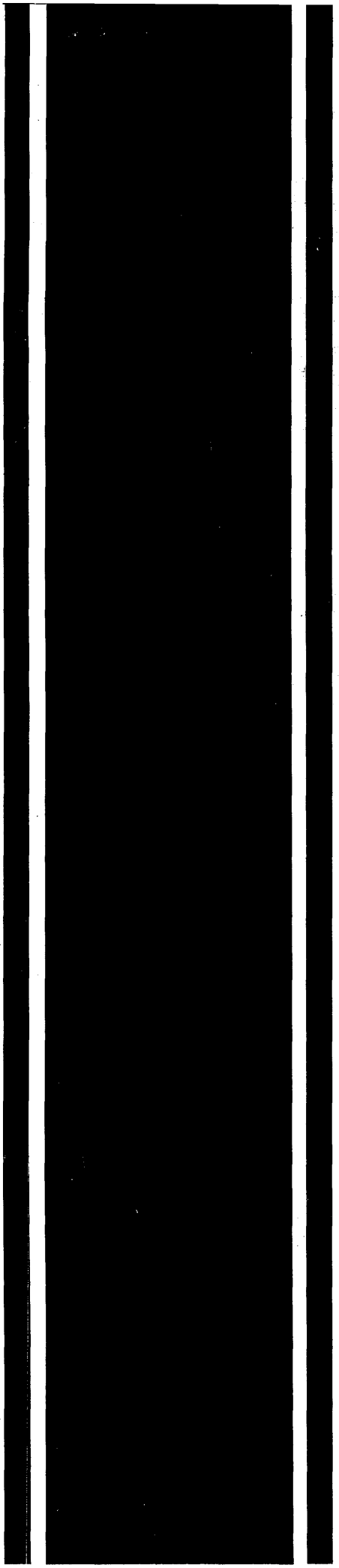


HEWLETT hp PACKARD

# SYMBOLIC EDITOR

# SYMBOLIC EDITOR


11000 Wolfe Road
Cupertino, California
95014

HP 02116-9016

April, 1970

First edition, February, 1967
Second edition, April, 1970



Printed in the U.S.A.



Printed in the U.S.A.

# PREFACE

This manual is the user's guide to the Hewlett-Packard Symbolic Editor, a program for making changes to ASCII code files on punched tape or in mass storage. Typically, such files are source programs written in a programming language such as FORTRAN, ALGOL, Assembler, etc.

No previous experience with HP software is required for use of the Symbolic Editor; it's an independent program that processes its own simple programming language. However, the user should know how to operate the Basic Binary Loader (BBL) as described in the *Operating Manual* (HP 02116-9057).

If the Symbolic Editor is to be used with a mass storage device, the user should also have two other manuals -- *Magnetic Tape System* (HP 02116-91752), and *Prepare Tape System* (HP 02116-91751).

This manual supercedes all previous editions, manual references, and addenda dated prior to April 1970, for the Symbolic Editor. It contains reorganized and expanded descriptions to cover new features for the Edit File, improvements in the operating procedures, and processing for multiple inputs. The Introduction explains the general organization of the Symbolic Editor. Section I defines all the elements of an Edit File, Section II covers the operating procedures, and Section III explains error conditions and correction techniques. Samples of the inputs required, and the outputs produced, by the Symbolic Editor are shown in Appendix A, configuration procedures are described in Appendix B, and card deck processing is described in Appendix C.

An extensive Operating Procedures Flowchart is included at the end of Section II. That flowchart is paginated such that it can be removed from the manual and used as a reference by any Symbolic Editor user.

# CONTENTS

CONTENTS

# INTRODUCTION

Source programs, data records, and other ASCII code information contained in punched tape or mass storage files, are often subject to change. The HP Symbolic Editor (S.E.) provides editing convenience so that a user can access specific lines and characters in an ASCII code file, make the desired changes, then reproduce the entire changed file. The user can access any line number up through $9999_{10}$, and any character number up through $72_{10}$ within a line.

To control the Symbolic Editor, the user writes an ASCII code file of Edit Control lines and Edit Data lines which is called an Edit File (EF). An ASCII file to be edited is called a symbolic file input (SFI), and a reproduced file output is called a symbolic file output (SFO).

## FEATURES OF THE SYMBOLIC EDITOR

In addition to editing, the S.E. can be used to copy a file or process more than one file, editing some and copying others. It can be used to list the contents of any one file. The following list summarizes the features of the Symbolic Editor:

EDITING and/or COPYING:

- A single punched tape to a new single punched tape or to a mass storage file.
- Several punched tapes to a single punched tape, to several new punched tapes, to a single mass storage file, or to several mass storage files.
- A single mass storage file to a punched tape.
- Several mass storage files to several punched tapes.

LISTING:

- The contents of a single punched tape or a single mass storage file.
- The contents of any single mass storage file in a series of mass storage files.

## HARDWARE REQUIREMENTS

The Symbolic Editor is furnished with each HP computer; thus, it operates on a minimum hardware requirement of:

- An HP 2116B, 2115A, 2114A, or 2114B Computer with 4K memory, and
- An HP 2752A Buffered Teleprinter (ASR-33) or an HP 2754B Buffered Heavy-Duty Teleprinter (ASR-35).

Operating convenience of the Symbolic Editor can be significantly improved if the hardware includes:

- An HP 2737A Punched Tape Reader,
- An HP 2753A High-Speed Tape Punch,
- An HP 2778A Line Printer.

To use the Symbolic Editor with mass storage, the hardware must include one of the following devices:

- An HP 2020 Magnetic Tape Unit,
- An HP 3030 Magnetic Tape Unit,
- An HP 7970A Magnetic Tape Unit, or
- A disc or drum memory mass storage device.

## SOFTWARE REQUIREMENTS

The HP Symbolic Editor exists as a single absolute binary tape that must be configured with a set of SIO drivers for the input/output devices. Software requirements are the same as other software subsystems, such as the Assembler, the FORTRAN Compiler, etc. Configuration is described in Appendix B.

## ORGANIZATION

The Symbolic Editor processes a simple programming "language" described in Section I. The language is used to create an Edit File, a series of instructions for modifying an existing ASCII code file (usually a source program). The Edit File can be read only from punched tape or from a Teleprinter keyboard.

After being loaded into core and placed in operation (Section II), the S.E. reads the Edit File and stores the edit instructions in available memory; then the Editor reads the symbolic file input, makes the changes requested in the Edit File, and produces the appropriate symbolic file output.

As it reads the Edit File, the S.E. checks each edit instruction for errors. Valid instructions are stored in available core; erroneous instructions cause an error message to be printed and are omitted from storage in core. The entire Edit File is checked and valid instructions are stored without operator intervention. If an **EDIT OVERFLOW error (see Section III) is encountered, the S.E. terminates the edit cycle.

As it reads each symbolic file input, the S.E. counts each line and each character. These counts determine where changes requested in the Edit File are to be made. During this process of counting and making changes, other errors may be found. Valid edit instructions are executed. Erroneous instructions cause an error message to be printed and are deleted from the edit cycle, but will not prevent the S.E. from completing the edit cycle. If any error messages are printed, after the edit cycle is complete the user consults the messages, writes another Edit File, and uses the S.E. again to accomplish further changes, if needed.

If the S.E. is only to copy or list a symbolic file input, there are slight differences in the above procedure. The Edit File consists only of one or a few edit instructions related to the entire symbolic file input and so will probably not have any errors; the output is identical to the input.

## Multiple Input Editing

Four types of multiple inputs are provided for:

> *Multi-tape Edit File*, in which the Edit File is read from more than one punched tape. Any number of Edit File tapes can be included in a single edit cycle, as long as the total number of edit instructions does not cause an **EDIT OVERFLOW error (see Section III).

*Multi-tape Symbolic File Input,* in which the symbolic file input is read from more than one punched tape. The S.E. continues the line numbering sequence from one input tape to the next to produce a single symbolic file output.

*Multiple Symbolic File Input,* in which the Edit File includes edit instructions for more than one symbolic file input. In this scheme, more than one symbolic file input can be processed in the same edit cycle. Each symbolic file input is submitted one-at-a-time, and the S.E. maintains a separate line numbering sequence for each.

*Both Multiple and Multi-tape Symbolic File Inputs,* in which more than one symbolic file input can be processed in the same edit cycle and each may be submitted from more than one punched tape.

Operation of the S.E. for multiple inputs is similar to that for single inputs. Only slight differences in the keyboard entries from the user are required, as described in Section II.

## CARD DECK PROCESSING

The Symbolic Editor can process punched or marked cards instead of punched tape. The user prepares the S.E. for card processing by configuring it with an SIO driver for the appropriate card reader device instead of for a punched tape reader. The S.E. then reads Edit Files from cards or the keyboard, and symbolic file inputs from cards. However, card processing has two limitations:

a. The symbolic file outputs are on punched tape or mass storage only.

b. Neither the Edit File nor any symbolic file input can be read from punched tape.

The Symbolic Editor is intended primarily for users whose software is normally contained on punched tape. Therefore, this manual is written primarily for those users, with descriptions for card processing in Appendix C.

# SECTION I
# THE EDIT FILE

A user controls the Symbolic Editor by following the rules of a relatively simple programming language to write an Edit File. In fact, these rules can be used to introduce a novice to some of the concepts of a normal programming language.

## BASIC ELEMENTS

Each line in an Edit File must be either an *Edit Control* line or an *Edit Data* line.

An Edit Control line must begin with a slash (/) and end with the RETURN then LINE FEED codes (or keys). The line must also contain at least an *edit code* to identify the action to be taken, and usually includes an *operand* to identify the part or parts of the symbolic file input to be affected by the action. Ten edit codes are available, as shown in the following pages. In any operand the $\ell$ (line) number can be no larger than $9999_{10}$ and the $c$ (character) number can be no larger than $72_{10}$.

Within a given Edit File, and within any given Edit Control line, the $\ell$ numbers must be unique and in ascending order. Within any given Edit Control line the $c$ numbers must be unique and, if more than one is used, in ascending order.

An Edit Data line must follow any Edit Control line that has an edit code for insertion or replacement. The Edit Data line supplies the new characters to be used in the insertion or replacement action. All characters required must appear in the Edit Data line; spaces (blank characters) are considered characters.

A slash (/) in the first character position of any Edit File line always signals the beginning of an Edit Control line. For this reason the first character in an Edit Data line can never be a slash; if one is needed in that position, it must be preceded by an exclamation point (!).

# /I,ℓ

Purpose:

To insert one or more lines following a symbolic file input line.

Format:

/I,ℓ

where ℓ is the decimal number of the line after which the insertion is to be made. The value ℓ cannot be Ø.

Comments

This Edit Control line must be followed by at least one Edit Data line that includes all necessary characters, including spaces. More than one Edit Data line may be used following a single "/I,ℓ" line to insert more than one new line into the symbolic file.

# /D,ℓ

PURPOSE:

To delete one or more lines from the symbolic file input.

Format

/D,ℓ
or
/D,ℓ1,ℓ2

where ℓ is the decimal number of the single line to be deleted, or
ℓ1 is the first number, and
ℓ2 is the last number of a consecutive series of lines to be deleted.

Comments

The intent of the Edit Control line is only to delete one or more lines of the symbolic file input. Thus, it is normally followed by another Edit Control line for any other edit operation.

If a "/D,ℓ" Edit Control line is inadvertently followed by one or more Edit Data lines (that do not start with a slash in the first character position), it will act as a "/R,ℓ" Edit Control line, as described on the next page.

# /R,ℓ

## Purpose

To replace one or more lines of the symbolic file input by one or more new lines.

## Format:

/R,ℓ

or

/R,ℓ1,ℓ2

where ℓ is the decimal number of the single line to be replaced, or

ℓ1 is the first number, and

ℓ2 is the last number of a consecutive series of lines to be replaced.

## Comments

This Edit Control line must be followed by at least one Edit Data line that includes all necessary characters, including spaces. More than one Edit Data line may be used following a single "/R,ℓ" or "/R,ℓ1,ℓ2" line to replace any number of consecutive lines by any number of replacement lines.

# /CI,ℓ,c

Purpose:

To insert one or more characters following a character within a single symbolic file input line.

Format:

/CI,ℓ,c

where ℓ is the decimal number of the line, and
c is the decimal number of the character after which the insertion is to be made. The value c cannot be Ø.

Comments

This Edit Control line must be followed by only one Edit Data line that includes only the characters to be inserted, including spaces (blank characters).

The number of characters that can be inserted depends on the number of characters in the original symbolic file input line. After the insertion, the total number of characters, including spaces, cannot exceed $72_{10}$.

# /CD,ℓ,c

Purpose:

To delete one or more characters from a line in the symbolic file input.

Format:

/CD,ℓ,c
or
/CD,ℓ,c1,c2

where ℓ is the decimal number of the line,
c is the decimal number of a single character to be deleted, or
c1 is the first number and
c2 is the last number of a consecutive series of characters to be deleted.

Comments

In normal use this Edit Control line only deletes one or more characters from a given line in the symbolic file input. Thus, it is normally followed by another Edit Control line for any other edit operation.

If a "/CD,ℓ,c" Edit Control line is inadvertently followed by an Edit Data line (that does not begin with a slash in the first character position), it will act as a "/CR,ℓ,c" Edit Control line, as described on the next page.

# /CR,ℓ,c

Purpose:

To replace one or more characters within a line of the symbolic file input by one or more new characters.

Format:

/CR,ℓ,c

or

/CR,ℓ,c1,c2

where ℓ is the decimal number of the line,

c is the decimal number of a single character to be replaced, or

c1 is the first number, and

c2 is the last number of a consecutive series of characters to be replaced.

Comments

This Edit Control line must be followed by only one Edit Data line that includes only the characters to be used as replacements, including spaces. Any number of characters can be replaced by any number of new characters, as long as the new characters do not produce a new line length greater than $72_{10}$ characters.

# /F,*f*

Purpose:

To identify one particular file in a series of multiple symbolic file inputs and to specify how many are to be included in the series.

Format:

/F,*f*

where *f* is the symbolic file input decimal number, in the order of the submittal during the edit cycle. The value *f* must be greater than 1.

Comments

This Edit Control line performs one of several functions, depending on its relative position within the Edit File:

a. If it is followed by other Edit File lines, it directs the Symbolic Editor to apply those following lines to that particular symbolic file input. The series of following Edit File lines is terminated by the next "/F,*f*" Edit Control line or by an "/E" Edit Control line.

b. If it is the last line in the Edit File (before the Edit File terminator "/E"), it directs the Symbolic Editor to process that many symbolic file inputs and only copy those files from the current file through file *f*.

c. If it is the only entry in the Edit File (other than than the Edit File terminator "/E"), it directs the Symbolic Editor only to copy that many symbolic file inputs.

The user informs the Symbolic Editor, by methods described in Section II, when each symbolic file input is complete.

It is not necessary to include a "/F,*f*" Edit Control line for all numbers within a range of *f* numbers. And, the entry "/F,1" is never needed since it is implicit. Any symbolic file input not named in a "/F,*f*" Edit Control line is copied without changes.

Within a given Edit File all *f* numbers must be unique and in ascending order.

# /L

Purpose:

To list the contents of a symbolic file input with a sequence number for each line of that input.

Format:

/L

or

/L,*f*

where *f* is used only with a series of multiple symbolic file inputs from mass storage, to specify the particular file of the series that is to be listed.

Comments

This Edit Control line must be the only entry in the Edit File (other than the Edit File terminator "/E"). Thus, the Edit File for a listing is best entered by typing it on the keyboard.

The form "/L,*f*" can be used only with a series of multiple symbolic file inputs from mass storage, to request a listing of a particular symbolic file input in the series. The entry "/L,1" need never be used, it is implicit in the entry "/L". If a "/L,*f*" Edit Control line is used for a symbolic file input from punched tape, the Symbolic Editor will consider it to be a "**CS ERR" condition (see Section III).

The format of a listing output prints a sequence number, two spaces, then the line itself, for each line of the symbolic file input.

## /↑

Purpose:

To delete an Edit File line.

Format:

/↑

Comments

This Edit Control line deletes the preceding Edit File line, regardless of whether that preceding line is an Edit Control line or an Edit Data line. Whenever the Symbolic Editor reads a "/↑" Edit Control line from either the keyboard or from a punched tape Edit File, it immediately prints a copy of the Edit File line deleted, the message "CONTROL STATEMENT DELETED!", and an asterisk (*). The Symbolic Editor prints the asterisk to signal that it will accept a new Edit File line. If the Edit File is being read from punched tape, the Symbolic Editor will immediately begin reading that tape again. If the Edit File is being typed, the user can then enter the next Edit File line.

A series of "/↑" Edit Control lines can be used to delete a series of preceding Edit File lines. In fact, this technique could be used to delete the entire Edit File. Deletion of the entire Edit File is indicated when the Symbolic Editor's deletion messages report that a "/↑" Edit Control line has been deleted.

# /E

Purpose:

To terminate the Edit File, or to copy the symbolic file input.

Format:

/E

Comments

This Edit Control line terminates the Edit File, if there is a series of other Edit Control lines in the Edit File. If "/E" is the only entry in the Edit File, it directs the Editor to copy the symbolic file input without making any changes.

# SECTION II
# OPERATING PROCEDURES

Before operation, the Symbolic Editor must be loaded into core memory by either of two methods: (1) If used as a "free-standing" program, it is loaded directly from punched tape, or (2) it can be made part of an operating system and subsequently loaded from mass storage. This section describes both methods of starting Editor operation, and how to control the Editor for various capabilities. At the end of the section is a summary in the form of an Operating Procedures Flowchart. That flowchart has unique page numbers so that it can be removed from and used independently of this manual.

## LOADING FROM PUNCHED TAPE

When the Symbolic Editor is loaded from punched tape, it must also be configured with a set of SIO (Software Input/Output) drivers, one for each I/O device with which it is to be used. This configuration is made on either a temporary or permanent basis. (See Appendix B).

### Temporary I/O Configuration

This technique allows the Symbolic Editor to be used on any HP computer, rather than one with a specific arrangement of I/O devices. Briefly, the method consists of two phases: (1) Loading the unconfigured Symbolic Editor tape and (2) loading and configuring* an SIO driver for each I/O device needed. The user completes these two phases by performing steps 1 through 5 in Appendix B. Then he starts Symbolic Editor operation by setting the computer to the Editor's starting address (LOAD ADDRESS) $100_8$ and pressing PRESET, then RUN.

---

*To configure an SIO driver is to identify the Select Code of its device.

## Permanent I/O Configuration

This technique produces a configured Symbolic Editor tape that eliminates the need to load and configure the SIO drivers again. The configured tape contains a copy of the Symbolic Editor program and a copy of each SIO driver with its Select Code identifications. The configured Symbolic Editor tape can then be used on the computer that produced it, or on any other HP computer that has the same (or similar) I/O devices in the same Select Code (I/O slot) locations.

The method consists of three phases: (1) loading the unconfigured Symbolic Editor tape, (2) loading and configuring an SIO driver for each I/O device to be used, and (3) loading and executing an SIO System Dump program. The user completes these three phases by performing steps 1 through 9 of Appendix B. Then he starts Symbolic Editor operation by loading the configured Symbolic Editor tape through the Basic Binary Loader, setting the computer to the Symbolic Editor's starting address (LOAD ADDRESS) $100_8$, and pressing PRESET, then RUN.

## LOADING FROM AN OPERATING SYSTEM

The Symbolic Editor can be made a part of an operating system, such as the HP Magnetic Tape System, by using the procedure given in the *Prepare Tape System* manual (HP 02116-91751). Then the user can use steps given in that system's manual (see *Magnetic Tape System*, HP 02116-91752) to load the Symbolic Editor into core, configure it with a set of SIO drivers, and start operation, all automatically.

## NORMAL EDITING

Once the Symbolic Editor has been configured, loaded, and started, the user assumes control of its functions. The first medium for control is the Edit File, as described in Section I. However, the Edit File alone is not enough. During each edit cycle the user must exchange information with the Editor through a series of execution control messages. That is, he must identify the source of the Edit File, the source of the symbolic file input(s), and the destination of the symbolic file output(s) at the times when the Editor needs the information.

The user's execution control messages are taken from a vocabulary of six words and are entered, at appropriate times, on the Teleprinter keyboard. The format of these words is similar to that of Edit Control lines in an Edit File, however they do not perform the same functions. Five of the execution control words have different applications within a given edit cycle. The applications of all six words are summarized in the following list, then fully described later in this Section II.

| Word | Application |
|---|---|
| /T | Specifies, in response to "EDIT FILE DEVICE?", that the Edit File is to be *typed* on the Teleprinter keyboard. |
| /P | (1) Specifies, in response to "EDIT FILE DEVICE?", that the Edit File is to be read from *punched tape*.<br>(2) Specifies, in response to "SYMBOLIC FILE SOURCE DEVICE?", that the symbolic file input(s) medium is *punched tape*.<br>(3) Specifies, in response to "SYMBOLIC FILE DESTINATION DEVICE?", that the symbolic file output(s) medium is *punched tape*. |
| /M | (1) Specifies, in response to "SYMBOLIC FILE SOURCE DEVICE?", that the symbolic file input(s) medium is *mass storage*.<br>(2) Specifies, in response to "SYMBOLIC FILE DESTINATION DEVICE?", that the symbolic file output(s) medium is *mass storage*. |
| /C | (1) Specifies, in response to "**END-OF-TAPE" and an asterisk (*) at the end of an Edit File tape, that an Edit File *control* entry is to be made on the keyboard. That control entry must be "/E" to terminate the Edit File.<br>(2) Specifies, in response to "**END-OF-TAPE" and an asterisk (*) at the end of a symbolic file input tape, that the next tape to be read is a *continuation* of the symbolic file input tape just read. |

| Word | Application |
|---|---|
| /E | (1) Specifies, after a "/C" response to "**END-OF-TAPE" and an asterisk (*) at the end of an Edit File tape, that the last tape read was the *end* of the Edit File. |
| | (2) Specifies, in response to "**END-OF-TAPE" and an asterisk (*) at the end of a symbolic file input tape, that the tape just read was the *end* of the current symbolic file input. If the Edit File contains one or more "/F,*f*" Edit Control lines, the edit cycle is for multiple symbolic file inputs which allows the entry of "GO" (see below) instead of "/E" to signal the end of a symbolic file input. |
| GO | (1) Specifies, in response to "**END-OF-TAPE" and an asterisk (*) at the end of a symbolic file input tape, that the last tape read was the end of the current symbolic file input and that the edit cycle is to *go ahead*. The Symbolic Editor then either starts processing for the next symbolic file input or ends the edit cycle if all symbolic file inputs requested in the Edit File have been processed. |
| | (2) Specifies, in response to "TEAR PUNCHED PAPER TAPE" and an asterisk (*), that the tape is torn and the edit cycle is to *go ahead* (as described above). |

## Execution Control Messages

Symbolic Editor (S.E.) operation begins when it starts the series of execution control messages. First, it prints "HP SYMBOLIC EDITOR" and "EDIT FILE DEVICE?". The user must then inform the Symbolic Editor which device he will use to submit the Edit File.

To read the Edit File from punched tape, the user places that tape into the reader, types "/P", and presses the RETURN then LINE FEED keys. The Symbolic Editor immediately reads the tape.

To type the Edit File on the Teleprinter keyboard, the user types "/T", then presses the RETURN and LINE FEED keys. The Symbolic Editor prints an asterisk (*) to signal that it is ready to accept entries on the keyboard. The user then types the Edit File.

Whenever the Symbolic Editor prints an asterisk (*), it is requesting the user to type some entry on the keyboard. In most cases, but not all, the user has some other action(s) to perform before typing an entry. All appropriate action(s) and keyboard entries are described in this section.

The user always signals the end of each line of typed entries by pressing the RETURN then LINE FEED keys. Or, if he makes a mistake, he presses the RUB OUT key, then the RETURN and LINE FEED keys. If he completes an entry that the Symbolic Editor cannot accept, the Symbolic Editor rejects that entry and prints one or more messages to signal that rejection. For example, it might repeat the last request message; or, it might print the asterisk (*) again. In any case, the user will have no trouble understanding the rejection signal.

After the Symbolic Editor has detected the end of the Edit File, it prints "SYMBOLIC FILE SOURCE DEVICE?". To read the symbolic file input from punched tape, the user places the tape into the reader, readies the reader, and types "/P". To read the symbolic file input from mass storage, the mass storage device should already be prepared for use; the user types "/M".

If the Edit File requests an edit or copy operation, the Symbolic Editor then prints "SYMBOLIC FILE DESTINATION DEVICE?". To direct the output to punched tape, the user turns on the tape punch device then types "/P"; the Symbolic Editor immediately begins to read the input and punch the output. To direct the output into mass storage (allowed only if the input is read from punched tape), the user types "/M"; the Symbolic Editor immediately begins to read the input and write the output.

If the Edit File requests a list operation, see "LISTING", later in this Section II.

When the Symbolic Editor reads the end of the input, it prints "**END-OF-TAPE", then an asterisk (*) to signal that it will accept an entry on the keyboard. For a "normal" edit operation in which only one symbolic file input tape is to be processed and that input is complete on the tape just read, the user types "/E". For any "abnormal" operation, see the following subsections.

Examples of Edit Files, Symbolic File Inputs, Symbolic File Outputs, and execution control messages are presented in Appendix A.

## MULTI-TAPE EDIT FILE

The user can elect to read the Edit File from more than one punched tape. To do so, he should ensure that only the last tape to be read for the multi-tape Edit File ends with "/E", the Edit File terminator. At the end of each Edit File tape that does not end with "/E" the Symbolic Editor prints "**END-OF-TAPE", then an asterisk (*) to signal that it will accept an entry on the keyboard. To read another Edit File tape, the user places that next tape into the reader, readies the reader, types "GO", and presses the RETURN then LINE FEED keys. Or, the user can elect to terminate the Edit File at the end of any Edit File tape. He does that by typing "/C" on one line, then "/E" on the next line.

## MULTI-TAPE SYMBOLIC FILE INPUT

At the end of each symbolic file input read from punched tape, the S.E. prints "**END-OF-TAPE" and "*" to signal that it will accept entries on the keyboard.

To continue the current symbolic file input from another punched tape, the user places that next tape into the reader, readies the reader, types "/C", and presses the RETURN then LINE FEED keys. The Symbolic Editor immediately reads that tape and produces the output as a continuation of the previous output. This cycle can be repeated for any number of tapes, but the user should remember one limitation: The

Symbolic Editor cannot make changes to any line number greater than 9999; after that number it can only copy.

When the last tape of a multi-tape symbolic file input has been read, the user answers the "**END-OF-TAPE" and asterisk messages by typing "/E" or "GO".

This feature can also be used in conjunction with the Multiple Symbolic File Inputs feature, as described in the following subsection.

## MULTIPLE SYMBOLIC FILE INPUTS

To use the Multiple Symbolic File Inputs feature, the Edit File must include one or more "/F,*f*" Edit Control lines. (See Section I.) The value *f* specified in the last (or only) "/F,*f*" entry instructs the Symbolic Editor to process that many complete symbolic file inputs and end the edit cycle after the last one is complete.

When the symbolic file inputs (SFIs) are being read from punched tapes, at the end of each tape read the Symbolic Editor prints "**END-OF-TAPE" and an asterisk to signal that it will accept an entry on the keyboard. Thus, the user is given two choices:

(1) To signal the end of the current SFI and begin processing for the next SFI (if any). To do so, he places the next SFI tape into the reader, readies the reader, types "GO" and presses the RETURN then LINE FEED keys. The Symbolic Editor action that results depends on the symbolic file output destination:

If the outputs are on punched tape, the Symbolic Editor punches a trailer at the end of the current output then prints "TEAR PUNCHED PAPER TAPE" and an asterisk (*). The user should then tear off that punched tape, type "GO", and press the RETURN then LINE FEED keys. The Symbolic Editor then begins to process the next input.

If the outputs are into mass storage, the Symbolic Editor writes an EOF (End-Of-File mark) into mass storage, and begins to process the next input.

(2) To continue the current symbolic file input by using the Multi-Tape Symbolic File Input feature described in the preceding subsection.

When the symbolic file inputs are being read from mass storage, the Symbolic Editor assumes that each is complete. That is, the user is given no opportunity to continue an SFI. Instead, the Symbolic Editor prints "TEAR PUNCHED PAPER TAPE" and an asterisk (*). The user should then tear off that punched tape output, type "GO" and press the RETURN then LINE FEED keys.

After the S.E. has processed (and, if necessary, punched trailers for) as many inputs as specified in the last "/F,*f*" entry in the Edit File, it automatically ends the edit cycle. The number of files submitted to the S.E. must always agree with the last "/F,*f*" number. If fewer files are submitted, the S.E. will not end the edit cycle; if more files are waiting to be submitted, the S.E. will end the cycle too soon.

## LISTING

To list the contents of a symbolic file input the Symbolic Editor needs to know only the Edit File Source and the symbolic file input source. Thus, the execution control messages will occur as described in the following paragraphs.

The Edit File for a listing operation is best entered by typing it on the keyboard. The user answers the Symbolic Editor request "EDIT FILE DEVICE?" by typing "/T", and pressing the RETURN then LINE FEED keys. The Symbolic Editor then prints an asterisk to signal that it is ready to accept entries on the keyboard. The user then types "/L" or "/L,*f*" on one line, then "/E" on the next line. (See Section I.)

To read the symbolic file input from punched tape, after the Symbolic Editor prints "SYMBOLIC FILE SOURCE DEVICE?" the user places the input tape into the reader, readies the reader, types "/P", and presses the RETURN then LINE FEED keys. The Symbolic Editor immediately begins to read the input tape and list the contents. At the end of the input tape the Symbolic Editor prints "**END-OF-TAPE" and an asterisk (*) to signal that it will accept an entry on the keyboard.

To continue the symbolic file input from another tape (which will be listed with line numbers continuing from the last line number printed), the user places that next tape into the reader, readies the reader, types "/C", and presses the RETURN then LINE FEED keys. This continuation cycle can be repeated for any number of tapes up to a line count of 9999.

To end the listing operation, after the Symbolic Editor prints "**END-OF-TAPE" and an asterisk, the user types "/E", and presses the RETURN then LINE FEED keys.

To read the symbolic file input from mass storage, after the Symbolic Editor prints "SYMBOLIC FILE SOURCE DEVICE?" the user types "/M" (the mass storage should already be prepared for use). The Symbolic Editor immediately begins to read the input and print the contents. At the end of the input the Symbolic Editor assumes the file is complete and it immediately ends the edit cycle.

*NOTE: The Symbolic Editor normally prints its execution control messages on the Teleprinter. However, for a listing operation the user may configure the Symbolic Editor with an SIO Line Printer driver to speed up the listing output. If so, the Symbolic Editor will print all of its messages on the Line Printer while the user types his messages on the Teleprinter. The user can avoid having to look at the Symbolic Editor messages on the line printer by using this sequence of Symbolic Editor pauses and responses:*

| Symbolic Editor Pause | User Message Required |
|---|---|
| first | "/T", followed by "/L" or "/L,*f*" (see Section I) then "/E". |
| second | To read the symbolic file from punched tape (place the tape into the reader and ready the reader), "/P".<br>To read the symbolic file from mass storage, "/M". |
| third (only if symbolic file is read from punched tape). | To terminate the listing operation, "/E".<br>To continue the listing operation from another punched tape (place the next tape into the reader and ready the reader), "/C". |

## END OF THE EDIT CYCLE

When the edit cycle is complete, the S.E. prints "*END" to indicate that it has done one of three things, depending on the operation just completed. For example, if it was producing a symbolic file output on punched tape, it has just punched the trailer for that output tape. Or, if it was producing a symbolic file output in mass storage, it has just written an EOF (End-Of-File mark) for that output then "reset" the mass storage unit to the SOT (Start-Of-Tape mark). Or, if it was listing the contents of a symbolic file input, it has just printed the last line of that listing.

If the tape punch device is on at this point, the user should turn it off.

After printing "*END" the Symbolic Editor terminates in one of two ways:

(1) If it has been loaded from punched tape, it halts at location $100_8$, its starting address. If it is to be used again, the user only need press the RUN button.

(2) If it has been loaded from an operating system, the user should consult the manual for that system (such as *Magnetic Tape System*, HP 02116-91752).

## REJECTING AN EDIT CYCLE

To reject an edit cycle at any time:

1. Press the computer's HALT button.
2. Set the computer to the Symbolic Editor's starting address $100_8$. (Set the Switch Register to $100_8$, press LOAD ADDRESS.)
3. Press RUN.

All of the operating procedures for the Symbolic Editor have been summarized in the Operating Procedures Flowchart.

# OPERATING PROCEDURES FLOWCHART

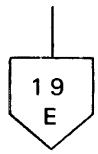
Operating Procedures for the Symbolic Editor are presented in flowchart form on the following pages. These pages are assigned unique numbers so that they can be removed from the *Symbolic Editor* manual and used as reminders for a user of the Symbolic Editor.

Do not attempt to read the entire flowchart. Rather, read only the individual symbols and lines that pertain directly to the action taking place. For example, a normal editing cycle follows the bold flow lines starting on page FC-3 and continues through page FC-11.

Symbols in this flowchart are based on the recommendations of the International Organization for Standarization (ISO) and the U.S.A. Standards Institute (USASI). In addition, continuations from one page of the flowchart to another are shown by off-page connector symbols as follows:

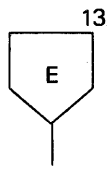
Outconnector

An outconnector is linked to an inconnector by a page number and an alphabetic character identifier, both within the outconnector symbol.




Inconnector

An inconnector is linked to an outconnector by a flowchart page number above the symbol, and an alphabetic character identifier within the symbol.



Abbreviations used in the flow chart and notes are:

BBL = Basic Binary Loader
EF = Edit File
EOF = End-of-File mark
MTS = Magnetic Tape System
PTS = Prepare Tape System
S.A. = Starting Address
S.E. = Symbolic Editor
SFI = Symbolic File Input
SFO = Symbolic File Output
SIO = Software Input/Output

NOTE 1

See the *Magnetic Tape System* manual (HP 02116-91752).

NOTE 2

See Appendix B in the *Symbolic Editor* manual (HP 02116-9016).

NOTE 3

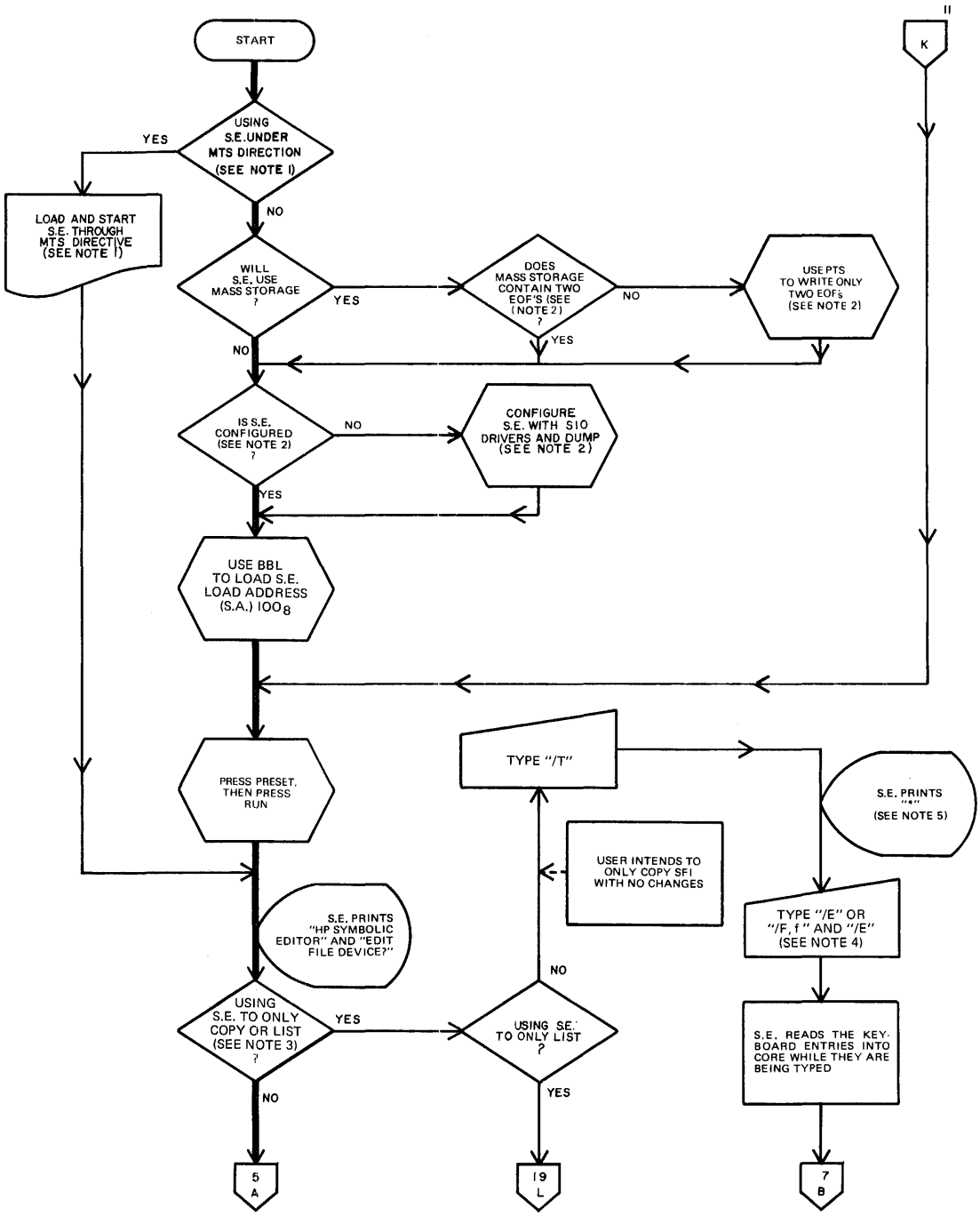
Use of the Symbolic Editor to list or to only copy is similar to edit usage. However, there are slight differences to consider, as shown in this flow chart.

NOTE 4

If the Edit File contains one or more "/F,*f*" Edit Control lines, the Multiple Symbolic File Input feature is to be used. That is, the current edit cycle is to read several symbolic file inputs from either punched tape or from mass storage. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

NOTE 5

The Symbolic Editor prints an asterisk (*) whenever it expects the user to type a message on the Teleprinter's keyboard. Before typing a message, the user may have to perform other actions. The proper actions and messages are indicated at appropriate locations within the flowchart.

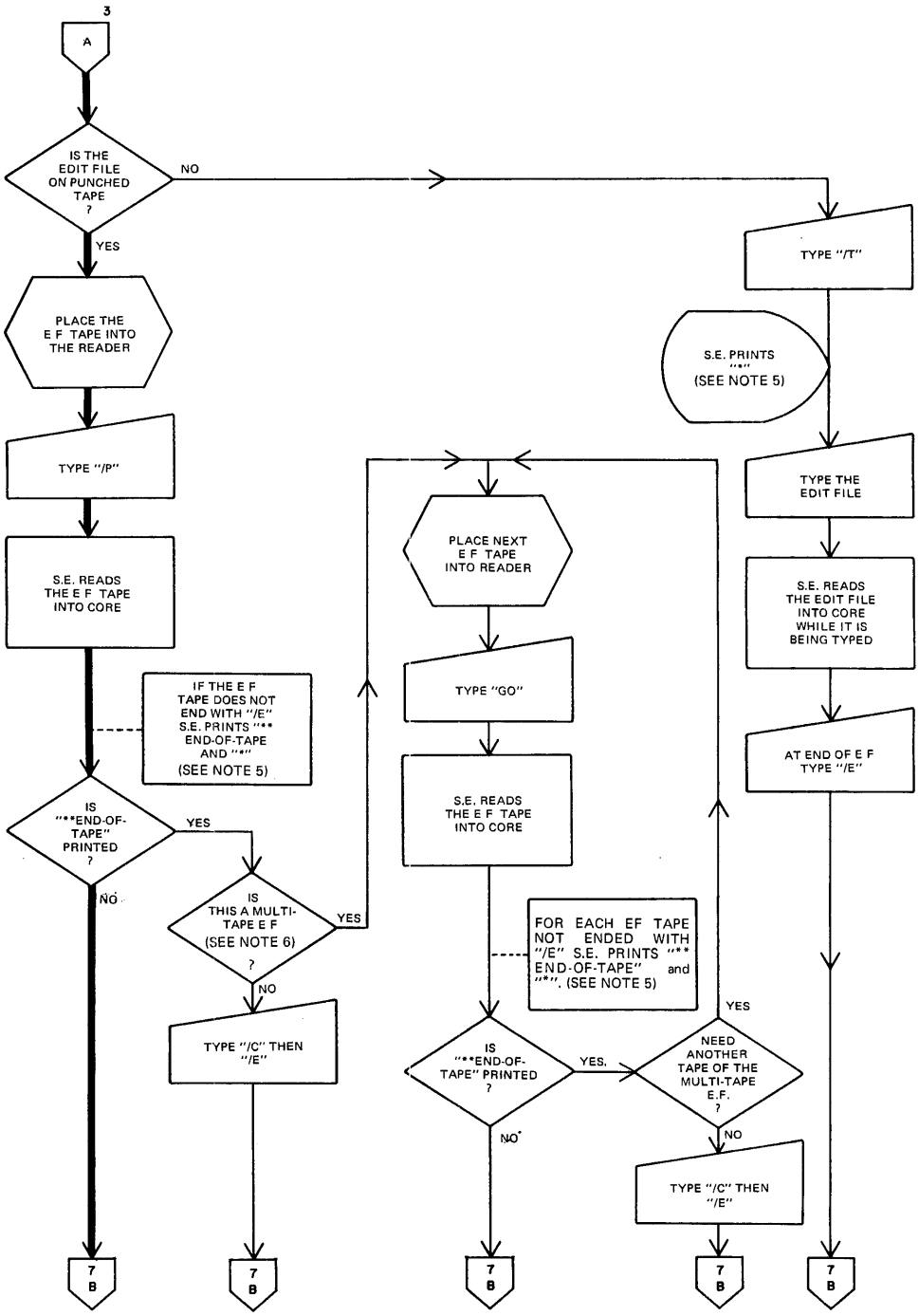

START
USING S.E. UNDER MTS DIRECTION (SEE NOTE 1)
YES
NO
LOAD AND START S.E. THROUGH MTS DIRECTIVE (SEE NOTE 1)
WILL S.E. USE MASS STORAGE ?
YES
NO
DOES MASS STORAGE CONTAIN TWO EOF'S (SEE (NOTE 2) ?
NO
YES
USE PTS TO WRITE ONLY TWO EOF's (SEE NOTE 2)
IS S.E. CONFIGURED (SEE NOTE 2) ?
NO
YES
CONFIGURE S.E. WITH SIO DRIVERS AND DUMP (SEE NOTE 2)
USE BBL TO LOAD S.E. LOAD ADDRESS (S.A.) $100_8$
PRESS PRESET, THEN PRESS RUN
S.E. PRINTS "HP SYMBOLIC EDITOR" AND "EDIT FILE DEVICE?"
USING S.E. TO ONLY COPY OR LIST (SEE NOTE 3) ?
YES
NO
USING S.E. TO ONLY LIST ?
NO
YES
TYPE "/T"
USER INTENDS TO ONLY COPY SF1 WITH NO CHANGES
S.E. PRINTS "*" (SEE NOTE 5)
TYPE "/E" OR "/F, f" AND "/E" (SEE NOTE 4)
S.E. READS THE KEY-BOARD ENTRIES INTO CORE WHILE THEY ARE BEING TYPED
K
11
5 A
19 L
7 B

NOTE 5

The Symbolic Editor prints an asterisk (*) whenever it expects the user to type a message on the Teleprinter keyboard. Before typing a message, the user may have to perform other actions. The proper actions and messages are indicated at appropriate locations within the flowchart.

NOTE 6

The Multi-Tape Edit File feature allows the Edit File to be read from more than one punched tape, if needed. The Edit File is terminated by any tape that contains "/E". If none of the Edit File tapes read contain "/E", the user must terminate the Edit File by typing "/C" on one line, then "/E" on the next line.


3
A
IS THE EDIT FILE ON PUNCHED TAPE ?
NO
YES
PLACE THE E F TAPE INTO THE READER
TYPE "/P"
S.E. READS THE E F TAPE INTO CORE
IF THE E F TAPE DOES NOT END WITH "/E" S.E. PRINTS "** END-OF-TAPE AND "*" (SEE NOTE 5)
IS "**END-OF-TAPE" PRINTED ?
YES
NO
IS THIS A MULTI-TAPE E F (SEE NOTE 6) ?
YES
NO
TYPE "/C" THEN "/E"
7 B
7 B
TYPE "/T"
S.E. PRINTS "*" (SEE NOTE 5)
TYPE THE EDIT FILE
S.E. READS THE EDIT FILE INTO CORE WHILE IT IS BEING TYPED
AT END OF E F TYPE "/E"
PLACE NEXT E F TAPE INTO READER
TYPE "GO"
S.E. READS THE E F TAPE INTO CORE
FOR EACH EF TAPE NOT ENDED WITH "/E" S.E. PRINTS "** END-OF-TAPE" and "*". (SEE NOTE 5)
IS "**END-OF-TAPE" PRINTED ?
YES.
NO
NEED ANOTHER TAPE OF THE MULTI-TAPE E.F. ?
YES
NO
TYPE "/C" THEN "/E"
7 B
7 B
7 B
7 B

NOTE 7

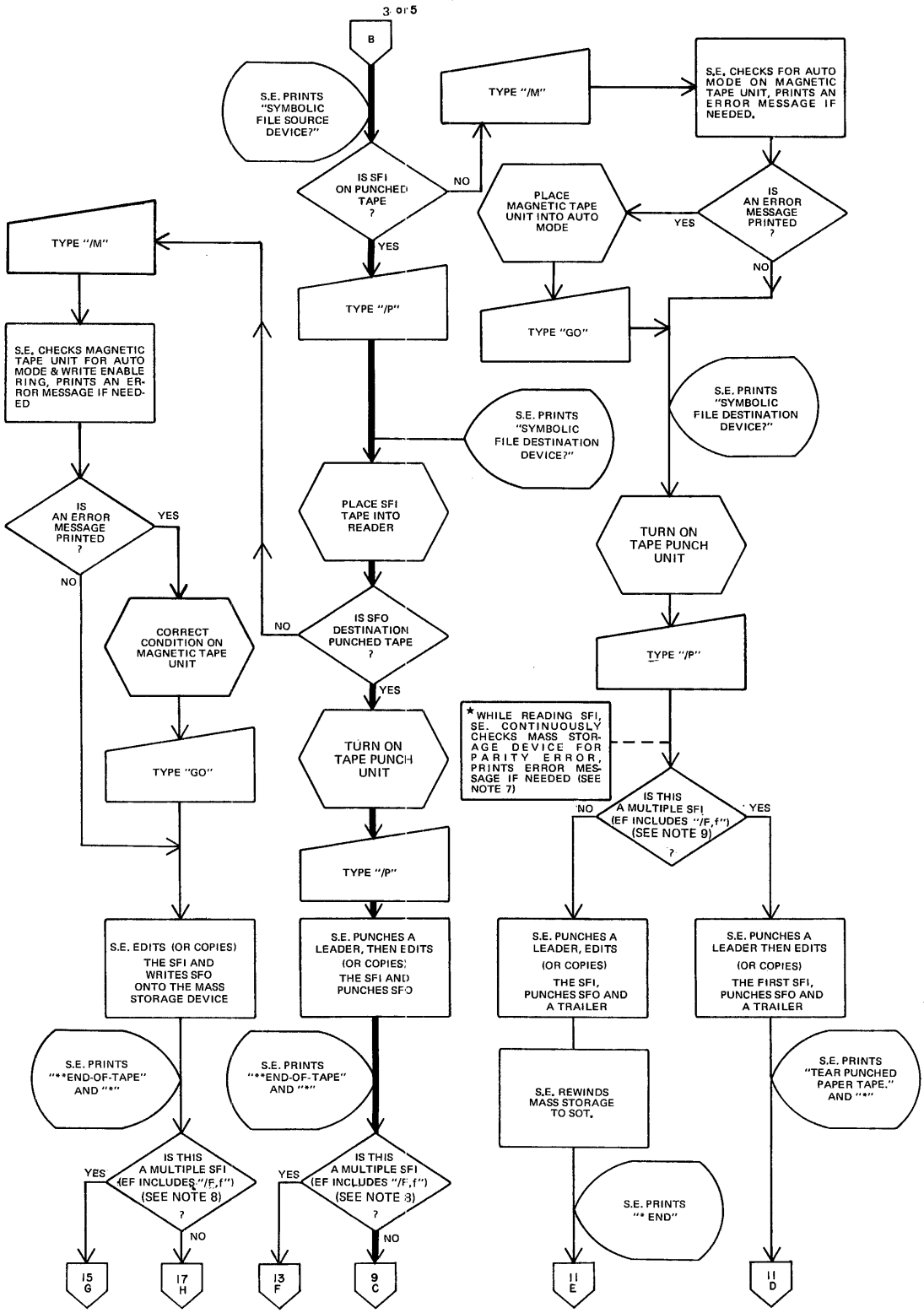
See Section III, Error Conditions, in the *Symbolic Editor* manual (HP 02116-9016).

NOTE 8

If the Edit File contains one or more "/F,*f*" Edit Control lines, the Multiple Symbolic File Input feature is to be used for inputs read from Punched tape. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

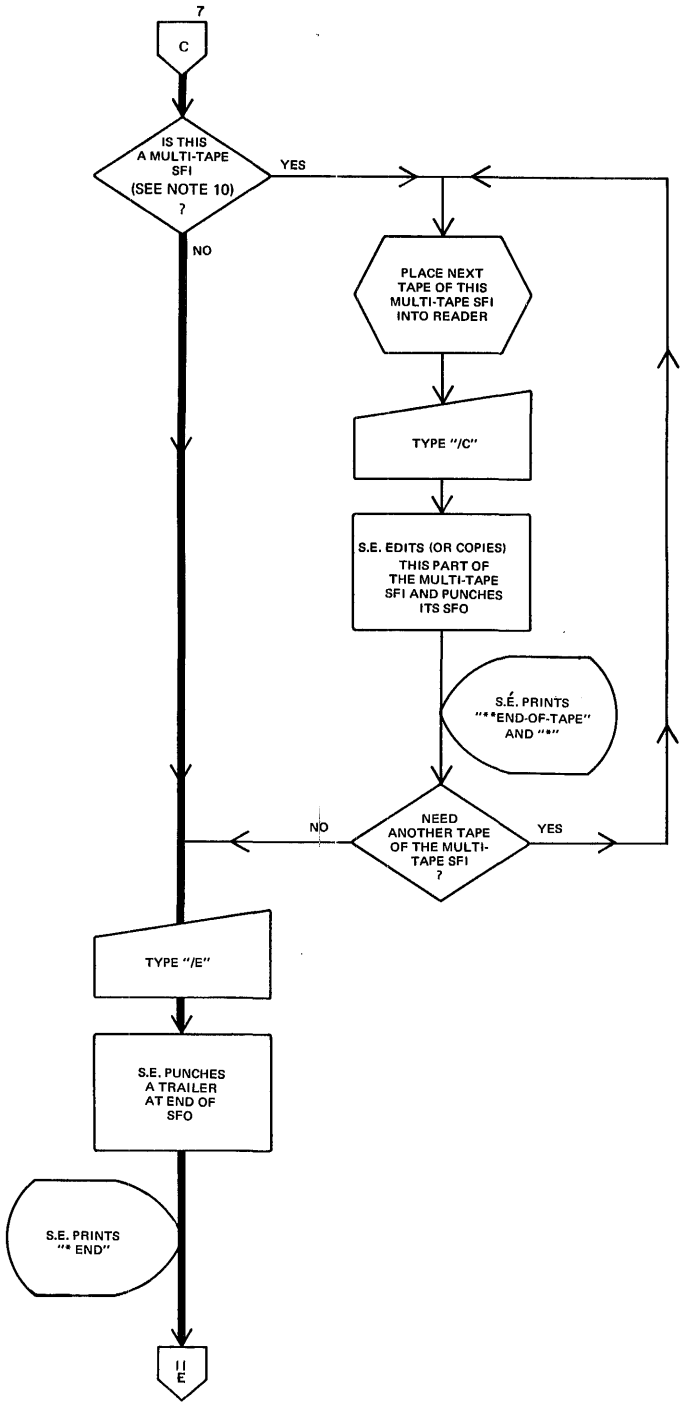
NOTE 9

If the Edit File contains one or more "/F,*f*" Edit Control lines, the Multiple Symbolic File Input feature is to be used for inputs read from mass storage. See Section II of the *Symbolic Editor* manual (HP 02116-9016).


3 of 5
B
S.E. PRINTS "SYMBOLIC FILE SOURCE DEVICE?"
IS SFI ON PUNCHED TAPE ?
NO
YES
TYPE "/M"
S.E. CHECKS FOR AUTO MODE ON MAGNETIC TAPE UNIT, PRINTS AN ERROR MESSAGE IF NEEDED.
IS AN ERROR MESSAGE PRINTED ?
YES
NO
PLACE MAGNETIC TAPE UNIT INTO AUTO MODE
TYPE "GO"
TYPE "/P"
S.E. PRINTS "SYMBOLIC FILE DESTINATION DEVICE?"
TYPE "/M"
S.E. CHECKS MAGNETIC TAPE UNIT FOR AUTO MODE & WRITE ENABLE RING, PRINTS AN ERROR MESSAGE IF NEEDED
IS AN ERROR MESSAGE PRINTED ?
YES
NO
CORRECT CONDITION ON MAGNETIC TAPE UNIT
TYPE "GO"
PLACE SFI TAPE INTO READER
IS SFO DESTINATION PUNCHED TAPE ?
NO
YES
TURN ON TAPE PUNCH UNIT
TYPE "/P"
TURN ON TAPE PUNCH UNIT
TYPE "/P"
*WHILE READING SFI, SE. CONTINUOUSLY CHECKS MASS STORAGE DEVICE FOR PARITY ERROR, PRINTS ERROR MESSAGE IF NEEDED (SEE NOTE 7)
IS THIS A MULTIPLE SFI (EF INCLUDES "/F,f") (SEE NOTE 9) ?
NO
YES
S.E. EDITS (OR COPIES) THE SFI AND WRITES SFO ONTO THE MASS STORAGE DEVICE
S.E. PUNCHES A LEADER, THEN EDITS (OR COPIES) THE SFI AND PUNCHES SFO
S.E. PUNCHES A LEADER, EDITS (OR COPIES) THE SFI, PUNCHES SFO AND A TRAILER
S.E. PUNCHES A LEADER THEN EDITS (OR COPIES) THE FIRST SFI, PUNCHES SFO AND A TRAILER
S.E. PRINTS "**END-OF-TAPE" AND "*"
S.E. PRINTS "**END-OF-TAPE" AND "*"
S.E. REWINDS MASS STORAGE TO SOT.
S.E. PRINTS "TEAR PUNCHED PAPER TAPE." AND "*"
IS THIS A MULTIPLE SFI (EF INCLUDES "/F,f") (SEE NOTE 8) ?
YES
NO
IS THIS A MULTIPLE SFI (EF INCLUDES "/F,f") (SEE NOTE 8) ?
YES
NO
S.E. PRINTS "* END"
15 G
17 H
13 F
9 C
11 E
11 D

NOTE 10

The Multi-Tape Symbolic File Input allows any punched tape SFI to be read from more than one tape, if needed. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

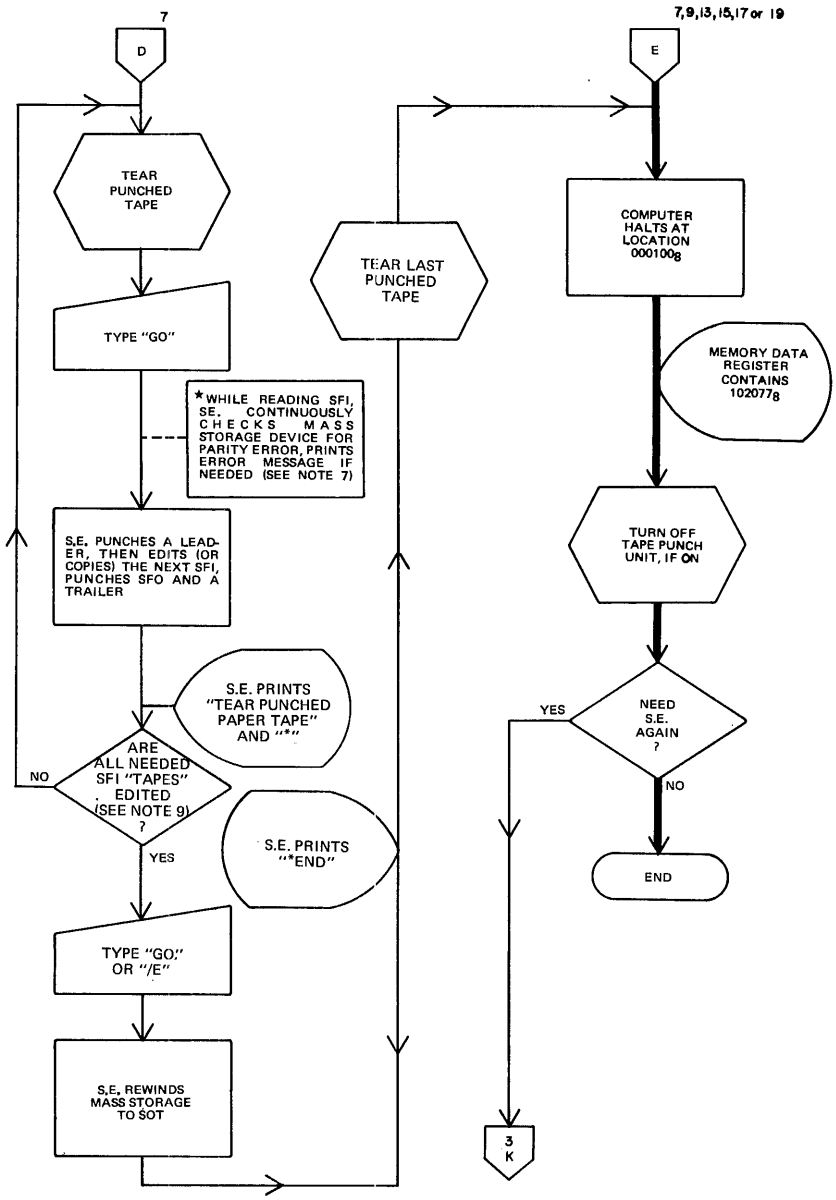

7
C
IS THIS A MULTI-TAPE SFI (SEE NOTE 10) ?
YES
NO
PLACE NEXT TAPE OF THIS MULTI-TAPE SFI INTO READER
TYPE "/C"
S.E. EDITS (OR COPIES) THIS PART OF THE MULTI-TAPE SFI AND PUNCHES ITS SFO
S.E. PRINTS "**END-OF-TAPE" AND "*"
NEED ANOTHER TAPE OF THE MULTI-TAPE SFI ?
NO
YES
TYPE "/E"
S.E. PUNCHES A TRAILER AT END OF SFO
S.E. PRINTS "* END"
11
E

NOTE 7

See Section III, Error Conditions, in the *Symbolic Editor* manual (HP 02116-9016).

NOTE 9

If the Edit File contains one or more "/F,*f*" Edit Control lines, the Multiple Symbolic File Input feature is to be used for inputs read from mass storage. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

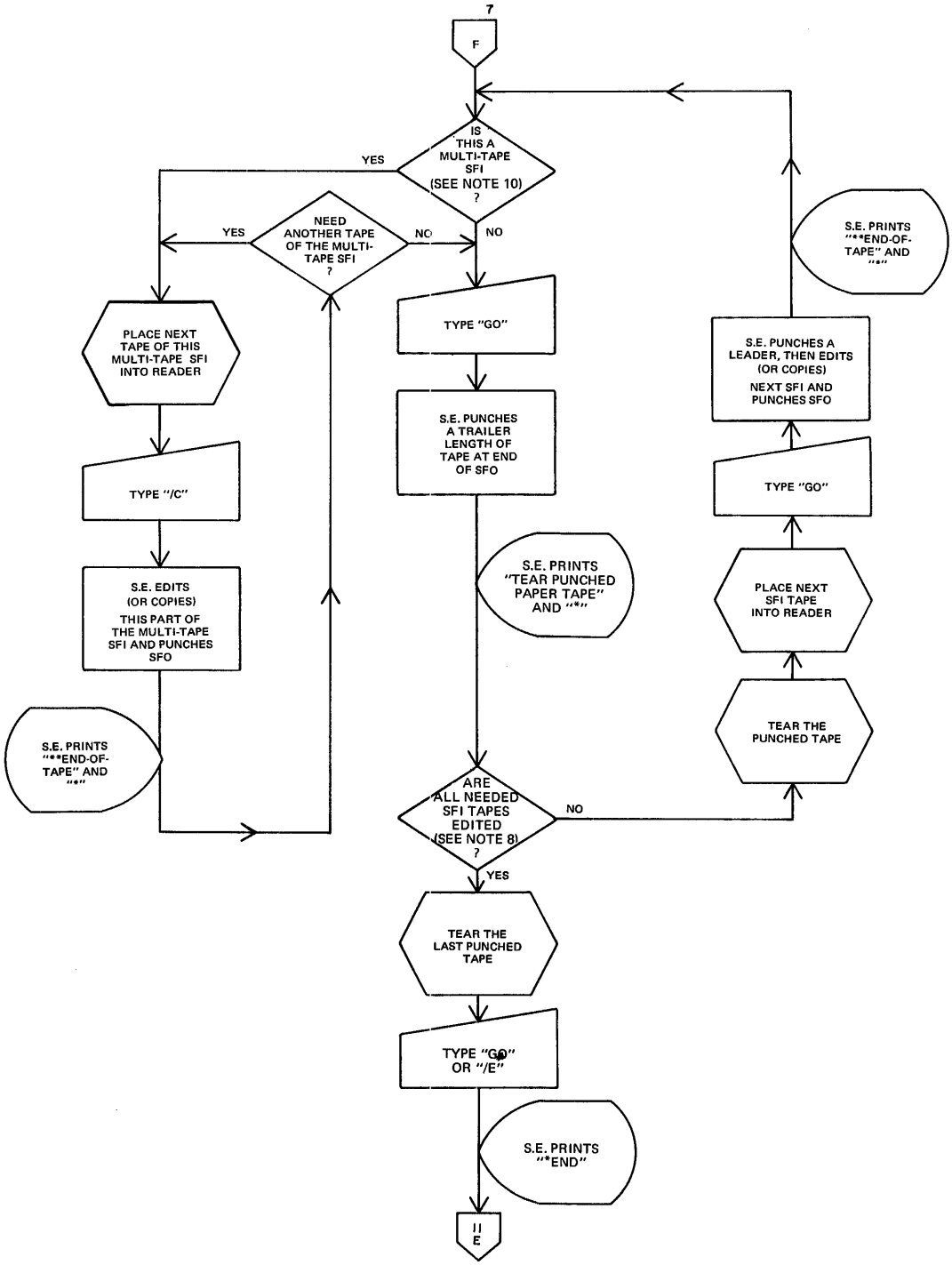

7
D
TEAR PUNCHED TAPE
TYPE "GO"
*WHILE READING SFI, SE. CONTINUOUSLY CHECKS MASS STORAGE DEVICE FOR PARITY ERROR, PRINTS ERROR MESSAGE IF NEEDED (SEE NOTE 7)
S.E. PUNCHES A LEADER, THEN EDITS (OR COPIES) THE NEXT SFI, PUNCHES SFO AND A TRAILER
S.E. PRINTS "TEAR PUNCHED PAPER TAPE" AND "*"
ARE ALL NEEDED SFI "TAPES" EDITED (SEE NOTE 9) ?
NO
YES
S.E. PRINTS "*END"
TYPE "GO" OR "/E"
S.E. REWINDS MASS STORAGE TO SOT
TEAR LAST PUNCHED TAPE
7,9,13,15,17 or 19
E
COMPUTER HALTS AT LOCATION $000100_8$
MEMORY DATA REGISTER CONTAINS $102077_8$
TURN OFF TAPE PUNCH UNIT, IF ON
NEED S.E. AGAIN ?
YES
NO
END
3
K

NOTE 8

If the Edit File contains one or more "/F,*f*" Edit Control lines, the Multiple Symbolic File Input feature is to be used for inputs read from punched tape. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

NOTE 10

The Multi-Tape Symbolic File Input allows any punched tape SFI to be read from more than one tape, if needed. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

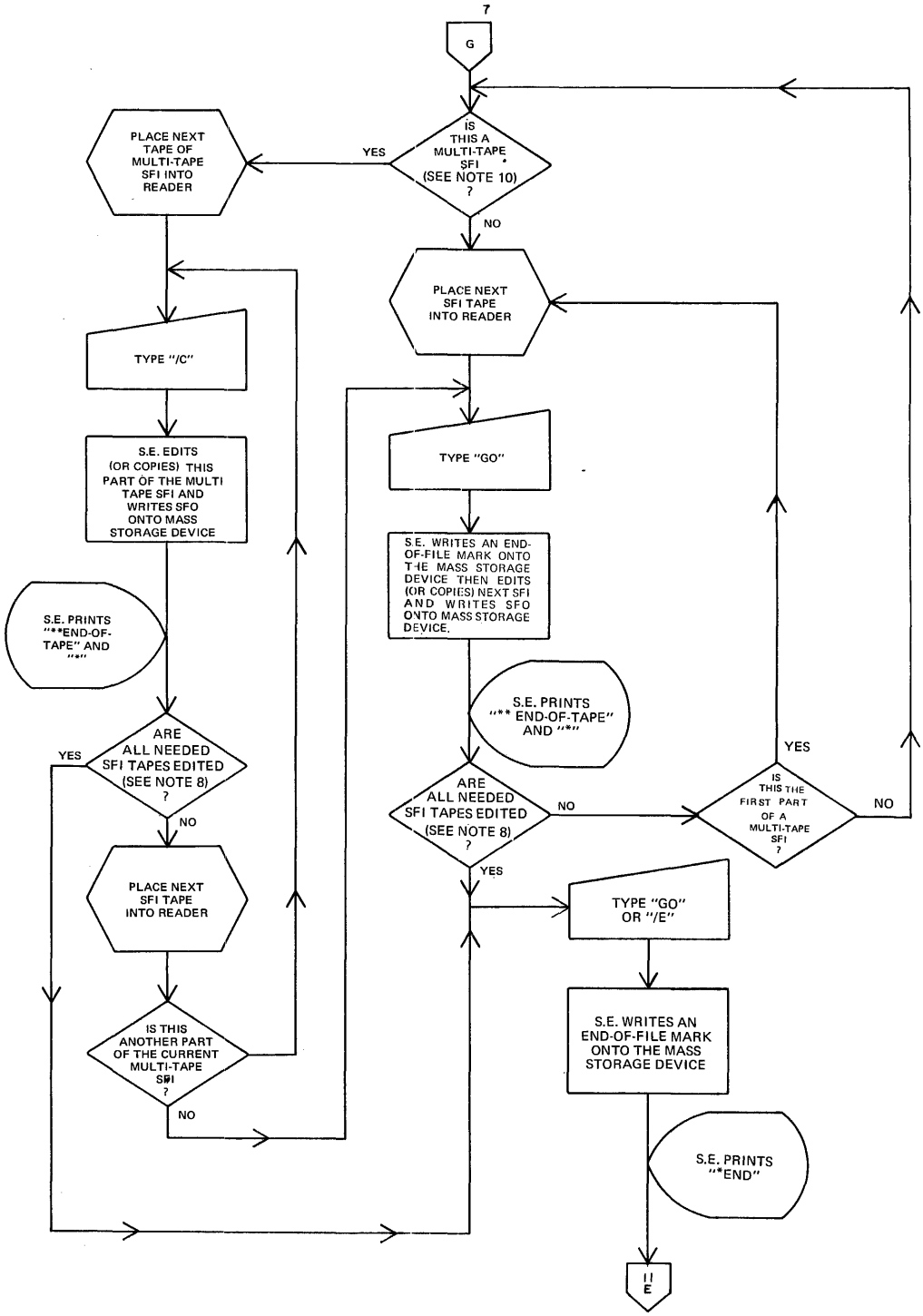

7
F
IS THIS A MULTI-TAPE SFI (SEE NOTE 10) ?
YES
NO
NEED ANOTHER TAPE OF THE MULTI-TAPE SFI ?
YES
NO
PLACE NEXT TAPE OF THIS MULTI-TAPE SFI INTO READER
TYPE "/C"
S.E. EDITS (OR COPIES) THIS PART OF THE MULTI-TAPE SFI AND PUNCHES SFO
S.E. PRINTS "**END-OF-TAPE" AND "*"
TYPE "GO"
S.E. PUNCHES A TRAILER LENGTH OF TAPE AT END OF SFO
S.E. PRINTS "TEAR PUNCHED PAPER TAPE" AND "*"
ARE ALL NEEDED SFI TAPES EDITED (SEE NOTE 8) ?
NO
YES
TEAR THE PUNCHED TAPE
PLACE NEXT SFI TAPE INTO READER
TYPE "GO"
S.E. PUNCHES A LEADER, THEN EDITS (OR COPIES) NEXT SFI AND PUNCHES SFO
S.E. PRINTS "**END-OF-TAPE" AND "*"
TEAR THE LAST PUNCHED TAPE
TYPE "GO" OR "/E"
S.E. PRINTS "**END"
II
E

NOTE 8

If the Edit File contains one or more "/F,*f*" Edit Control lines, the Multiple Symbolic File Input feature is to be used for inputs read from punched tape. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

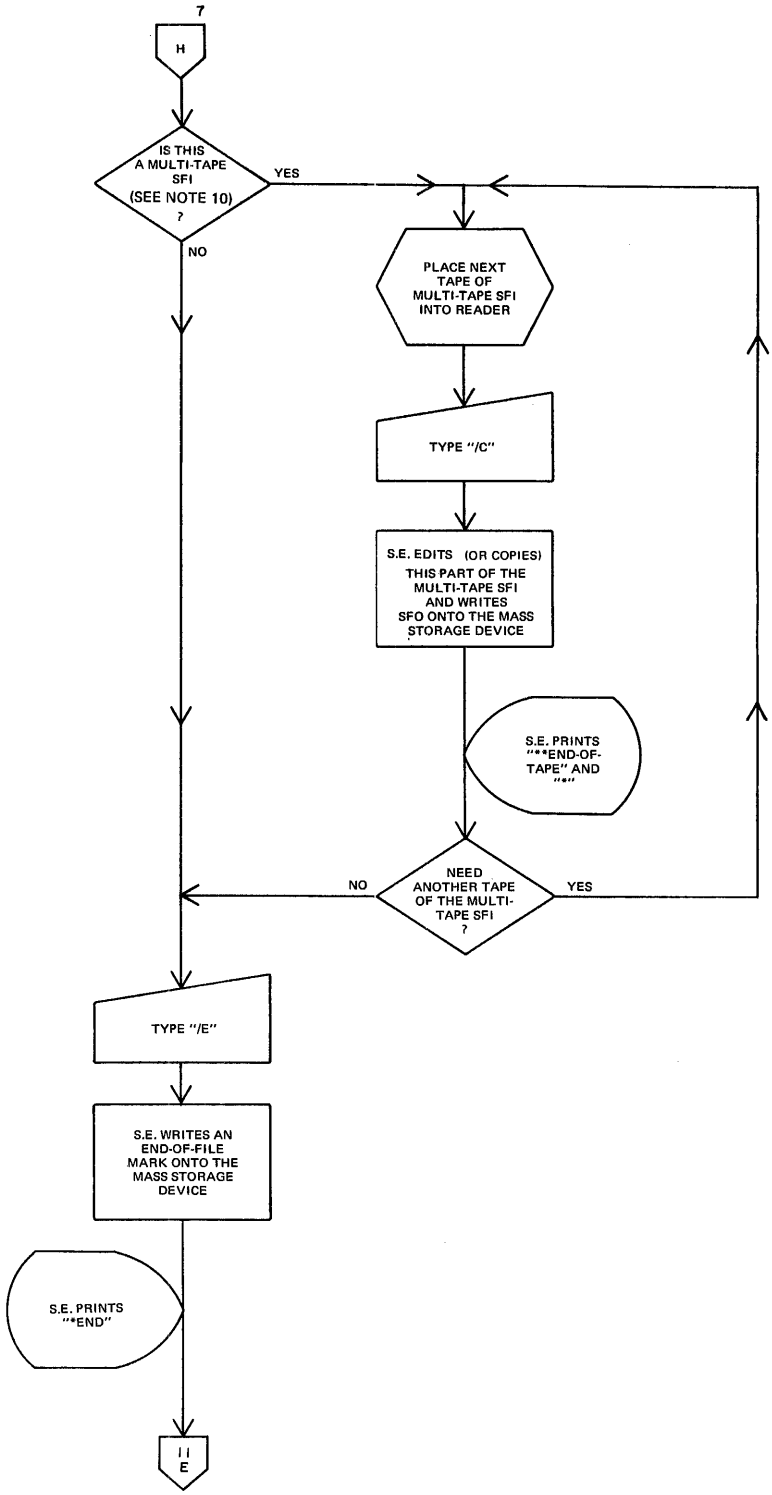
NOTE 10

The Multi-Tape Symbolic File Input allows any punched tape SFI to be read from more than one tape, if needed. See Section II of the *Symbolic Editor* manual (HP 02116-9016).


7
G
IS THIS A MULTI-TAPE SFI (SEE NOTE 10) ?
YES
PLACE NEXT TAPE OF MULTI-TAPE SFI INTO READER
NO
PLACE NEXT SFI TAPE INTO READER
TYPE "/C"
TYPE "GO"
S.E. EDITS (OR COPIES) THIS PART OF THE MULTI TAPE SFI AND WRITES SFO ONTO MASS STORAGE DEVICE
S.E. WRITES AN END-OF-FILE MARK ONTO THE MASS STORAGE DEVICE THEN EDITS (OR COPIES) NEXT SFI AND WRITES SFO ONTO MASS STORAGE DEVICE.
S.E. PRINTS "**END-OF-TAPE" AND "*"
S.E. PRINTS "** END-OF-TAPE" AND "*"
ARE ALL NEEDED SFI TAPES EDITED (SEE NOTE 8) ?
YES
NO
ARE ALL NEEDED SFI TAPES EDITED (SEE NOTE 8) ?
NO
YES
IS THIS THE FIRST PART OF A MULTI-TAPE SFI ?
YES
NO
PLACE NEXT SFI TAPE INTO READER
IS THIS ANOTHER PART OF THE CURRENT MULTI-TAPE SFI ?
NO
TYPE "GO" OR "/E"
S.E. WRITES AN END-OF-FILE MARK ONTO THE MASS STORAGE DEVICE
S.E. PRINTS "*END"
11
E

NOTE 10

The Multi-Tape Symbolic File Input allows any punched tape SFI to be read from more than one tape, if needed. See Section II of the *Symbolic Editor* manual (HP 02116-9016).

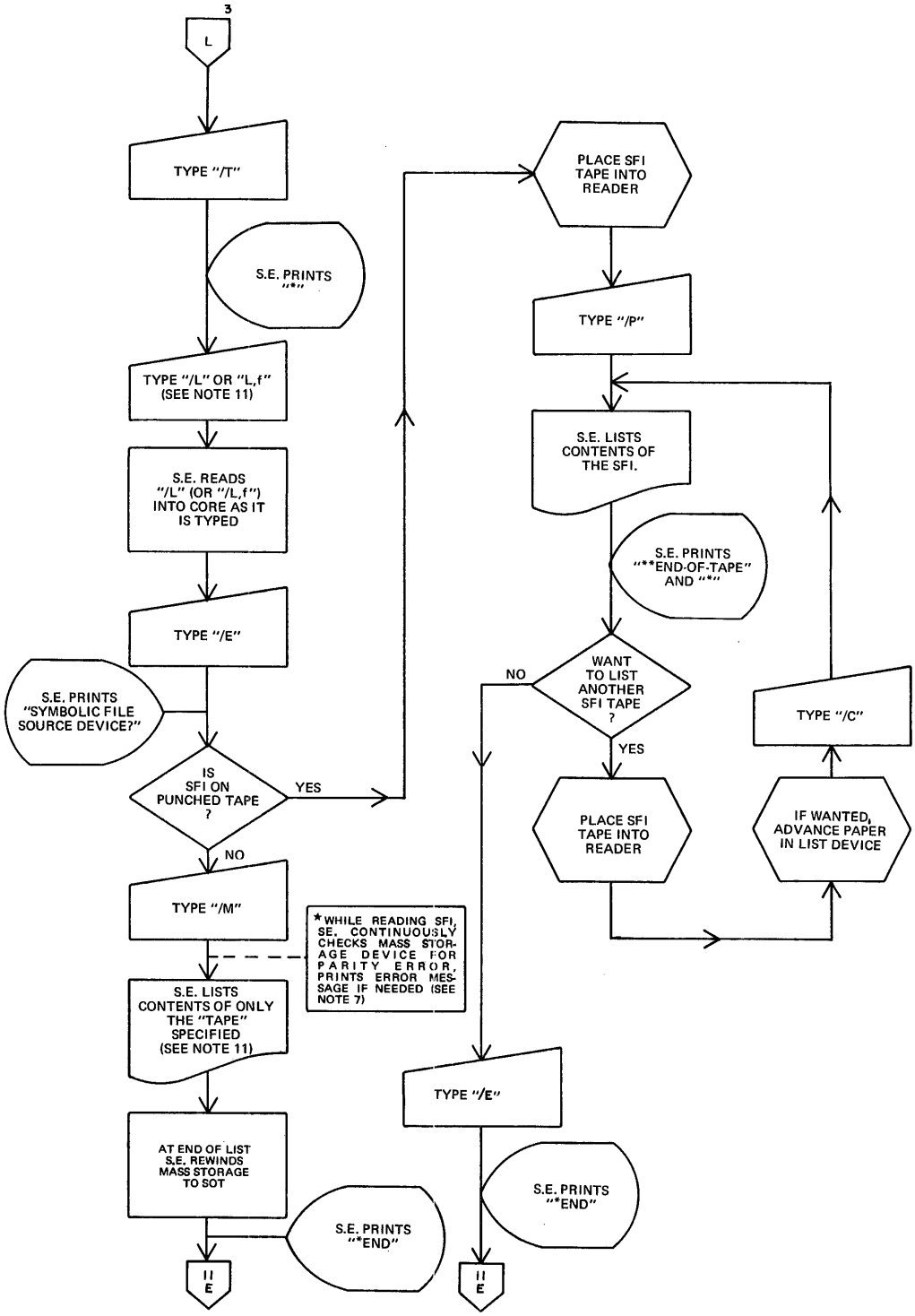

7
H
IS THIS A MULTI-TAPE SFI (SEE NOTE 10) ?
YES
NO
PLACE NEXT TAPE OF MULTI-TAPE SFI INTO READER
TYPE "/C"
S.E. EDITS (OR COPIES) THIS PART OF THE MULTI-TAPE SFI AND WRITES SFO ONTO THE MASS STORAGE DEVICE
S.E. PRINTS "**END-OF-TAPE" AND "*"
NEED ANOTHER TAPE OF THE MULTI-TAPE SFI ?
NO
YES
TYPE "/E"
S.E. WRITES AN END-OF-FILE MARK ONTO THE MASS STORAGE DEVICE
S.E. PRINTS "*END"
11
E

NOTE 7

See Section III, Error Condition, in the *Symbolic Editor* manual (HP 02116-9016).

NOTE 11

The "/L,*f*" edit instruction can be used only to list a single symbolic file input "tape" from mass storage. That "tape" is one of several that were written into mass storage either by the Symbolic Editor with or without changes, or by the Assembler. The number *f* identifies the relative position of the desired SFI in mass storage.


3
L
TYPE "/T"
S.E. PRINTS "*"
TYPE "/L" OR "L,f" (SEE NOTE 11)
S.E. READS "/L" (OR "/L,f") INTO CORE AS IT IS TYPED
TYPE "/E"
S.E. PRINTS "SYMBOLIC FILE SOURCE DEVICE?"
IS SFI ON PUNCHED TAPE ?
YES
NO
TYPE "/M"
*WHILE READING SFI, SE. CONTINUOUSLY CHECKS MASS STORAGE DEVICE FOR PARITY ERROR, PRINTS ERROR MESSAGE IF NEEDED (SEE NOTE 7)
S.E. LISTS CONTENTS OF ONLY THE "TAPE" SPECIFIED (SEE NOTE 11)
AT END OF LIST S.E. REWINDS MASS STORAGE TO SOT
S.E. PRINTS "*END"
11 E
PLACE SFI TAPE INTO READER
TYPE "/P"
S.E. LISTS CONTENTS OF THE SFI.
S.E. PRINTS "**END-OF-TAPE" AND "*"
WANT TO LIST ANOTHER SFI TAPE ?
NO
YES
PLACE SFI TAPE INTO READER
TYPE "/C"
IF WANTED, ADVANCE PAPER IN LIST DEVICE
TYPE "/E"
S.E. PRINTS "*END"
11 E

# SECTION III
# ERROR CONDITIONS

## EDIT FILE ERRORS

During each edit cycle, the Symbolic Editor has three opportunities to detect errors in the Edit File: (1) when it reads the Edit File, (2) as it attempts to perform the requested operations, and (3) after the user has signaled the end of the symbolic file input. When an error is found, the S.E. prints a message that signals the error and identifies the type.

### Deletion Of Edit File Errors

Each error causes the S.E. to either delete that Edit File line from storage in available memory or ignore that Edit File line when the operation is attempted.

### Format Of Edit File Error Messages

All error messages consist of five lines. The first line is blank, to separate the following lines from other execution control messages. The second line contains one of seven possible messages, to identify the type of error. The third line is a copy of the Edit Control line that caused the error condition. The fourth line reads "CONTROL STATEMENT DELETED!", and the fifth line contains only an asterisk "*", to signal the end of the error message sequence.

## Types of Edit File Error Messages

These are the seven types of error messages:

| Error Message | Cause(s) |
|---|---|
| **CS ERR | 1. Meaningless edit code. See Section I for all valid edit codes.<br>2. All or part of the operand is missing. See Section I for operand format requirements.<br>3. The second *c* number is smaller than or the same as the first one.<br>4. Two *ℓ* numbers have been specified in the operand of a /I edit code.<br>5. Two *c* numbers have been specified in the operand for a /CI edit code.<br>6. The entry "/P" has been used to specify the symbolic file input device when the Edit File contains a "/L,*f*" Edit Control line. |
| **CHAR OV | The number of characters to be inserted by a "/CI" or a "/CR" Edit Control line would cause the new line to be longer than $72_{10}$ characters. |
| **EDIT OVERFLOW | The Edit File is too large for storage in available memory. This is the only type of error that causes the S.E. to abort the current edit cycle. |
| **ILLEGAL VALUE | 1. An *ℓ* number is larger than 9999.<br>2. A *c* number is larger than 72. |

## ERROR CONDITIONS

| Error Message | Cause(s) |
|---|---|
| **INSERT ERROR | No Edit Data line follows this Edit Control line; i.e., the next line begins with a slash "/". See "Basic Elements" in Section I. |
| **REPLACE ERROR | No Edit Data line follows this Edit Control line; i.e., the next line begins with a slash "/". See "Basic Elements" in Section I. |
| **SEQUENCE ERROR | 1. The second $\ell$ number is smaller than or the same as the first.<br>2. The $\ell$ number is smaller than or the same as the $\ell$ number in a preceding Edit Control line.<br>3. The $\ell$ number is larger than the number of lines in the symbolic file input (detected only after the symbolic file input is complete). |

### Correcting Edit File Errors

Any Edit File that does not produce an **EDIT OVERFLOW error but does include other types of errors will be used by the Symbolic Editor (unless all Edit File lines are erroneous). After that edit cycle is ended, the user may correct for the Edit File errors by either of two methods:

(1) Correct the original Edit File and discard the symbolic file output(s) produced by that original Edit File, then perform another edit cycle. To do so, the user may repunch the entire Edit File "by hand," or he may use the Symbolic Editor to make the needed changes. If he uses the Symbolic Editor, he must use the exclamation point (!) before any first character slashes (/), as described in Section I.

(2) Save the symbolic file output(s) produced by that original Edit File, write a new Edit File to correct the first symbolic file output(s), then perform another edit cycle. If this method is used, the new Edit File must account for any new line and character numbers that resulted from

the first edit cycle. The best way to check for any new numbers is to use the Symbolic Editor's listing operation on the first symbolic file output(s).

An Edit File that produces an **EDIT OVERFLOW error is corrected only by breaking the Edit File into two or more parts, then performing an edit cycle for each of the parts.

## MASS STORAGE ERRORS

The Symbolic Editor, when used with a mass storage device, checks the status of that device prior to a read or a write operation and during a read operation.

### Mass Storage Errors Prior To Operations

If the device is a magnetic tape unit, prior to a read or a write operation, the Symbolic Editor may print "TAPE UNIT BUSY OR IN LOCAL MODE." or "WRITE ENABLE RING MISSING." and an asterisk (*) to signal that it will accept an entry on the keyboard. The user should then correct the condition, type "GO", and press the RETURN then LINE FEED keys.

### Mass Storage Error During A Read Operation

At any time during a read operation, the Symbolic Editor may detect a parity condition on the mass storage device. As a result, the Symbolic Editor prints "MT PARITY ERROR ON SOURCE FILE INPUT." and an asterisk (*) to signal that it will accept an entry on the keyboard. The user can request the Symbolic Editor to try the read operation again, by typing "GO" and pressing the RETURN then LINE FEED keys. If the condition persists, the Symbolic Editor will repeat the error message. Then the user has two choices:

(1) He can continue the edit cycle and use the last information obtained in the last reread attempt. To do so, he types "/C" and presses the RETURN then LINE FEED keys.

(2) He can terminate the edit cycle. To do so, he types "/E" and presses the RETURN then LINE FEED keys.

## End-Of-Tape (EOT) Condition Error

During an edit cycle using mass storage the computer may halt with $102001_8$ in the MEMORY DATA register. This occurs only if the mass storage device encounters an EOT before the end of the edit cycle. The user will then have to either shorten the symbolic file input(s) or enlarge the capacity of the mass storage device, then restart the Symbolic Editor.

# APPENDIX A

# EXAMPLES

Four examples of Symbolic Editor executions are presented in this Appendix to illustrate some of the Symbolic Editor's features.

Example 1 shows the basic edit cycle components, a symbolic file input to be changed, the Edit File used to make the changes, the execution control messages exchanged between the user and the Symbolic Editor, and the symbolic file output produced.

Example 2 shows the components of a Multi-tape Symbolic File Input edit cycle. In this case, a FORTRAN source program had been written to make a given computation. Then a second source program had been written to add a second computation to the first source program. The Symbolic Editor was used to combine the two source programs into one contiguous source program that includes both computations.

Example 3 shows the use of the Multi-tape and Multiple Symbolic File Inputs features combined, to make the same changes illustrated in Examples 1 and 2 in a single edit cycle.

Example 4 shows a typical Listing edit cycle in which the contents of a symbolic file input were only listed.

## Example Set 1, A Basic Edit Cycle.

```
ASMB,A,B
      ORG 115B
      LDA PRNTS
      STA COUNT
COUNT BSS 1
START LDA LNGTH
      LDB BUFF
      JSB 102B,I
      ISZ COUNT
      JMP START
      HLT 15
BUFF  DEF MSG
LNGTH OCT 14
PRNTS DEC -10
MSG   ASC 6,TEST OUTPUT.
      END COUNT
```

Example 1-1. Symbolic File Input

```
                    HP SYMBOLIC EDITOR

EDIT FILE DEVICE?
/T

*
/CD,6,13                    } The typed Edit File
/CD,13,3                    }
/E

SYMBOLIC FILE SOURCE DEVICE?
/P

SYMBOLIC FILE DESTINATION DEVICE?
/P


**END-OF-TAPE
*
/E

*END
```

Example 1-2. Typed Edit File and Execution Control Messages

```
ASMB,A,B
      ORG 115B
      LDA PRNTS
      STA COUNT
COUNT BSS 1
START LDA LNTH
      LDB BUFF
      JSB 102B,I
      ISZ COUNT
      JMP START
      HLT 15
BUFF  DEF MSG
LNTH OCT 14
PRNTS DEC -10
MSG   ASC 6,TEST OUTPUT.
      END COUNT
```

Example 1-3. Symbolic File Output

## Example Set 2, A Multi-tape Symbolic File Input Edit Cycle.

```
FTN,R,L,B
C  THIS PROGRAM FINDS THE E.T.E. FOR A
C  GIVEN VELOCITY AND DISTANCE.
      PROGRAM DTET1
      WRITE(2,200)
10    WRITE(2,300)
      READ(1,400)VEL
      IF (VEL)9999,9999,20
20    READ(1,400)DST
      ETE=DST/VEL
      ETEH=IFIX(ETE)
      ITEH=ETEH
      ITEM=IFIX((ETE-ETEH)*60.)
      WRITE(6,600)VEL,DST
      WRITE(6,700)ITEH,ITEM
      GO TO 10
9999  WRITE(6,800)
200   FORMAT("THIS PROGRAM DETERMINES E.T.E.")
300   FORMAT("TYPE, IN TWO LINES, VALUES FOR VELOCITY
     1, THEN DISTANCE, IN F4.2 FORMAT.")
400   FORMAT(F4.2)
600   FORMAT(2/"-VELOCITY =",F8.4," DISTANCE =",F8.4)
700   FORMAT(/"-ESTIMATED TIME ENROUTE =",I3," HOURS, ",I3,
     1" MINUTES."/)
      STOP
800   FORMAT(6/"-DONE"/"1")
      END
      END$
```

Example 2-1. Symbolic File Input Tape #1

```
101   WRITE(2,900)
      READ(1,1000)CANS
      IF(CANS)10,10,30
30    WRITE(2,1100)
      READ(1,400)RTE
      COST=RTE*ETE
      WRITE(6,1200)COST
      WRITE(2,1300)
      READ(1,1000)LANS
      IF (LANS)9696,9696,10
9696  WRITE(6,800)
900   FORMAT(3/"WANT COST?  YES = '1', NO = '0'.")
1000  FORMAT(I2)
1100  FORMAT(2/"TYPE RATE IN F5.2 FORMAT.")
1200  FORMAT(2/"-COST =",F8.4)
1300  FORMAT(3/"ANOTHER LEG?  1 OR 0.")
      END
      END$
```

Example 2-2. Symbolic File Input Tape #2

```
/CD,1,7,8
/CI,3,31
 AND COST, IF WANTED.
/CR,4,15,19
ETCO
/D,8
/D,16,17
/CR,20,28
5
/R,21
400   FORMAT(F5.2)
/D,25
/I,26
C  THIS IS THE SECOND PART, FOR COST.
/D,27,28
/E
```

Example 2-3. Edit File (on tape)

## Example Set 2 (cont.), A Multi-tape Symbolic File Input Edit Cycle.

```
                    HP SYMBOLIC EDITOR

EDIT FILE DEVICE?
/P

SYMBOLIC FILE SOURCE DEVICE?
/P

SYMBOLIC FILE DESTINATION DEVICE?
/P


**END-OF-TAPE
*
/C


**END-OF-TAPE
*
/E

*END
```

Example 2-4. Execution Control Messages

```
FTN,R,B
C  THIS PROGRAM FINDS THE E.T.E. FOR A
C  GIVEN VELOCITY AND DISTANCE. AND COST, IF WANTED.
       PROGRAM ETCO
       WRITE(2,200)
10     WRITE(2,300)
       READ(1,400)VEL
20     READ(1,400)DST
       ETE=DST/VEL
       ETEH=IFIX(ETE)
       ITEH=ETEH
       ITEM=IFIX((ETE-ETEH)*60.)
       WRITE(6,600)VEL,DST
       WRITE(6,700)ITEH,ITEM
200    FORMAT("THIS PROGRAM DETERMINES E.T.E.")
300    FORMAT("TYPE, IN TWO LINES, VALUES FOR VELOCITY
     1, THEN DISTANCE, IN F5.2 FORMAT.")
400    FORMAT(F5.2)
600    FORMAT(2/"-VELOCITY =",F8.4," DISTANCE =",F8.4)
700    FORMAT(/"-ESTIMATED TIME ENROUTE =",I3," HOURS, ",I3,
     1" MINUTES."/)
800    FORMAT(6/"-DONE"/"1")
C  THIS IS THE SECOND PART, FOR COST.
101    WRITE(2,900)
       READ(1,1000)CANS
       IF(CANS)10,10,30
30     WRITE(2,1100)
       READ(1,400)RTE
       COST=RTE*ETE
       WRITE(6,1200)COST
       WRITE(2,1300)
       READ(1,1000)LANS
       IF (LANS)9696,9696,10
9696   WRITE(6,800)
900    FORMAT(3/"WANT COST?  YES = '1', NO = '0'.")
1000   FORMAT(I2)
1100   FORMAT(2/"TYPE RATE IN F5.2 FORMAT.")
1200   FORMAT(2/"-COST =",F8.4)
1300   FORMAT(3/"ANOTHER LEG?  1 OR 0.")
       END
       END$
```

Example 2-5. Symbolic File Output

## Example Set 3, A Multi-tape and Multiple Symbolic File Inputs Edit Cycle.

```
FTN,R,L,B
C  THIS PROGRAM FINDS THE E.T.E. FOR A
C  GIVEN VELOCITY AND DISTANCE.
      PROGRAM DTET1
      WRITE(2,20.....
.
.
.
.
00    FORMAT(6/"-DONE"/"1")
      END
      END$
```

Example 3-1. Symbolic File Input 1 Tape 1
(same as Example 2-1)

```
101   WRITE(2,900)
      READ(1,1000)CANS
      IF(CANS)10,10,30
30    WRIT......
.
.
.
.
100   FORMAT(2/"TYPE RATE IN F5.2 FORMAT.")
1200  FORMAT(2/"-COST =",F8.4)
1300  FORMAT(3/"ANOTHER LEG?  1 OR 0.")
      END
      END$
```

Example 3-2. Symbolic File Input 1 Tape 2
(same as Example 2-2)

```
ASMB,A,B
      ORG 115B
      LDA P......
.
.
.
.
MSG   ASC 6,TEST OUTPUT.
      END COUNT
```

Example 3-3. Symbolic File Input 2
(same as Example 1-1)

## Example Set 3 (cont.), A Multi-tape and Multiple Symbolic File Inputs Edit Cycle.

```
/CD,1,7,8
/CI,3,31
 AND COST, IF WANTED.
/CR,4,15,19
ETCO
/D,8
/D,16,17
/CR,20,28
5
/R,21
400    FORMAT(F5.2)
/D,25
/I,26
C  THIS IS THE SECOND PART, FOR COST.
/D,27,28
/F,2
/CD,6,13
/CD,13,3
/E
```

Example 3-4. Edit File (on tape)

```
                    HP SYMBOLIC EDITOR

EDIT FILE DEVICE?
/P

SYMBOLIC FILE SOURCE DEVICE?
/P

SYMBOLIC FILE DESTINATION DEVICE?
/P


**END-OF-TAPE
*
/C


**END-OF-TAPE
*
GO

TEAR PUNCHED PAPER TAPE.
*
GO


**END-OF-TAPE
*
/E

*END
```

Example 3-5. Execution Control Messages

## Example Set 3 (cont.), A Multi-tape and Multiple Symbolic File Inputs Edit Cycle.

```
FTN,R,B
C  THIS PROGRAM FINDS THE E.T.E. FOR A
C  GIVEN VELOCITY AND DISTANCE. AND COST, IF WANTED.
      PROGRAM ETCO
      WRITE(2,200)
10    WRITE(2,300)
      READ(1,400)VEL
20    READ(1,400)DST
      ETE=DST/.......
.
.
.
.
.
.
.
1300  FORMAT(3/"ANOTHER LEG?  1 OR 0.")
      END
      END$
```

Example 3-6. Symbolic File Output 1 (same as Example 2-5)

```
ASMB,A,B
      ORG 115B
      LDA PRNTS
      STA COUNT
COUNT BSS 1
START LDA LNTH
      LDB BU.....
.
.
.
.
.
.
MSG   ASC 6,TEST OUTPUT.
      END COUNT
```

Example 3-7. Symbolic File Output 2 (same as Example 1-3)

## Example Set 4, A Listing Edit Cycle.

```
                        HP SYMBOLIC EDITOR

EDIT FILE DEVICE?
/T

*
/L                  } The typed Edit File
/E                  }

SYMBOLIC FILE SOURCE DEVICE?
/P

0001  /I,164
0002          STD(18,5)=133.0
0003          STD(19,5)=131.1
0004          STD(20,5)=129.3
0005          STD(21,5)=127.5
0006          STD(22,5)=125.7
0007          STD(23,5)=123.9
0008          STD(24,5)=122.1
0009          STD(25,5)=120.3
0010          STD(26,5)=118.5
0011          STD(27,5)=116.7
0012          STD(28,5)=114.9
0013          STD(29,5)=113.2
0014          STD(30,5)=111.4
0015          STD(31,5)=109.7
0016          STD(32,5)=107.9
0017          STD(33,5)=106.2
0018          STD(34,5)=104.5
0019          STD(35,5)=102.8
0020          STD(36,5)=101.1
0021          STD(37,5)=99.4
0022          STD(38,5)=97.7
0023          STD(39,5)=96.0
0024          STD(40,5)=0.0
0025  /CR,281,11,14
0026  5800
0027  /CR,282,17
0028  5
0029  /CR,288,30,32
0030  194
0031  /CR,290,17,19
0032  194
0033  /CR,296,18,20
0034  189
0035  /CR,324,20
0036  5
0037  /CI,357,37
0038  .
0039  /E


**END-OF-TAPE
*
/E

*END
```

Example 4-1. Execution Control Messages, Typed Edit File, and Listing Produced

# APPENDIX B
# CONFIGURATION PROCEDURES

Configuring the Symbolic Editor is the process of combining it with a set of SIO drivers (one for each I/O device needed) and identifying the Select Code of each device. This appendix is a summary of the detailed configuration procedure descriptions given in the *Operating Manual* (HP 02116-9057).

## CONFIGURATION PROCEDURE

Before starting this procedure, the user should determine whether the configured Symbolic Editor will ever be used to edit from or to mass storage. If it will, he must consult the "*Mass Storage Use*" subsection of this appendix before using the following steps.

1. Use the Basic Binary Loader (BBL) to load the unconfigured Symbolic Editor tape.
2. Use the BBL again to load the SIO Teleprinter Driver, then configure* that driver.
3. If a punched tape reader is available, use BBL again, to load its SIO driver, then configure that driver.
4. If a tape punch unit is available, use the BBL again to load its SIO driver, then configure that driver.
5. If a mass storage device is to be used, use the BBL again to load its SIO driver, then configure that driver.
6. Use the BBL again, to load the SIO System Dump tape.
7. LOAD ADDRESS (set to S.A.) $2_8$.
8. Set the Switch Register to $100000_8$ and turn on the tape punch unit.

---

*Configuration of SIO drivers procedures are described in the *Operating Manual*.

9. Press PRESET and RUN. The SIO System Dump program punches the configured Symbolic Editor tape. When the computer halts, if the MEMORY DATA register contains $102077_8$, the tape is complete and ready for use. If the MEMORY DATA register contains $102066_8$, the tape supply in the tape punch unit was too low; replenish the supply and repeat steps 7, 8, and 9.

   *NOTE: Each time the tape supply is replenished the chad box should be emptied.*

## MASS STORAGE USE

The Symbolic Editor can edit to or from mass storage. However, to protect a system that might already exist in that mass storage (such as the Magnetic Tape System), the Symbolic Editor is programmed to skip two EOFs (End-Of-File marks) before reading or writing data. See the *Magnetic Tape System* manual (HP 02116-91752).

If the Symbolic Editor is to be used as a free-standing program and it is to use mass storage at any time, two EOFs must be written into mass storage first. The user may use any appropriate program at his disposal or the Prepare Tape System (PTS) program available from Hewlett-Packard. To use the Prepare Tape System program, the user performs the steps given in the *Prepare Tape System* manual (HP 02116-91751) for writing only two EOFs.

# APPENDIX C
# CARD DECK PROCESSING

This Appendix explains the operating and instruction differences for processing Edit File or symbolic file inputs from cards. The main text of this manual is referenced, with the appropriate modifications.

## THE EDIT FILE

In Section I, in the subsection:

### Basic Elements

To read an Edit File from punched or marked cards, each line of the Edit File must be contained on one card. The end of a line is signaled by the end of a card. All blank characters (spaces) between the last character of the line and the end of the card are ignored.

The edit code "/↑" is punched as "/|" (numeric Y) in Hollerith code, or the user can easily delete a line from the Edit File by removing the card for that line.

## OPERATING PROCEDURES

In Section II, in the subsections:

### Loading From Punched Tape

To configure the S.E. for processing punched or marked cards, the user should use the descriptions in this appendix but substitute the words "card reader" for "punched tape reader" in step 3 in Appendix B. That configuration replaces the ability to read punched tapes by the ability to read cards.

## Execution Control Messages

To read the Edit File from cards, the user answers the S.E. query "EDIT FILE DEVICE?" by placing the Edit File card deck into the reader, readying the reader, and typing "/P". The S.E. then reads the cards because the SIO driver for the punched tape reader has been replaced by the SIO driver for the card reader.

To read the symbolic file input from cards, the user answers the S.E. query "SYMBOLIC FILE SOURCE DEVICE?" by placing the symbolic file card deck into the reader, readying the reader, and typing "/P".

## Multi-tape Edit File

This feature is not used when the Edit File is read from cards. If the Edit File is contained on more than one deck of cards, those decks should be merged into one before they are submitted to the S.E. Only the last card of an Edit File deck may contain "/E" to terminate the Edit File. If an Edit File deck is submitted without a "/E" card, the user must include one blank card to signal the end of that deck. Then, after the S.E. has printed "**END-OF-TAPE" and "*", the user terminates the Edit File by typing "/C" then "/E" on the keyboard.

## Multi-tape Symbolic File Input

This feature is not used when the symbolic file input is to be read from cards. If the symbolic file input is contained on more than one deck of cards, those decks should be merged into one before they are submitted to the S.E. The end of that merged deck is signaled by one blank card.

## Multiple Symbolic File Inputs

For inputs read from cards, the end of each input is signaled by one blank card. Then the user responds to the S.E. messages "**END-OF-TAPE" and "*" by placing the next input deck into the reader and

typing "GO". As with punched tape processing, if the outputs are on punched tape, the user must type "GO" again after the S.E. has printed "TEAR PUNCHED PAPER TAPE" and "*".

## Listing

To list a symbolic file input read from cards, the user answers the "SYMBOLIC FILE SOURCE DEVICE?" query by placing the input deck into the card reader and typing "/P". The S.E. immediately begins to read that deck and print the contents.

## Operating Procedures Flow Chart

Throughout the flow chart itself, and the other notes, substitute "card decks" for "tapes" in all references to the Edit File (EF) and the symbolic file input (SFI).

CUT ALONG LINE

HEWLETT hp PACKARD

# READER COMMENT SHEET

SYMBOLIC EDITOR

HP 02116-9016 April 1970

Hewlett-Packard welcomes your evaluation of this text. Any errors, suggested additions, deletions, or general comments may be made below. Use extra pages if you like.

FROM PAGE__OF__

NAME: _____

ADDRESS: _____

_____

NO POSTAGE NECESSARY IF MAILED IN U.S.A.

FOLD ON DOTTED LINES AS SHOWN ON OTHER SIDE AND TAPE

FOLD FOLD

FIRST CLASS
PERMIT NO. 141
CUPERTINO
CALIFORNIA

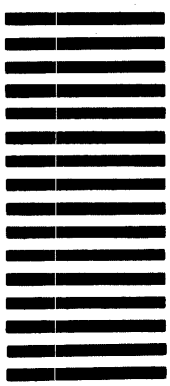
**BUSINESS REPLY MAIL**

No Postage Necessary if Mailed in the United States Postage will be paid by

SUPERVISOR, SOFTWARE PUBLICATIONS

HEWLETT - PACKARD

**CUPERTINO DIVISION**
**11000 Wolfe Road**
**Cupertino, California**
**95014**



FOLD FOLD