGND)
CALL ENDPL(2)

C-----PLOT TOTAL POWERS VS. TIME
CALL BGNPL(3)
LINEST(IPAK,40,25)
CALL ALGPLT(POWMIN,POWMAX,YINCH,YOR,YCYCLE)
CALL CARTOG
CALL TITLE(LABEL,-100,'TIME (SEC)',10,'POWERS (MW)',11, XAXIS,YINCH)
CALL RESET('CARTOG')
CALL YLOG(XOR,XSTEP,YOR,YCYCLE)
CALL BLNKR=BLNKL+2.75
CALL BLNKB=BLNKL+3.25
CALL BLNKK=BLNKL+3.5
CALL BLNK1(BLNKL,BLNKR,BLNKB,BLNKT,1)
CALL CURVE(TIME,CPA,T,NPTS,IMARK)
CALL CURVE(TIME,CPLOS,NPTS,IMARK)
CALL GRID(2,1)
CALL RESET('BLNKB')
CALL LINES('TOTAL DEUTERIUM FUELING RATES',IPAK,1)
CALL LINES('TRITIUM FUELING RATES',IPAK,2)
CALL LEGEND(IPAK,2,XLEGND,YLEGND)
CALL ENDPL(3)

C-----PLOT COLD FUELING RATES VS. TIME
CALL BGNPL(4)
LINEST(IPAK,40,25)
CALL ALGPLT(SDCMIN,SDCMAX,YINCH,YOR,YCYCLE)
CALL CARTOG
CALL TITLE(LABEL,-100,'TIME (SEC)',10,'FUELING RATES (•/Tt**3/SEC)',26,XAXIS,YINCH)
CALL YLOG(XOR,XSTEP,YOR,YCYCLE)
CALL RESET('CARTOG')
CALL BLNKL=XAXIS-3.5
CALL BLNKR=BLNKL+3.25
CALL BLNKB=BLNKL+3.5
CALL BLNK1(BLNKL,BLNKR,BLNKB,BLNKT,1)
CALL CURVE(TIME,SCOLD,NPTS,IMARK)
CALL CURVE(TIME,STCOLD,NPTS,IMARK)
CALL GRID(2,1)
CALL RESET('BLNKB')
CALL LINES('DEUTERIUM FUELING RATES',IPAK,1)
CALL LINES('TRITIUM FUELING RATES',IPAK,2)
'LEGND=BLNKL+0.1
CALL LEGEND(IPAK,2,XLEGND,YLEGND)
CALL ENDPL(4)

C-----PLOT TIME-STEP VS. TIME
CALL BGNPL(5)
CALL ALGPLT(TSMIN,TSMAX,YINCH,YOR,YCYCLE)
CALL CARTOG
CALL TITLE(LABEL,-100,'TIME (SEC)'.,10,'TIME-STEP (SEC)'.,15.,
AXIS.XINCH)
CALL RESET('CAPTOG')
CALL YLOG(XOR,YSTEP,YOR,YCYCLE)
CALL CURVE(TIME,HB,NPTS,MARK)
CALL GRID(2,1)
CALL ENDBLI5
C
500 CALL MANCED
C
CLOSE(UNIT=20,FILE="PLOT.PLOT")
C
STOP
END
DISVEC.CMD - Linkage Command File for Versatek Plots

```
DSKC: PLOTS. REL[10, 100], DSKC: PLOT. REL[10, 100], DSKC: VECW. REL[10, 100], DSKC: FFEED. REL[10, 100], DSKC: QGVTEC. REL[10, 100], DSKC: LIB. REL[10, 100] /SEA, DSKC: LIB. REL[10, 100] /SEA, DSKC: LIB. REL[10, 100] /SEA
```

DISTEK.CMD - Linkage Command File for Tektronix Plots

```
```
7EQ.CMD - Batch Command File

.EX EBT7EQ.F4,EXTINT.F4
.IF (NDERROR) .GOTO C
A::.CLOSE
.IF (ERROR) .GOTO C
C::.PRINT PRINT.7EQ/DISPOSE:RENAME
.IF (ERROR) .GOTO D
D::.EX DIS7EQ.F4.@DISVEC.CMD
.IF (ERROR) .GOTO E
E::.CLOSE
.IF (ERROR) .GOTO F
F::.R VECVER
*
.IF (ERROR) .GOTO G
G::.DEL *.BAK,*.REL,*.VEC,*.LPT
XERR::.DEL *.BAK,*.REL,*.VEC,*.LPT
XCERR::.CLOSE
Appendix D

SAMPLE OUTPUT

PRINT.7EQ
PLOT.7EQ
SEGMNT.EPL
PRINT.7EQ - Partial Listing of Printer Output from Sample Run

**OUT NAME**: STATES

**DATE**: 12/18/76

**INPUT PARAMETERS**:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X2</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X3</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X4</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X5</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X6</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X7</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X8</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X9</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X10</td>
<td>0.000000+0</td>
</tr>
</tbody>
</table>

**INITIAL CONDITIONS**:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X2</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X3</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X4</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X5</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X6</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X7</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X8</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X9</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>X10</td>
<td>0.000000+0</td>
</tr>
</tbody>
</table>

**CONTROL PARAMETERS**:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>XSTART</td>
<td>0.000000+0</td>
</tr>
<tr>
<td>XEND</td>
<td>0.100000+0</td>
</tr>
<tr>
<td>XFINAL</td>
<td>0.100000+0</td>
</tr>
<tr>
<td>EPS</td>
<td>0.100000+0</td>
</tr>
<tr>
<td>PRTN</td>
<td>4</td>
</tr>
</tbody>
</table>

**TIME INTEGRATIONS**:

<table>
<thead>
<tr>
<th>Time</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.510000+0</td>
<td>1.370000+0</td>
</tr>
<tr>
<td>1.200000+0</td>
<td>0.130000+1</td>
</tr>
<tr>
<td>1.890000+0</td>
<td>0.212000+0</td>
</tr>
<tr>
<td>2.580000+0</td>
<td>0.311000+0</td>
</tr>
<tr>
<td>3.270000+0</td>
<td>0.412000+0</td>
</tr>
<tr>
<td>3.960000+0</td>
<td>0.511000+0</td>
</tr>
<tr>
<td>4.650000+0</td>
<td>0.611000+0</td>
</tr>
<tr>
<td>5.340000+0</td>
<td>0.711000+0</td>
</tr>
<tr>
<td>6.030000+0</td>
<td>0.811000+0</td>
</tr>
<tr>
<td>6.720000+0</td>
<td>0.911000+0</td>
</tr>
</tbody>
</table>

**PRTN** = 13
<table>
<thead>
<tr>
<th>429</th>
<th>0.311E+16</th>
<th>0.104E+01</th>
<th>0.130E+16</th>
<th>0.372E+14</th>
<th>0.273E+01</th>
<th>0.255E+01</th>
<th>0.130E+01</th>
<th>0.597E+02</th>
<th>0.153E+01</th>
<th>0.136E+01</th>
</tr>
</thead>
<tbody>
<tr>
<td>36</td>
<td>0.298E+16</td>
<td>0.234E+21</td>
<td>0.182E+16</td>
<td>0.832E+14</td>
<td>0.347E+01</td>
<td>0.394E+01</td>
<td>0.124E+01</td>
<td>0.174E+01</td>
<td>0.179E+01</td>
<td>0.179E+01</td>
</tr>
<tr>
<td>70</td>
<td>0.357E+16</td>
<td>0.103E+21</td>
<td>0.307E+14</td>
<td>0.677E+01</td>
<td>0.426E+02</td>
<td>0.140E+02</td>
<td>0.137E+02</td>
<td>0.282E+02</td>
<td>0.225E+02</td>
<td>0.225E+02</td>
</tr>
<tr>
<td>90</td>
<td>0.293E+16</td>
<td>0.234E+21</td>
<td>0.233E+14</td>
<td>0.233E+01</td>
<td>0.368E+01</td>
<td>0.264E+02</td>
<td>0.190E+02</td>
<td>0.225E+02</td>
<td>0.267E+02</td>
<td>0.267E+02</td>
</tr>
<tr>
<td>120</td>
<td>0.327E+16</td>
<td>0.233E+21</td>
<td>0.233E+14</td>
<td>0.394E+01</td>
<td>0.432E+02</td>
<td>0.163E+02</td>
<td>0.242E+02</td>
<td>0.288E+02</td>
<td>0.303E+02</td>
<td>0.303E+02</td>
</tr>
<tr>
<td>150</td>
<td>0.312E+16</td>
<td>0.234E+21</td>
<td>0.315E+14</td>
<td>0.640E+01</td>
<td>0.306E+01</td>
<td>0.177E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
</tr>
<tr>
<td>180</td>
<td>0.356E+16</td>
<td>0.234E+21</td>
<td>0.233E+14</td>
<td>0.347E+01</td>
<td>0.426E+02</td>
<td>0.190E+02</td>
<td>0.225E+02</td>
<td>0.267E+02</td>
<td>0.267E+02</td>
<td>0.267E+02</td>
</tr>
<tr>
<td>210</td>
<td>0.293E+16</td>
<td>0.234E+21</td>
<td>0.233E+14</td>
<td>0.233E+01</td>
<td>0.368E+01</td>
<td>0.264E+02</td>
<td>0.190E+02</td>
<td>0.225E+02</td>
<td>0.267E+02</td>
<td>0.267E+02</td>
</tr>
<tr>
<td>240</td>
<td>0.312E+16</td>
<td>0.234E+21</td>
<td>0.315E+14</td>
<td>0.640E+01</td>
<td>0.306E+01</td>
<td>0.177E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
</tr>
<tr>
<td>270</td>
<td>0.327E+16</td>
<td>0.233E+21</td>
<td>0.233E+14</td>
<td>0.394E+01</td>
<td>0.432E+02</td>
<td>0.163E+02</td>
<td>0.242E+02</td>
<td>0.288E+02</td>
<td>0.303E+02</td>
<td>0.303E+02</td>
</tr>
<tr>
<td>300</td>
<td>0.312E+16</td>
<td>0.234E+21</td>
<td>0.315E+14</td>
<td>0.640E+01</td>
<td>0.306E+01</td>
<td>0.177E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
</tr>
<tr>
<td>330</td>
<td>0.327E+16</td>
<td>0.233E+21</td>
<td>0.233E+14</td>
<td>0.394E+01</td>
<td>0.432E+02</td>
<td>0.163E+02</td>
<td>0.242E+02</td>
<td>0.288E+02</td>
<td>0.303E+02</td>
<td>0.303E+02</td>
</tr>
<tr>
<td>360</td>
<td>0.312E+16</td>
<td>0.234E+21</td>
<td>0.315E+14</td>
<td>0.640E+01</td>
<td>0.306E+01</td>
<td>0.177E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
<td>0.239E+03</td>
</tr>
</tbody>
</table>

PLOT.7EQ - Partial Listing of Intermediate Plot File from Sample Run
SEGMENT.EPL - Versatek Printer Plots from DIS7EQ.F4 from Sample Run