Library File
Editor
(LFE)
User's Manual

093-000074-05

For the latest enhancements, cautions, documentation changes, and other information on this product, please see the Release Notice (085-series) supplied with the software.

Ordering No. 093-000074

Printed in the United States of America
Revision 05, November 1977

NOTICE

Library File Editor
(LFE)
User's Manual
093-000074

Revision History:

Original Release - March 1972
First Revision - December 1972
Second Revision - October 1973
Third Revision - June 1974
Fourth Revision - September 1975
Fifth Revision - November 1977

093-000074-05 is a minor revision of this manual. A vertical bar or an asterisk in the margin of a page indicates substantive change or deletion, respectively, from 093-000074-04.

CONTENTS

CHAPTER 1

USE OF THE LIBRARY FILE EDITOR

REFERENCE NUMBERS

The following tape and manual numbers apply to the Library File Editor.

STAND-ALONE OPERATING SYSTEM (SOS)

Tapes:

091-000057 Absolute binary tape for paper tape configurations that use the absolute binary loader.

089-000081 Relocatable binary tape for SOS configurations that use the relocatable binary loader.

Manuals:

093-000003 Binary Loader

093-000080 Extended Relocatable Loaders

093-000062 Stand-alone Operating System

DISK OPERATING SYSTEMS (RDOS and DOS)

Tapes:

088-000031 (LFE.SV) Dump Tape

Manuals:

093-000003 Binary Loader

093-000040 Extended Assembler

093-000080 Extended Relocatable Loaders

093-000081 RDOS/DOS Macroassembler

GENERAL

The Library File Editor provides a means of updating and interpreting library files. A library file (also called library) is comprised of a set of relocatable binary files (produced by the Extended Assembler or by the Macroassembler) that is denoted by special beginning and ending blocks. For example,

```
LIBRARY START BLOCK
prog1.RB
        .
        .
        .
progn.RB
LIBRARY END BLOCK
```

where each prog.RB represents one of a set of relocatable binary programs.

Library START and END blocks are described in the Extended Relocatable Loaders Manual. Library tapes are supplied with the DGC operating systems and with subsystems such as ALGOL and FORTRAN.

With LFE, the user can analyze the contents of a library file, list titles in a library file, merge libraries, update libraries, extract logical records from a library file, and create new library files from existing system files or new material. The LFE is of special importance in ordering relocatable programs in a library file, since the relocatable loader uses this order to determine which programs will be loaded.

To be loaded, a program must have a global entry which resolves to an external declaration in a previously loaded program, or you must have included the LFE /F switch, which force-loads this library. This means that if program A on library file 1 has been loaded and contains a call to program B on library file 1, then B must follow A in the RLDR command line to be loaded. If there are no unresolved external symbols defined as entries in the relocatable binary program and thus no calls to the program, the program is not loaded (unless you included the LFE /F switch).

In some cases it may be necessary to provide two or more copies of a given program on a library file to insure proper referencing. For example:

Program A calls → B calls → C calls → A

(Assume that C follows A in the library file.) If a previously loaded program has called A, then A, B and C are loaded via the standard mechanism. However, if a previously loaded program has called B, then only programs B and C would be loaded. For this case, a second copy of program A should be placed after program C. One of the LFE commands, Analyze (A), allows the user to determine whether the programs on the file are the proper selection and in the correct order for his purposes, since the command causes listing of global declarations of the library file.

The LFE is supplied in three forms for stand-alone operation: absolute binary, relocatable binary, and core image files on cassette and magnetic tape (See above). The core image form is included on a master tape for stand-alone system operation from cassette or magnetic tape. The LFE is supplied as a dump tape under RDOS.

All forms of the LFE operate in nearly identical fashion, their principal differences being the ways in which they communicate with the operator (error messages, operator cues, command format, etc.).

The Stand-alone Operating System LFE is described in Chapter 2, RDOS LFE in Chapter 3.

The terminology used to describe the relocatable binaries that make up a library is as follows:

logical record - A relocatable binary record contained within a library is called a logical record.

binary - Outside the library file, the relocatable binary may be a separate file called a binary. A binary may be produced by extracting one or more relocatable binary records from a library or can be the output of an assembly.

update - A binary that is to be inserted into a library, either to replace a current logical record or to create a new logical record, is called an update.

END OF CHAPTER

CHAPTER 2
STAND-ALONE OPERATING SYSTEM LFE

The Stand-alone Operating System LFE is supplied as either an absolute binary tape for paper tape configurations that use the absolute binary loader, or a relocatable binary tape for SOS configurations that use the relocatable binary loader.

Users having SOS cassette or magnetic tape systems are supplied with system programs on cassette or magnetic tape in both relocatable binary and core image format. The core image LFE can be loaded with the core image loader.

For further information on loading LFE, see the appropriate manual: Binary Loader; Extended Relocatable Loaders; Stand-alone Operating System (see page 1-1).

OPERATION

When either tape is loaded with the appropriate loader, the LFE issues the prompt LFE and the user types in command strings on the console keyboard. Error, caution, and prompting messages are issued by the LFE to the console output.

COMMANDS

Key Letters

Each command string begins with a key letter that indicates to the LFE what operation is to be performed on the arguments that follow the key letter. The following table lists the key letters and the commands they represent:

Key Letter	Command
A	Analyze a set of library files and/or binaries, or analyze selected records in a library.
D	Delete logical records from a library.
I	Insert binaries into either a new or an existing library.
M	Merge a library and binaries to form a new library.
R	Replace logical records in a library with new binaries.
T	List titles in a set of libraries or binaries.
X	Extract specific logical records from a library.

Command String Structure

A command string begins with a key letter followed by arguments. Arguments can be either device names or logical record names. Arguments are separated by at least one space; additional spaces are ignored.

An argument may have one or two switch options. A switch is indicated by a / following the argument; the / is immediately followed by a letter or a number.

Each command string is terminated by a carriage return (). Command strings can be extended beyond one line by typing SHIFT N (↑) immediately before the carriage return. The teletypewriter echoes a carriage return/line feed and the command string can be continued. Only one key letter is permitted per command string.

Typing control A at any time aborts the current operation and causes the LFE to output the prompt LFE.

Typical command string structure is:

key letter devicename/switch recordname)

Device Names

The following device names are recognized by the stand-alone LFE:

Device Name	Device Description
$TTP	Teletype punch output
$TTO	Teletype printer output
$TTI	Teletype keyboard input
$LPT	Line printer output
$PTR	Paper tape reader input
$PTP	Paper tape punch output
$TTR	Teletype reader input
CTx:yy	Cassette tape drive
MTx:yy	Magnetic tape drive

where:

x stands for a number in the range 0-7 representing the tape drive selected.

yy stands for a number in the range 0-99 representing the tape file selected.

Record Names

Record names are those names assigned to relocatable binary records by the .TITL pseudo-op. Only the first five characters in a record name are meaningful.

Switches

Arguments may be modified by switches. A switch is indicated by a right slash (/) followed by either a letter or a decimal digit. A blank space between the switch indicator / and the argument it modifies is optional. However, no space is permitted between the slash and the letter or number following.

Numeric Switches

Numeric switches specify the number of times that the previous argument is to be repeated. For example:

```
$PTR/2
```

is equivalent to $PTR $PTR. A numeric switch of one (/1) has no effect. Numeric switches will be indicated by /# in the descriptions of specific commands.

If two numeric switches are associated with an argument, only the more recent one will be accepted. For example, $PTR/3/2 is equivalent to $PTR/2.

Letter Switches

Letter switches have distinct meanings that depend upon the arguments they modify and the command string in which they are found. All allowable letter switches will be explained in the descriptions of specific command functions.

Arguments having switches /I (input library), /L (listing switch) or /O (output switch) can be situated anywhere in the command string following the key letter.

Command String Corrections

An entire command string can be deleted by typing SHIFT L (\). Single characters in a command string can be deleted by depressing RUBOUT. The most recently entered command string character will be deleted each time RUBOUT is depressed, and the back arrow "←" will be printed once per deletion. If the first character in a command string is deleted, the standard prompt LFE will be printed, indicating that the program is ready to accept a new command string. For example:

```
D $PTR←←←   is equivalent to D $.
D $PTR←←←←←←   deletes the entire command string,
               causing the prompt LFE to be output.
```

COMMAND DESCRIPTIONS

Following are definitions and descriptions of each Library File Editor command. Extra switches not specified in the format are generally ignored. Optional switches and arguments are enclosed in square brackets in the specification of command string format.

Analyze (A)

The Analyze command itemizes the global declarations of a library file, of specific logical records within a file, or of single relocatable binary records. Records are analyzed in the order of their appearance during the serial scanning of the input.

An analysis produces the following output:

1. Listing of all global declarations (symbol, symbol type, and flags).
2. Cross-reference of all external records in the file called by each analyzed record.
3. Title of the module containing each external record referenced by the analyzed record.
4. Count of ZREL and NREL locations required by each analyzed record.

At the end of a library analysis a total count of all needed ZREL and NREL locations is given. (The total count given after a single binary analysis is the same as the count named in 4 above.)

Symbol types are:

T	Title of record
ED	Entry Displacement (must be referenced by an external displacement)
EN	Entry Normal (must be referenced by an external normal)
EO	Entry Overlay (must be referenced by an external normal)
D	External Displacement
N	External Normal

Each entry containing either a definition error or phase error is also flagged. Symbol flags are:

M	Multiply-defined entry (note that symbol definitions must be unique in their first five characters)
U	Undefined entry (an external normal or external displacement references an undefined entry)
P	Phase error (an external normal or external displacement whose entry was defined before the external reference)

Format

A[listingname/L] inputname[/B][/#]...[recordname[/R]...

where: listingname is the name of the device used to output the analysis;

inputname is the name of the input device which reads the binaries or libraries to be analyzed;

recordname specifies a particular logical record in the input library to be analyzed.

Switches:

/B Binary record switch, indicates that one or more binary records will be read on the input device. Absence of /B in the inputname indicates that one or more libraries will be read.

/L Listing switch, indicates that the listingname device will output the analysis. Absence of this switch causes output to the $TTO by default.

/R Record switch, specifies analysis of the logical recordname preceding the switch, to the exclusion of the other logical records in the library.

/# Number switch, indicates how many binary records or libraries will be read.

Examples

A $TTO/L $PTR ABC/R)

causes logical record ABC, in a library loaded on the high speed reader, to be analyzed. The analysis will be output on the teletype printer.

A $PTR ABC/R CDE/R)

causes logical records ABC and CDE, in a library loaded on the high speed reader, to be analyzed. The analysis will be output on the teletype printer by default.

Output

The following is a sample of output generated by the A command.

```
LFE A $PTR $TTO/L

LOAD $PTR, STRIKE ANY KEY.
   T  RESID
   EN RBIN
   EN RLIN
   EN WRIB
   EN WRLI
   ED RTRN       OVLA1    OVLA2
   ED SAVE       OVLA1    OVLA2
   ED STBT       OVLA1    OVLA2
   ED LDBT       OVLA1    OVLA2
   ED K30        OVLA2
   ED K10        OVLA2
U  N  BEND
U  N  BTAB
U  N  RLOC
U  N  OVLA3
   N  OVLA2      OVL2
   N  OVLA1      OVL1
PAGE ZERO RELOCATABLE DATA = 000166
NORMAL RELOCATABLE DATA = 001700

   T  OVL1
   EO OVLA1      RESID
P  D  STBT       RESID
P  D  LDBT       RESID
P  N  RTRN       RESID
P  N  SAVE       RESID
PAGE ZERO RELOCATABLE DATA = 000000
NORMAL RELOCATABLE DATA = 000336

   T  OVL2 <-------------- Title
   ED OVLA2      RESID <-- Entry Overlay
P  D  LDBT       RESID  }
P  D  STBT       RESID  }  External Displacement
P  D  K30        RESID  }
P  D  K10        RESID  }
P  N  RTRN       RESID  }  External Normal
P  N  SAVE       RESID  }
PAGE ZERO RELOCATABLE DATA = 000000 } Locations Count
NORMAL RELOCATABLE DATA = 000322    } for OVL2

TOTAL ZREL COUNT:  000166 }
TOTAL NREL COUNT:  002560 } <-------- Total Locations Count
-----------------------------------

LFE
```

Delete (D)

To delete one or more logical records from a library.

Format

D inputname/I outputname/O recordname [recordname...]

where: inputname is the name of the device containing the library whose selected logical recordname(s) will be deleted;

outputname is the name of the device which will produce the new library.

Switches

/I Input library file switch, specifying the device which will read the existing library.

/O Output library file switch, naming the device which will produce the new library.

Example

D $PTR/I $PTP/O ABC DEF)

deletes logical records ABC and DEF from the input library file, and outputs the new library on the high speed punch.

Insert (I)

To insert binaries into an existing library or to create a new library.

Format

I [inputname/I] outputname/O [insertname[/#]...]
[recordname/A insertname[/#]...]...
[recordname/B insertname[/#]...]...

where: inputname is the name of the device to read in the existing library;

outputname is the name of the device producing the new library;

insertname is the name of the device reading in the binaries which are to be inserted;

recordname is the name of the logical record (in the existing library) before or after which the insertions will occur.

Switches

/I Input switch, labeling inputname. When no inputname argument is specified, no logical record names appear and all updates are read in the order given to become a new library file.

/# Numeric switch, indicating the number of binaries which will be inserted.

/O Output switch, identifying the outputname device.

/A After switch, naming the logical recordname after which insertions will occur.

/B Before switch, naming the logical recordname before which insertions will occur.

When LFE reaches a record specified by an A or B switch, it prints that record's name as a prompt before accepting updates.

Examples

I $PTP/O $PTR/3)

causes three relocatable binary records mounted on the high speed reader to be punched as a library file by the high speed punch.

I MT0:3/I $PTP/O $PTR)

causes one relocatable binary record from file number 3 on magnetic tape drive number 0 to be added to the beginning of a library file on the high speed reader, and outputs the new library file on the high speed punch.

I $PTR/I $PTP/O M/A $PTR)

causes a library to be updated with binaries. If the input library consists of logical records L, M, and N, and the update record is Q, Q is inserted after record M. The updated library consists of records L, M, Q, and N. The high speed reader is the single input device, and the new library is output on the high speed punch.

Caution:

If [insertname[/#]...] is used with no preceding recordname, all updates read in via this device will be inserted at the beginning of the new library file.

If inputname is the same as insertname, the input library master must be reread from the beginning of the file after insertion has occurred. Logical records which were read before insertion will be ignored.

All the insert options (/A, /B, etc.) may be used in the same command string, but the same recordname cannot be used more than once in an insert command string.

Merge (M)

To combine binaries and/or libraries into one library.

Format

M outputname/O inputname [/B] [/#]...
[inputname [/B] [/#]]...

where: outputname is the name of the device which will produce the new library;

inputname is the name of the device which will read in the binaries or libraries to be merged.

Switches

/O Output library file switch, naming the device which is to output the new library file.

/B Binary switch, labeling the inputname device which will read in update binaries. If /B is omitted, libraries will be read by the inputname device.

/# Numeric switch, detailing the number of libraries or binaries which will be merged.

Examples

M $PTP/O $PTR $PTR/B)

causes one binary record to be added to the end of a library file. The new library file is punched by the high speed punch.

M CT1:23/O $PTR/3 $PTR/B/2)

causes three libraries to be input, then two binaries, forming a new library output as file number 23 on cassette drive number 1.

Caution:

Libraries and binaries must be input in the same order as they occur in the command string.

Replace (R)

To replace logical records in a library file with new binaries.

Format

R inputname/I outputname/O recordname updatename
[recordname updatename]...

where: inputname is the name of the device containing the existing library;

outputname is the name of the device which will output the new library;

recordname is the name of the logical record in the existing library which will be replaced;

updatename is the name of the device which will read in the update.

Switches

/I Input library switch, naming the device to read in the existing library.

/O Output library switch, naming the device which will produce the new library.

Example

R $PTR/I $PTP/O REC1 $PTR)

causes a library file, input on the high speed reader, to have one of its logical records (REC1) replaced by an update input via the high speed reader. The new library will be punched on the high speed punch.

The name of a logical record will be given as a prompt before its update is read in.

Caution:

Update recordnames and their associated updatename devices must be given in pairs in the command string. Record replacement occurs in the same order as the updates are read in.

Title (T)

To produce a title (.TITL) listing of binaries or libraries.

Format

T [outputname/L] inputname [/B] [/#]...
[inputname [/B] [/#]]...

where: outputname is the name of the device which will produce the listing;

inputname is the name of the device which will read in the library or binary.

Switches

/L Listing switch, indicating the device which will produce the listing. Absence of /L causes the $TTO to be selected by default.

/B Binary switch, labeling the inputname device which will read in one or more binaries. If /B is omitted, libraries will be read by the inputname device.

/# Numeric switch, indicating how many libraries or binaries will be input.

Example

T $PTR)

causes the titles of all logical records in the library file (mounted on the high speed reader) to be printed. Since no listing device is given, the teletype printer is selected by default.

Caution:

Titles will be listed in the order of their appearance on the inputname device(s).

Extract (X)

To extract a copy of a logical record from a library file.

Format

X inputname/I outputname/O recordname

where: inputname is the name of the device to read in the library;

outputname is the name of the device which will produce a copy of the extracted record;

recordname is the name of the logical record which will be extracted.

Switches

/I Input switch, labeling the inputname device.

/O Output switch, labeling the outputname device which will produce a copy of the extracted record.

Example

X $PTR/I $PTP/O ABC)

causes logical record ABC to be extracted from its library file. The library file is input on the high speed reader; record ABC is punched by the high speed punch. Output will be a binary.

ERROR MESSAGES

Errors may be caused by an improper command string or occur during the attempted execution of a command. Messages are listed to flag errors of both types. If an error condition is noted in the command string, an error message will be issued as soon as the string terminator (↓) is detected. No output will result.

ILLEGAL KEY: followed by input illegal key.

First typed letter (key) does not match any of the available commands. Example:

```
C $PTR↓
↑
```

SWITCH ERROR: followed by the faulty switch.

A switch not permitted by the command format is used. Example:

```
T $PTR/R↓
      ↑
```

TOO MANY ARGUMENTS IN COMMAND LINE

The command line buffer (200_8 characters) has been exceeded.

NO INPUT FILE?

Command requires an input device and none was specified. Example:

```
D $PTP/O ABC↓
```

NO OUTPUT FILE?

Command requires an output device and none was specified. Example:

```
X $TTR/I ABC↓
```

If an error condition occurs during the execution of a command, an error message is output. The error message attempts to name the file/device responsible for the error. If a library file is being output, it will be closed with an end block. The following list summarizes the command execution errors.

ERROR CONDITION IN INPUT FILE:
inputname

Faulty name assigned to device which will read in the input file. Example:

```
T $TTO/L $PTI↓
         ↑
```

ERROR CONDITION IN OUTPUT FILE:
outputname

Faulty name assigned to device which will punch the output file. Example:

```
I $PPP/O $PTR/3↓
  ↑
```

ERROR CONDITION IN UPDATE FILE:
updatename

Faulty name assigned to device which will read in the update record. Example:

```
R $PTR/I $PTP/O DATAB $PTI↓
                      ↑
```

ERROR CONDITION IN LISTING FILE:
listingname

Faulty name specified for listing device. Example:

```
T $TOT/L $PTR↓
      ↑
```

CHECKSUM ERROR IN LOGICAL RECORD:
recordname

The paper tape logical record has been damaged or incorrectly punched.

CHECKSUM ERROR IN UPDATE FILE:
updatename

The paper tape update file has been damaged or incorrectly punched.

BLOCK ERROR IN UPDATE FILE:
updatename

Format of input block was improper. For example, a library was read in as an update when a relocatable record was called for.

BLOCK ERROR IN LOGICAL RECORD:
inputname

Logical record format error. Typically caused by inputting a file that is not a library file.

LOGICAL RECORD NOT RECOVERABLE:
recordname

A device is shared by both the input library and an update record. The second reading of the input file differs from the first reading. For example, the input library and the update record share a common input device. An attempt is made to replace one record in a master file. During the second pass of the input file, the wrong master is read.

UPDATE FILE NOT FOUND FOR L.R:

R command format requires both a recordname and an updatename. One or both is missing. For example:

R $PTR/I $PTP/O ABC)

SYMBOL TABLE OVERFLOW

During execution of an Analysis, insufficient core memory is available for the symbol table.

UNEXPECTED ERROR FROM SYSTEM

Hardware malfunction has occurred.

If an error occurs while a library start block is being read, the appropriate error message will be followed by "LB.ST" instead of a file or record name.

If no particular file or record device name can be associated with an error condition, an error message will be output followed by a blank line. A typical error causing this condition is the presence of a control character in a device name: (form feed) $PTR.

CAUTION MESSAGES

Two caution messages are used to signal the presence of non-fatal error conditions. In the T or A commands, if no listing device is named, the message:

NO LISTING FILE: DEFAULT LISTING ON TTO

is output and the teletypewriter is used for the listing device.

If any of the record names appearing in an argument list is not found:

LOGICAL RECORD NOT FOUND:
recordname

is output.

The above caution messages may indicate non-fatal errors in the command string. Such errors might produce commands which are executable, yet whose operation differs from what was intended.

OPERATOR CUE MESSAGES

The program provides several messages to prompt operator action during use of the editor. The message LFE at the left hand margin indicates that the program is ready to accept commands.

Before any input device is activated, the message:

LOAD devicename, STRIKE ANY KEY

is issued. An INPUT or UPDATE message will preceed this message if the message refers to an INPUT or UPDATE device.

When an update record is to be read in the same device which inputs the library file (in an Insert or Replace command), the message:

REMOVE INPUT MASTER AND LOAD U.F

is issued. After the update record has been read in, the library file must again be read. The message:

REMOVE U.F AND LOAD BACK INPUT MASTER

is issued. The library file should be loaded at the beginning. The message:

LOAD inputname, STRIKE ANY KEY

is issued and carried out; the input device reads from the start of the library. Any logical records on the file which have already been processed by the editor will be ignored.

Before executing an Insert or Replace command, the LFE will print the name of the last logical record it has read.

END OF CHAPTER

CHAPTER 3
RDOS LIBRARY FILE EDITOR

The RDOS LFE is supplied with RDOS and DOS systems, or as a DUMP tape, and has the name LFE.SV.

The LFE works with conventional RB files and with extended RB files (produced by certain compilers and by the Macroassembler if global /T was specified to MAC).

OPERATION

The operator communicates with the LFE through the command line interpreter (CLI). When the CLI prints its ready prompt (R) on the teletype, the operator may enter an LFE command string on the teletypewriter. Error and caution messages are issued by the program on the console output.

COMMANDS

Key Letters

Each command string starts with the CLI command "LFE" followed by a single LFE command key letter. This key letter indicates what operation is to be performed on the arguments that follow the key letter. The following table lists the key letters and the commands they represent.

Key Letter	Command
A	Analyze a set of library files and/or binaries, or analyze selected records in a library.
D	Delete logical records from a library.
I	Insert binaries into either a new or an existing library.
M	Merge libraries to form a new library.
N	Create a New library from one or more binaries.
R	Replace logical records in a library with new binaries.
T	List Titles in a set of libraries or binaries.
X	EXtract specific logical records from a library.

Command String Structure

An LFE command string consists of "LFE," followed by a command key letter, followed by arguments. Arguments can be file names or logical record names. Arguments must be separated by at least one space; additional spaces are ignored.

An argument may have one or two switch options. A switch is indicated by a / following the argument; the / is immediately followed by a letter or a number.

Each command string is terminated by a carriage return (). Typical command string structure is:

LFE keyletter filename/switch recordname)

The following five rules apply to LFE command strings and to the commands involved.

1. Only one command key may be given per command string.
2. An input library file and an update file cannot reside on the same device, for example, the $PTR. Both can, of course, be on disk.
3. An input file is searched for in the specified RDOS directory as inputname.LB; if not found, a search is made for inputname.
4. An update file is searched for in the specified RDOS directory as filename.RB (or .LB when using A, M or T command keys); if not found, a search is made for filename.
5. All references to logical records are satisfied by the first matching five-character record title in the library file. Therefore, each logical record on a file should have a unique title.

Command strings can be extended beyond one line by typing SHIFT-N or SHIFT-6 (on a DASHER) immediately before the carriage return ().

Switches

Arguments may be modified by switches. A switch is indicated by a right slash (/) followed by either a letter or a decimal digit. A blank space between the switch indicator (/) and the argument it modifies is optional. However, no space is permitted between the slash and the letter or number following.

Letter Switches

Letter switches have distinct meanings that depend upon the arguments they modify and the command string in which they are found. All allowable letter switches will be explained in the descriptions of specific command functions. Arguments having switches /I (input library), /L (listing switch), /O (output switch), or /F (force-load switch) can be situated anywhere in the command string following the key letter.

COMMAND DESCRIPTIONS

The following pages describe LFE commands. Optional switches and arguments are enclosed in square brackets in the command string format.

Analyze (A)

The A command itemizes the global declarations of a library file, of specific logical records within a library file, or of binaries. Records are analyzed in the order of their appearance during the serial scanning of the input.

An analysis produces the following output.

1. The name of the library, printed at the start of the analysis.
2. Listing of all global declarations (symbol, symbol type, and flags).
3. Cross-reference of all external records in the file called by each analyzed record.
4. Title of the module containing each external record referenced by the analyzed record.
5. Count of ZREL and NREL locations of unlabeled common required by each analyzed record.

At the end of a library analysis a total count of all needed ZREL and NREL locations is given. (The total count given after a single binary analysis is the same as the count named in 5 above.)

Symbol types are:

CM	Named CoMmon symbol
ED	Entry Displacement (a page zero entry)
EN	Entry Normal (an entry outside page zero)
EO	Entry Overlay
GD	GADD reference
GL	GLOC reference
GR	GREF reference
T	Title of record
XD	EXternal Displacement
XN	EXternal Normal

Each entry containing either a definition error or phase error is also flagged. Symbol flags are:

M	Multiply-defined entry (note that symbol definitions must be unique in their first five characters). References to multiply-defined entry names are preceded by an asterisk (*).
U	Undefined entry (an external normal or external displacement references an undefined entry).
P	Phase error (an external normal or external displacement whose entry was defined before the external reference).

Format

LFE A [listingname/L] inputname [recordname]...

LFE A/M [listingname/L inputname inputname]...

where: listingname is the name of the file into which the analysis is to be written;

inputname is the name of the file from which the input is read;

recordname specifies a particular logical record in the input library to be analyzed.

Switches

/M Multiple input library files. The switch modifies the command key A and causes all file names following, with the exception of a listing file, to be analyzed as one library.

/B Inputname is a Binary. Default extension is ".RB".

/F Form feed. Each logical record analysis is on a separate page.

/L Listing file. By default the analysis is listed on $LPT. The switch assigns the preceding file for listing.

Examples

LFE A/M MATH1.LB MATH2.LB $LPT/L)

The library files MATH1.LB and MATH2.LB are analyzed as one library and the results are printed on the line printer. (Note that recordname may not be used with switch M.)

LFE A DP1:M.LB)

The input file is M.LB in directory DP1. All the logical records in this library file are analyzed and the results are printed on $LPT (default listing file).

LFE A MATH.LB SIN COS TAN $LPT/L)

The input file is MATH.LB. The logical records SIN, COS, and TAN are analyzed and the results are printed on the line printer.

Output

The following is a sample of output generated by the A command.

```
AEL.LB

    T   RESID
    ED K10          OVL2
    ED K30          OVL2
    ED LDBT         OVL1        OVL2
    ED STBT         OVL1        OVL2
    ED SAVE         OVL1        OVL2
    ED RTRN         OVL1        OVL2
    EN WRLI
    EN WRIB
    EN RLIN
    EN RBIN
U   GL LOCAT
    XN OVLA1        OVL1
    XN OVLA2        OVL2
U   XN OVLA3
U   XN RLOC
U   XN BTAB
U   XN BEND
U   XN LOCAT

PAGE ZERO RELOCATABLE DATA = 000166
NORMAL RELOCATABLE DATA    = 001700
UNLABELED COMMON SIZE      = 000011

    T  OVL1
    EO OVLA1        RESID
P   XD LDBT         RESID
P   XD STBT         RESID
U   GD LOCA2
P   XN SAVE         RESID
P   XN RTRN         RESID
U   XN LOCA2

PAGE ZERO RELOCATABLE DATA = 000000
NORMAL RELOCATABLE DATA    = 000336

    T  OVL2
    CM CMAR
    ED OVLA2        RESID
P   XD K10          RESID
P   XD K30          RESID
P   XD STBT         RESID
P   XD LDBT         RESID
P   XN SAVE         RESID
P   XN RTRN         RESID

PAGE ZERO RELOCATABLE DATA = 000000
NORMAL RELOCATABLE DATA    = 000322

TOTAL ZREL COUNT = 000166
TOTAL NREL COUNT = 002560
---------------------------------
```

Delete (D)

To delete one or more logical records from a library.

Format

LFE D inputname outputname/O recordname
[recordname...]

where: inputname is the name of the file containing the library whose selected logical recordname(s) will be deleted;

outputname is the name of the device that will produce the new library.

Switches

/O Output master library file. The switch must always modify the name of the output library file, which can appear anywhere within the command line.

Example

```
LFE D $TTR UTIL.LB/O MOVE LDBYT STBYT DIVI)
    MULT COMP )
```

The input file is $TTR.
The output file is UTIL.LB.
The logical records deleted from the input file are:

MOVE
LDBYT
STBYT
DIVI
MULT
COMP

Insert (I)

The I command merges update files and logical records on an input library file to produce an output library file.

By default, update files in the order listed in the command will be inserted before the first logical record in the input file. To insert an update file or files before or after a given logical record, use the /A or /B switches as described below. A given logical record may appear only once in a command.

No local symbols present in the update files are transferred to the output file.

Format

LFE I [inputname] outputname/O[insertname[/F]...]...
[recordname/A insertname[/F]...]...
[recordname/B insertname[/F]...]...

where: inputname is the name of the file from which the existing file is taken;

outputname is the name of the file to contain the new library;

insertname is the name of a file from which binaries are taken for insertion;

recordname is the name of a logical record in the existing library before or after which insertions are to be made.

Switches

/O Output library file. The switch must always modify the name of the output library file.

/F Force-load this binary. This sets the force-load flag in the binary; RLDR will load this binary whether or not it satisfies an external reference.

/A Insert after. The switch is used to insert arguments after the logical recordname which it modifies.

/B Insert before. The switch appears after a logical recordname, and is used to insert arguments before this recordname.

Example

```
LFE I $PTR MATH.LB/O A.RB B.RB SINE/A C.RB D.RB
      COS/A  X.RB  Y.RB  Z.RB
```

inputname is $PTR. outputname is MATH.LB. Files A.RB and B.RB are inserted at the beginning of the output file. Files C.RB and D.RB are inserted after the program SINE in the output file. Files X.RB, Y.RB and Z.RB are inserted after the program COS in the output file. (Note that SINE need not precede COS on the input file.)

Merge (M)

To combine existing libraries into one library.

Format:

LFE M outputname/O inputname [inputname...]

where: outputname is the name of the file to receive the new library;

inputname is the name of a file from which a library is to be taken.

Switches

/O Output library file switch, naming the file that is to receive the new library.

Example

```
LFE M FORT.LB/O FORT1.LB FORT2.LB FORT3.LB↑)
      FORT4.LB)
```

The four FORTRAN library files are merged into a single FORTRAN library file called FORT.LB.

New (N)

To create a new library file named outputname from one or more relocatable binary files.

Format

LFE N outputname/O inputname [inputname [/F]...]

where: outputname is the name of the file to receive the new library;

inputname is the name of the file containing a relocatable binary to be included in the new library.

Switches

/O Output library file. The switch always modifies the outputname file.

/F Force-load this binary; set the force-load flag. RLDR will load this binary whether or not it satisfies an external reference.

Example

```
LFE N LIB3.LB/O A.RB B.RB/F C.RB
```

The output is file LIB3.LB; it consists of binaries A.RB, B.RB, and C.RB. B.RB has the force-load flag set; RLDR will always load it when LIB3.LB appears in the RLDR command line.

Replace (R)

To produce an output file, replacing logical records in the input file with relocatable binary update files.

No local symbols present in the update files are transferred to the output master.

Format

LFE R inputname outputname/O recordname updatename [/F]
[recordname updatename [/F] ...]

where: inputname is the name of the file containing the existing library;

outputname is the name of the file that is to receive the new library;

recordname is the name of a logical record in the existing library that is to be replaced;

updatename is the name of the file from which a binary is to be taken.

Switches

/O Output library file. The switch always modifies the outputname file name.

/F Force-load this binary; set the force-load flag. RLDR will load this binary whether or not it satisfies an external reference.

Example

```
LFE R MATH.LB MT0:2/O ATAN $PTR TAN TAN.RB↑)
    HSINE $PTR ACOS X.RB)
```

The input file is MATH.LB. The output file is MT0:2. Logical record ATAN is replaced by a file mounted in the paper tape reader, $PTR. Logical record TAN is replaced by file TAN.RB. Logical record HSINE is replaced by the file mounted in the paper tape reader, $PTR. Logical record ACOS is replaced by file X.RB.

Note that all these replacements will be made regardless of the order of the specified logical records on the input file.

TITLES (T)

To produce a title (.TITL) listing of logical records of one or more input libraries.

Format

LFE T inputname [outputname/L] [inputname]...

where: inputname is the name of the file containing an input library;

outputname is the name of the device that is to produce the listing.

Switches

/L indicates the listing device. The listing device argument may appear anywhere in the command line after the function key T.

Example

LFE T $LPT/L $PTR F1.LB $TTR)

The library file is $PTR. Additional library files are F1.LB and $TTR. Titles are listed on the line printer.

Extract (X)

To extract one or more logical records on a library file as separate relocatable binary files. The relocatable binary files will retain the names they had as logical records.

Format

LFE X inputname recordname [recordname...]

where: inputname is the name of the file containing the library from which records are to be extracted;

recordname is the name of a record to be extracted from the input library file.

Switches

None.

Example

LFE X MATH.LB SINE COSIN TAN)

Library file MATH.LB is searched and the logical records SINE, CONSIN and TAN are extracted, creating relocatable binary files SINE.RB, COSIN.RB, and TAN.RB.

ERROR MESSAGES

The following messages indicate fatal errors in the LFE command string. The system will return to the CLI without processing any files.

NOT ENOUGH ARGUMENTS

For example, unpaired arguments to the Replace (R) command.

UNEXPECTED ARGUMENT AT OR FOLLOWING: string

For example, filename/A followed by filename/A for an Insert (I) function.

When there is no string following the colon in the error message, the error occurred at the end of the command line.

INVALID SWITCH FOR: string

For example, a switch other than /M in the analyze (A) function will cause the following message: ILLEGAL SWITCH FOR: A.

NOT A LFE COMMAND: key

A command key that is not recognized by the LFE: any letter key other than A, D, I, M, N, R, T, or X.

TOO MANY ARGUMENTS

The argument string is too long for allocated storage (currently, 500 characters).

ILLEGAL HEADER IN INPUT LIBRARY

No header or an incorrect header block in the library file.

The following messages indicate fatal errors while processing files. When these errors occur, the output file will be terminated with a binary end block before the system returns to the CLI.

CHECKSUM ERROR IN UPDATE FILE: filename

Typically, the message indicates a bad record within filename.

CHECKSUM ERROR IN LOGICAL RECORD: recordname

Very likely the message indicates a bad record. If the checksum occurs within a title block itself, recordname will be the name of the previous logical record. If no previous record exists, recordname will be the name of the library itself.

ILLEGAL BLOCK UPDATE FILE: filename

For example, if a source file is specified as input instead of a binary file, illegal blocks will be encountered.

ILLEGAL BLOCK IN LOGICAL RECORD: recordname

A bad block within a logical record will produce this message. If the expected title is missing, the record name will be the name of the previous logical record within the library.

The following message indicates a fatal error detected by the 'system' rather than LFE:

FILE DOES NOT EXIST, FILE: filename

filename indicates a library file. The error occurs when no input file is found for the command. The error can occur on command lines having commands other than New (N).

Other fatal errors from the 'system' will refer to the LFE.SV file.

CAUTION MESSAGES

The following messages result from non-fatal errors. Processing will continued as indicated for each error.

FILE DOES NOT EXIST, FILE: filename

An update file cannot be found. Search is made for filename and filename.RB. When not found, the file is omitted in processing.

LOGICAL RECORD NOT FOUND - recordname

The input master does not contain recordname. The record (and any corresponding argument) are passed in processing.

DEFAULT OUTPUT IN FILE - filename

An output file specification was expected and not found. filename is used instead as the output file.

FILE ALREADY EXISTS - filename

On an Extract (X) command there is already a file on the output device with the same name as the logical record to be extracted. The logical record is omitted in processing.

UPDATE FILE MATCHES INPUT MASTER: filename

The result is non-fatal as long as there exists at least one valid update file argument. In this case, the matching update file is ignored.

END OF CHAPTER

INDEX

DataGeneral

Software Documentation Remarks Form

How Do You Like This Manual?

Title _____ No. _____

We wrote the book for you, and naturally we had to make certain assumptions about who you are and how you would use it. Your comments will help us correct our assumptions and improve our manuals. Please take a few minutes to respond.

If you have any comments on the software itself, please contact your Data General representative. If you wish to order manuals, consult the Publications Catalog (012-330).

Who Are You?

- ☐ EDP Manager
- ☐ Senior System Analyst
- ☐ Analyst/Programmer
- ☐ Operator
- ☐ Other _____

What programming language(s) do you use? _____

_____

How Do You Use This Manual?

(List in order: 1 = Primary use)

- ____ Introduction to the product
- ____ Reference
- ____ Tutorial Text
- ____ Operating Guide
- ____ _____

Do You Like The Manual?

Yes	Somewhat	No	
☐	☐	☐	Is the manual easy to read?
☐	☐	☐	Is it easy to understand?
☐	☐	☐	Is the topic order easy to follow?
☐	☐	☐	Is the technical information accurate?
☐	☐	☐	Can you easily find what you want?
☐	☐	☐	Do the illustrations help you?
☐	☐	☐	Does the manual tell you everything you need to know?

Comments?

(Please note page number and paragraph where applicable.)

From:

Name _____ Title _____ Company _____

Address _____ Date _____

SD-00742

FOLD DOWN FIRST FOLD DOWN

FIRST
CLASS
PERMIT
No. 26
Southboro
Mass. 01772

BUSINESS REPLY MAIL

No Postage Necessary if Mailed in the United States

Postage will be paid by:

Data General Corporation

Southboro, Massachusetts 01772

ATTENTION: Software Documentation

FOLD UP SECOND FOLD UP

STAPLE