ENDF Utility Codes Release 6.13

C. L. Dunford

May 20, 2002

The present release of the ENDF utility codes, version 6.13, supersedes all previous versions of the codes and this document supersede all previous documentation. The programs are written to process ENDF-6 formatted files including all formats approved up to and including the November 2001 CSEWG meeting except for the Generalized R-Matrix resonance region format and the generalized format for covariances (file 30). The programs will also process ENDF-5 formatted files.

The distributed version of the programs, STANEF, CHECKR, FIZCON, PSYCHE, and INTER have been modernized to conform to the Fortran-95 standard. The GETMAT, LISTEF, PLOTEF and SETMDC meet the Fortran-95 standard but have not been mordernized to incorporate the new features. Support for GETMAT, LISTEF, and PLOTEF will be discontinued when ENDF-7 formats are adopted. The basic codes are designed for batch mode operation as were previous released versions. Input records consist of free format data fields with each field except the last delimited by a comma. Each program can process more than one input file (ENDF tape) per run with the user supplying the input file specifications appropriate to his computer. Standard processing options can be selected by specifying a completely blank option specification record.

In addition to the standard batch mode versions of the programs, there are three interactive versions, one for WINDOWS using Digital Visual Fortran, one for OpenVMS and one for UNIX using the Lahey Fortran Compiler. The these versions prompt the user for input data instead of reading it from a file.

Finally, versions of CHECKR, FIZCON, PSYCHE, STANEF and INTER are available which have a graphical interface. The WINDOWS versions have been created with Digital Visual Fortran and the UNIX versions with Lahey Winteracter. This interface consists of a dialog box for supplying the input data. The program output is displayed in a window. The programs, CHECKR, FIZCON, PSYCHE and STANEF have an additional windows for editing the input file.

All versions are based on a single Fortran source code for each program. A small computer program, SETMDC, is distributed with the programs which can convert the source code from one version to another. The standard batch mode version is indicated by "ANS" code, the interactive versions by "VMS", "WIN" and "UNX"; the versions with a graphical interface with "DVF" and "LWI" codes. The versions with a graphical interface require numerous additional source files which depend on the WINDOWS or UNIX program development environment. These source files will not be distributed except on special request. WINDOWS and UNIX versions of the ENDF Utility codes with a graphical interface will be distributed as executables.

List of Codes in the ENDF 6.13 Utility Package

SETMDC	—	Convert source code between batch and interactive modes
GETMAT	—	Retrieve materials from an ENDF format file
STANEF	—	Creates directory, adds tape label and converts numeric fields
		Converts to binary format
CHECKR	—	Format checking program
FIZCON	—	Procedures and simple physics checking program
PSYCHE	—	More complicated physics checking program
INTER	—	Calculates selected cross sections and integrals
LISTEF	—	Generates file summary and annotated data listing
PLOTEF	—	Generates simple data plots

This page intentionally blank

SETMDC

SETMDC is a small Fortran program written to ANSI-77 standards which is designed to convert a computer program source for use on different computers. Within the source code are specially formatted comment cards which flag sections of machine dependent code. The code is sandwiched between a comment card `C+++MDC+++` and a card `C---MDC---`. Code for a specific machine is initiated by a comment card `C...machine_1, machine_2, etc`. For Fortran-90/95 code replace the C with an ! (e.g. `!+++MDC+++`).

Input Requirements:

The user must supply the following information repeated for each input file to be processed.

RECORD 1 — Input file specification
RECORD 2 — Output file specification
RECORD 3 — Operating system selection
  ANS – batch mode program, standard distribution
  VMS – interactive mode program for VMS
  DVF – interactive mode program for PC
  UNX – interactive mode program for Unix
RECORD 4 — Fortran standard
  F77 – code conforms to Fortran-77 standard
  F95 – code conforms to Fortran-90/95 standard
  –

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a blank record.

Sample Input:

This sample input will convert the STANEF source code as distributed to the interactive OpenVMS version.

	Record Type
`STANEF.BAT`	(1)
`STANEF.FOR`	(2)
`VMS`	(3)
`F95`	(4)
(blank record)	end of run

This page intentionally blank

GETMAT

GETMAT is designed to retrieve one or more materials from an ENDF formatted data file. The output file will contain only the selected materials.

Fortran Logical Units Used:

5	—	Input
6	—	Output
20	—	Input data file, ENDF format
21	—	Output data file, ENDF format

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1	—	Input file specification
RECORD 2	—	Output file specification
RECORD 3	—	Tape label number (up to 4 digits). $\leq$0 means tape has no label $>$0 means the value is used as the tape label number
RECORD 4	—	Up to 66 character text for ENDF tape label (use only if tape label number is greater than zero)

The materials to be extracted are specified on one or more records with the following contents. They need not be given in material order number although the output file will be ordered that way. Material specification is terminated by a blank record.

RECORD(S) 5	—	One or two material numbers in integer format One number to give the material number of the evaluation needed. Two numbers to specify a range of material numbers.

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query.

Sample Batch-mode Input:

	Record Type
`TAPE.603`	(1)
`NEW.20`	(2)
`20`	(3)
`Sample material extraction from TAPE 603`	(4)
`2800,2865`	(5) range
`2925`	(5) single
`325,330`	(5) range
(blank record)	(5) end
`TAPE.702`	(1)
`NEW.40`	(2)
`0`	(3)
`9440`	(5) single
(blank record)	(5) end
`DONE`	end of run

STANEF

STANEF is designed to perform bookkeeping operations on a data file containing one or more material evaluations in ENDF format. These operations include:

1. Creation or modification of a "tape ID" record,
2. Creation or update of the directory in MT=451,
3. Create or modify special hollerith ID records in MT=451 (ENDF-6 only),
4. Resequencing,
5. Conversion of integer and floating point fields to standard format,
6. Creation of a binary (ENDF alternate format) file.

Fortran Logical Units Used:

5	—	Input
6	—	Output
20	—	Input data file, ENDF format
21	—	Output data file, ENDF format
22	—	Temporary storage

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1 — Input file specification
RECORD 2 — Output file specification
RECORD 3 — Tape label
FIELD 1 – Tape label number (integer up to 4 digits)
<0 means tape has no label
$=0$ means copy old label
>0 means the value is used as the tape label number
FIELD 2 – Up to 66 character text for output file ENDF tape label

RECORD 4 — Options selection (4 fields)
FIELD 1 – Mode of output file (integer)
0 – Character format output
1 – Binary format (fields 2 and 3 ignored)
FIELD 2 – Update directory and special hollerith (character)
Y – execute this option
N – do not execute this option
FIELD 3 – Numeric field standardization (character)
Y – execute this option
N – do not execute this option
FIELD 4 – Numeric field standardization FOR 2/151(character)
Y – execute this option
N – do not execute this option

If RECORD 4 is left entirely blank, then the "standard" options are executed. Those are character format output file, update MT=451 and standardize numeric fields.

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query; in graphical mode, via a dialog box.

Sample Batch-mode Input:

	Record Type
`RAW.603`	(1)
`INDEX.603`	(2)
`0`	(3)
`0,Y,N`	(4)
`RAW.610`	(1)
`TAPE.610`	(2)
`610,ENDF/B-VI FISSION PRODUCT TAPE 2`	(3)
(blank record)	(4) default
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `STANEF :== $STANEF.EXE`, has been defined, then some of the input data may be given on the command line as follows

`STANEF` *input_file/output_file/standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. If no output file is given, the output file specification is assumed to be the same as the input file. Any tape label is copied unchanged. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.

This page intentionally blank

CHECKR

CHECKR is a program for checking that an evaluated data file conforms to the ENDF format. It can recognize the difference between ENDF-6 and ENDF-5 formats and performs its tests accordingly. Integer control fields are checked to see that ENDF/B procedural limits on those fields are not violated. To the extent possible, fatal format errors are trapped to prevent unwanted termination of the program. Any file which passes through CHECKR without error messages fully conforms to the ENDF format. CHECKR will now operate on a file which has not been processed with STANEF. This has been done to facilitate processing by eliminating error messages in CHECKR which would be automatically fixed by STANEF which requires that the input file be in legal ENDF format.

Fortran Logical Units Used:

5 — Input
6 — Output
20 — Input data file, ENDF format
21 — Message file for program checking results

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1 — Input file specification
RECORD 2 — Message file specification
(if blank, messages go to standard output file on unit 6)
RECORD 3 — Options selection (2 fields)
FIELD 1 – Material number where processing starts (integer)
(If zero, then checking begins with the first material)
FIELD 2 – Material number where processing ends (integer)
(If zero, then checking continues to end of the file)

If RECORD 3 is left entirely blank, then the entire input file is processed.

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query; in graphical mode, via a dialog box.

Sample Batch-mode Input:

	Record Type
`TAPE.603`	(1)
(blank record)	(2) output to unit 6
`5200,5642`	(3)
`TAPE.610`	(1)
`CHECKR.610`	(2)
(blank record)	(3) default
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `CHECKR :== $CHECKR.EXE`, has been defined, then some of the input data may be given on the command line as follows

`CHECKR` *input_file/ output_file/ standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.

FIZCON

FIZCON is a program for checking that an evaluated data file has valid data and conforms to recommended procedures. It can recognize the difference between ENDF-6 and ENDF-5 formats and performs its tests accordingly. Some of the tests performed include

1. data arrays are in increasing energy order,
2. resonance parameter widths add up to the total,
3. Q-values are reasonable and consistent,
4. no required sections are missing and all cover the proper energy range,
5. secondary distributions are normalized to 1.0,
6. energy conservation in decay spectra.

Optional tests can be performed to check that redundant cross sections such as the inelastic cross section has an energy grid which is the union of all its components and the the cross section values are the sum of the component values at each energy (SUMUP test). Also optionally, algorithms are used to check for possible incorrect entry of data values (Deviant Point test). It is assumed the the file being checked has passed the CHECKR program without any errors being detected.

Fortran Logical Units Used:

5	—	Input
6	—	Output
20	—	Input data file, ENDF format
21	—	Message file for program checking results
22 & 23	—	Temporary paging files for large data arrays
24 & 25	—	Temporary files for the SUMUP tests

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1	—	Input file specification
RECORD 2	—	Message file specification (if blank, messages go to standard output file on unit 6)

RECORD 3 — Options selection (5 fields)

FIELD 1 – Material number where processing starts (integer)
(If zero, then checking begins with the first material)

FIELD 2 – Material number where processing ends (integer)
(If zero, then checking continues to end of the file)

FIELD 3 – Deviant point test control (character)
- Y – Do the test
- N – Do not do the test

FIELD 4 – SUMUP test control (character)
- Y – Do the test
- N – Do not do the test

FIELD 5 – Fractional acceptable difference (real)
The floating point number entered here represents the maximum fractional difference tolerated in an equality test such as a SUMUP test. The default value if none is entered is .001 (1/10 of a percent).

If RECORD 3 is left entirely blank, then the"standard" options are executed. Those are to process the entire input file, to omit the SUMUP and Deviant Point tests and to assume a allowed fractional error of .001.

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query; in graphical mode, via a dialog box.

Sample Batch-mode Input:

	Record Type
`TAPE.603`	(1)
(blank record)	(2) output to unit 6
`5200,5642,N,Y,.002`	(3)
`TAPE.610`	(1)
`FIZCON.610`	(2)
(blank record)	(3) default
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `FIZCON :== $FIZCON.EXE`, has been defined, then some of the input data may be given on the command line as follows

`FIZCON` *input_file/ output_file/ standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.

This page intentionally blank

PSYCHE

PSYCHE is a program for checking the physics content of an evaluated data file. It can recognize the difference between ENDF-6 and ENDF-5 formats and performs its tests accordingly. The present version checks for energy conservation for emitted neutrons and photons, checks Wick's limit for elastic scattering, analyzes resonance parameter statistics, calculates thermal cross sections and resonance integrals, examines continuity across resonance region boundaries and checks "Q" values against mass tables. It is assumed the the file being checked has passed the CHECKR program without any errors being detected.

Fortran Logical Units Used:

5	—	Input
6	—	Output
20	—	Input data file, ENDF format
21	—	Message file for program checking results
22 - 25	—	Temporary files for energy conservation tests

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1 — Input file specification
RECORD 2 — Message file specification
(if blank, messages go to standard output file on unit 6)
RECORD 3 — Options selection (2 fields)
FIELD 1 – Material number where processing starts (integer)
(If zero, then checking begins with the first material)
FIELD 2 – Material number where processing ends (integer)
(If zero, then checking continues to end of the file)

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query; in graphical mode, via a dialog box.

Sample Batch-mode Input:

	Record Type
`TAPE.603`	(1)
(blank record)	(2) output to unit 6
`5200,5642,`	(3)
`TAPE.610`	(1)
`PSYCHE.610`	(2)
(blank record)	(3) default
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `PSYCHE :== $PSYCHE.EXE`, has been defined, then some of the input data may be given on the command line as follows

`PSYCHE` *input_file*/ *output_file*/ *standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.

INTER

INTER is a program for calculating thermal cross sections, g-factors, resonance integrals, fission spectrum averaged cross sections and 14.0 Mev (or other energy) cross sections for major reactions in an ENDF-6 or ENDF-5 format data file. To operate properly, the cross sections must be given pointwise in File 3 and have been linearized. Therefore evaluations containing resonance parameters must first be processed by a code such as RECENT to produce a complete pointwise data file and interpolation codes which are not constant or linear-linear must be processed by a code such as LINEAR.

Fortran Logical Units Used:

5	—	Input
6	—	Output
20	—	Input data file, ENDF format
21	—	Output file for program results

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1 — Input file specification
RECORD 2 — Message file specification
(if blank, messages go to standard output file on unit 6)
RECORD 3 — Options selection (7 fields)

- FIELD 1 – Material number where processing starts (integer)
  (If zero, then checking begins with the first material)
- FIELD 2 – Material number where processing ends (integer)
  (If zero, then checking continues to end of the file)
- FIELD 3 – Thermal calculation control (character)
  - Y – Calculate thermal cross sections and g-factors
  - N – Do not do the calculations
- FIELD 4 – Resonance integral calculation control (character)
  - Y – Calculate resonance integrals
  - N – Do not do the calculations
- FIELD 5 – Fission spectrum calculation control (character)
  - Y – Calculate fission spectrum averages
  - N – Do not do the calculation

FIELD 6 – Single energy (real)
The energy entered here in eV is used to calculate and list the cross sections at any single required energy, for example, at 14.0 Mev.

FIELD 7 – Fractional error (real)
The floating point number entered here is the fractional error which will be tolerated when integrating between two points. The default value is .001.

If RECORD 3 is left entirely blank, then defaults for all values are assumed which are to process all materials, calculate both thermal cross sections and resonance integrals, calculate 14.0 Mev cross sections and allow a fractional error of .001. Neither of the following two records are required and the default parameters described below are assumed.

RECORD 4 — Thermal calculation parameters (3 fields)
(If thermal calculation is not done, omit this record)

FIELD 1 – Thermal energy (real)
The energy entered here in eV is used to calculate and list the Maxwellian averaged cross sections and g-factors, for example, at 0.0253 eV.

FIELD 2 – Lower limit for Maxwellian integral(eV) (real)

FIELD 3 – Upper limit for Maxwellian integral(eV) (real)

If RECORD 4 is left entirely blank, then defaults for all values are assumed which are a thermal energy of 0.0253 eV and integration between .01 mV and 10.0 eV.

RECORD 5 — Resonance integral calculation parameters (2 fields)
(If resonance calculation is not done, omit this record)

FIELD 1 – Lower limit for Resonance integral(eV) (real)

FIELD 2 – Upper limit for Resonance integral(eV) (real)

If RECORD 5 is left entirely blank, then defaults for all values are assumed which are integration between .5 eV and 100.0 keV.

RECORD 6 — Fission spectrum calculation parameters (3 fields)
(If fission spectrum calculation is not done, omit this record)

FIELD 1 – Fission spectrum temperature(real)
The temperature entered here is in eV and is used as the temperature in a Maxwellian distribution, for example, 1.3E+6 eV.

FIELD 2 – Lower limit for fission integral(eV) (real)

FIELD 3 – Upper limit for fission integral(eV) (real)

If RECORD 6 is left entirely blank, then defaults for all values are assumed which are a temperature of 1.02E+6 eV and integration between 1.0 keV and 20.0 Mev.

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query; in graphical mode, via a dialog box.

Sample Batch-mode Input:

	Record Type
`TAPE.639`	(1)
`VTOCU.INT`	(2)
`2300,2931,Y,Y,Y,14.+6,.001`	(3)
(blank record)	(4) default
`1.,100.+3`	(5)
`1.12E+6,1.E+3,20.E+6`	(6)
`TAPE.610`	(1)
`FISSPR.610`	(2)
(blank record)	(3) default
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `INTER :== $INTER.EXE`, has been defined, then some of the input data may be given on the command line as follows

`INTER` *input_file/output_file/standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.

This page intentionally blank

LISTEF

LISTEF is a program designed to produce summary and annotated listings of a data file in either ENDF-6 or ENDF-5 format.

Fortran Logical Units Used:

5	—	Input
6	—	Output
20	—	Input data file, ENDF format
21	—	Summary file
22	—	Annotated listing file
23	—	Temporary file for paging large data arrays

Input Requirements:

In batch mode operation, the user must supply the following control information repeated for each input file to be processed.

RECORD 1 — Input file specification
RECORD 2 — Summary file specification
(if blank, summary goes to standard output file on unit 6)
RECORD 3 — Annotated listing file specification
(if blank, no annotated listing is generated)
RECORD 4 — Options selection (2 fields)
FIELD 1 – Output paging control for annotated listing (character)
Y – New page for each section (MT)
N – No new page for each section (MT)
FIELD 2 – Material processing control (character)
Y – List the entire file
N – List only selected portions of the file

If RECORD 4 is left entirely blank, then default values are assumed which are to process all materials and start each section on a new page of the annotated listing. If only selected materials and files are to be processed, they are specified on one or more records with the following contents. They need not be given in material order number although the output file will be ordered that way. Material and file specification are terminated by a blank record.

RECORD(S) 5 — Selection of materials and files to be processed

FIELD 1 – Material number or ZA number

Material number must be 1 to 4 digits with no period.

ZA number must be a 4 or 5 digit number with a trailing period having the value 1000.*Z + A for the desired material.

FIELD(S) 2 thru 20 – File selections for the material (integer)

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query.

Sample Batch-mode Input:

	Record Type
`TAPE.654`	(1)
(blank record)	(2) default
(blank record)	(3) no annotated list
(blank record)	(4) default
`TAPE.633`	(1)
`IRON.SUM`	(2)
`IRON.FULL`	(3)
`N,N`	(4)
`26054.,1,3,8,9,10`	(5) ZA no.
`2634,1,2`	(5) MAT no.
(blank record)	(5) end
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `LISTEF :== $LISTEF.EXE`, has been defined, then some of the input data may be given on the command line as follows

`LISTEF` *input_file/summary_file/output_file/standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. If no summary or output file is given, then the file is not produced. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.

PLOTEF

PLOTEF is a program designed to produce graphical displays of a data file in either ENDF-6 or ENDF-5 format. The form of graphical output depends on the graphical devices available at the installation where this code will be used.

Fortran Logical Units Used:

5 — Input
6 — Output (other than graphical)
20 — Input data file, ENDF format
21 — Temporary file for plot selection data
22 — Temporary file for paging large data arrays

Input Requirements:

In batch mode operation, the user must first supply a record to specify the format and the name of the graphics output file.

RECORD 1 — Selection of graphic output file format and name
FIELD 1 – Graphic format selection
1 — PostScript format instructions
2 — REGIS format instructions
FIELD 2 – Graphic output file name (PostScript only)

Then the user must supply the following control information repeated for each input file to be processed.

RECORD 2 — Input file specification
RECORD 3 — Material processing control (character)
Y — Plot the entire file
N — Plot only selected portions of the file

If RECORD 3 is left blank, then all materials, all files and all sections will be plotted. If only selected materials, files and sections are to be processed, the selections, one material per record up to a maximum of 30, are given with the following contents. They need not be given in material number order although the graphics output will be ordered that way. Plot specification records are terminated by a blank record.

RECORD(S) 4 — Selection of files and sections for a material

FIELD 1 – Material number or ZA number

Material number must be 1 to 4 digits with no period.

ZA number must be a 4 or 5 digit number with a trailing period having the value 1000.*Z + A for the desired material.

FIELD(S) 2 thru 13 – Plot selections for the material (integer)

These records can be up to 250 characters long but are read by the program 80 characters at a time, so use a dash(-) to indicate that input for the material continues. Plot selections for a given material are specified in the form

```
MF-specification(MT-specification;MT-specification;...)
```

where

MF-specification format
- Blank – all values of MF for the material
- MF-value – a single selected MF for the material
- MF/MF – a range of MF-values

MT-specification format
- Blank – all values of MT for the selected MF's
- MT-value – a single selected MT for the selected MF's
- MT/MT – a range of MT-values for the selected MF's

Examples:

1. 9440,3,12/14
   Files 3, 12, 13 and 14 of material 9440 will be plotted.

2. 9440
   All of material 9440 will be plotted.

3. 9440,(51/90)
   All inelastic reaction data in all files of 9440 will be plotted.

4. 94240.,3(18/21;38)
   All fission cross section of Plutonium-240 will be plotted.

5. 94241.,1(452;455/456),3/5(18/21;38),-
   8(457),12/15
   Complex selection with continuation(-).

Multiple input files can be processed to produce multiple output files by repeating the above input data sequence. The program execution is terminated by a record containing the word DONE.

In interactive mode operation, the above data is supplied in response to the appropriate query.

Special Requirements:

The PLOTEF source code contains graphics subroutines that generate and manipulate graphic data which are independent of the output graphics device. In order to produce output for a given graphics device, a device driver must be written by the user for his device. The PLOTEF software have small amounts of clearly labeled code which should be modified to meet the requirements of the user's computer facility and output graphics device. Drivers for PostScript and for REGIS are included.

Sample Batch-mode Input:

Input	Record Type
`1,FE.PS`	(1)
`TAPE.654`	(2)
(blank record)	(3) default
`TAPE.633`	(2)
`N`	(3)
`26054.,1,3/5(4;51/90),8,-`	(4) ZA no.
`11/15`	(4) continuation
`2634`	(4) MAT no.
(blank record)	(4) end
`DONE`	end of run

OpenVMS Command Mode Operation:

If the OpenVMS command, `PLOTEF :== $PLOTEF.EXE`, has been defined, then some of the input data may be given on the command line as follows

`PLOTEF` *input_file/ output_file/ graphic_format/ standard_options*`[Y or N]`

If no input data is given on the command line, the program operates in the normal interactive mode. The graphic format choices are 1(PostScript) or 2(REGIS). The default format is PostScript. If non-standard options are selected, then the program prompts for the non-standard option input in the usual way.