

Information in this document is subject to change without notice and does not represent a commitment on the part of Microsoft. The software described in this document is furnished under a license agreement or non-disclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy any part of this product on cassette tape, disk or any other medium for any purpose other than personal convenience.

(C) Microsoft, 1981

LIMITED WARRANTY

MICRO.SOFT shall have no liability or responsibility to purchaser or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by this product, including but not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of this product. This product will be exchanged within twelve months from date of purchase if defective in manufacture, labeling or packaging, but except for such replacement the sale or subsequent use of this program is without warranty or liability.

THE ABOVE IS A LIMITED WARRANTY AND THE ONLY WARRANTY MADE BY MICROSOFT. ANY AND ALL WARRANTIES FOR MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR PURPOSE ARE EXPRESSLY EXCLUDED.

TRSDOS is a **registered trademark** of Tandy Corporation

Contents

Summary of Disk EDTASM-PLUS Features	5
Terms	6
Loading Disk EDTASM-PLUS	7
Loading and Writing (Saving) Source Files	8
Exiting Disk EDTASM-PLUS Programs	11
Pseudo-Operations	12
Assembling Source Files	14
Symbol Table Switch	16
Z-BUG Commands	17
Exit	17
Load	17
Punch	17
Loading Stand-Alone Z-BUG	21
A Hint for Handling Long Programs	22
Disk Version Utility Programs	25
SYSMACRO/ASM	26
LPTSET/BAS	28
PATCHER/BAS	32

Microsoft's EDITOR/ASSEMBLER-PLUS, supplied on disk, takes advantage of the extended features and flexibility available when assembly language programs are assembled, stored in, and retrieved from disk files. Not only are storage and retrieval much faster than for cassette tape programs, but also additional files can be assembled into your main program by using the INCLUDE pseudo-operation provided in the disk version.

Disk EDTASM-PLUS continues to support cassette tape files with one exception: the INCLUDE pseudo-operation cannot be used to include tape files into your main program. INCLUDED files must be disk files. However, your main program may be a tape file which contains an INCLUDE statement. Otherwise, there are no differences between support for disk and tape files, except obvious operational differences. You can continue to use programs you may have written already using Cassette EDTASM-PLUS.

Summaries of the Disk EDTASM-PLUS features and their operation are listed following the Contents page. The features are described under headings shown on the Contents page. Following each heading in this document is a reference to the heading in the EDITOR/ASSEMBLER-PLUS manual. These references (followed by page numbers in parentheses) indicate where the changes resulting from the disk version apply to the descriptions in the Microsoft EDITOR/ASSEMBLER-PLUS manual.

IMPORTANT

Read the EDITOR/ASSEMBLER-PLUS Manual before reading the Additional Instructions for Disk EDTASM-PLUS. The Manual provides all of the necessary basic information you need; the Additional Instructions amend and supplement the Manual for the Disk Version of EDITOR/ASSEMBLER-PLUS.

Summary of Disk EDTASM-PLUS Features

1. Users can write and load editor files to and from disk.
2. Users can assemble to disk object files with symbol tables.
3. Under Z-BUG, users can load from disk object files with symbol tables.
4. Users can Punch memory images to disk under Z-BUG.
5. Users have three more pseudo-operations to use:

PAGE, which directs EDTASM-PLUS to start a new listing page

TITLE, which directs EDTASM-PLUS to print a running title on each page of the assembly listing

INCLUDE, which inserts additional source disk files into assembled main program

6. Users have three more programs to use:

PATCHER/BAS, a BASIC program for patching the EDTASM-PLUS program

LPTSET, a BASIC program to set line printer parameters (page size, line length, lines per page)

SYSMACRO, a program used with the INCLUDE pseudo-operation that contains macro definitions for TRSDOS system I/O calls

**** SHIFT ^ is SHIFT-Up Arrow**

Terms

Under TRSDOS, the user can specify the disk drive of files plus filename extensions. These capabilities do not exist for cassette files. Therefore, a distinction is made between cassette files and disk files in the command syntax descriptions throughout this document. Cassette file names are indicated by the term *filename*. Disk file names are indicated by the term *filespec*.

filename is simply the (up to) eight character filename you give a program when you save it on cassette tape.

filespec is a TRSDOS file specification consisting of (up to) four parts:

filename/extension.password:d

The *filespec* may consist of from 1 to 23 characters: up to 8 characters for *filename*; up to 3 characters for *extension*; up to 8 characters for *password*; and 1 character for drive designation. The three punctuation marks (slash, period, and colon) are counted in the 23 characters.

If you do not observe these requirements, EDTASM will return the error message:

BAD OR MISSING FILESPEC

Other *filespec* errors return the appropriate TRSDOS ERROR nn message (where nn is the error code).

Refer to the *TRSDOS & DISK BASIC Reference Manual*, pages 3-6, for a full discussion of file specifications.

Loading Disk EDTASM-PLUS

Refer to **How** to Load EDTASM-PLUS (p. 12)

To load EDTASM-PLUS from disk:

1. Insert the disk in drive 0.
2. Press the RESET button. The disk will boot and print the messages:

TRSDOS - DISK OPERATING SYSTEM - VER 2.3
DOS READY

3. Enter:

EDTASM

When loaded, EDTASM/CMD will print the messages:

MICROSOFT EDITOR/ASSEMBLER-PLUS
COPYRIGHT (C) 1981 BY MICROSOFT
DISK VERSION 1.00 — FOR TRSDOS
*

All editor, assembler, and Z-BUG commands are now available just as described in the EDITOR/ ASSEMBLER-PLUS manual.

4. To return to TRSDOS
 - a. From Editor — type K ENTER
 - b. From Z-BUG — type SHIFT ^ K (escape K)

Loading and Writing (Saving) Source Files

Refer to A Review of the Basic Editor Commands (p. 19)

The Load and Write commands have been altered to support both cassette and disk files.

Note that all Load and Write commands now must include either a C or a D with the L or W to indicate whether EDTASM-PLUS will load or write to a *Cassette* file or a *Disk* file. There is no default if the C or the D is omitted. If you fail to enter either the C or the D, EDTASM returns a BAD COMMAND error message.

Command	Format	Description
Load	LC filename	Load the named file from cassette. The filename must be a legal EDTASM-PLUS cassette file name.
	LCA filename	Append the named cassette file to the file(s) already in the edit buffer. Normally, TRSDOS empties the buffer when a new Load command is given. This additional command (not available in the cassette version) allows you to append the file without emptying the buffer. The appended file is always loaded at the end of the file(s) already in the edit buffer.

NOTE

If the first line number of the file being appended is less than or equal to the highest line number in the edit buffer, you must enter the N command to renumber the lines before further editing can take place. If the first line number of the appended file is greater than the highest line number in the edit buffer, renumbering is not necessary.

- LD filespec Load the specified file from disk. The filespec must be a legal TRSDOS disk file specification. See the summary description under the heading Terms on page 6, above, and refer to TRSDOS manual.
- LDA filespec Append the named file to the file(s) already in the edit buffer. Normally, TRSDOS empties the buffer when a new Load command is given. This additional command (not available in the cassette version) allows you to append the file without emptying the buffer. The same rules governing file specifications under LD apply to LDA. The appended file is always loaded at the end of the file(s) in the edit buffer.

NOTE

If the first line number of the file being appended is less than or equal to the highest line number in the edit buffer, you must enter the N command to renumber the lines before further editing can take place. If the first line number of the appended file is greater than the highest line number in the edit buffer, renumbering is not necessary.

Write	WC filename	Write the named file to cassette. The rest of this procedure follows the description in the EDITOR/ASSEMBLER-PLUS manual, page 19.
	WD filespec	Write the specified file to disk. The filespec must be a legal TRSDOS disk file specification. See the summary description under the heading Terms on page 6 above, and refer to TRSDOS manual. If, as part of the 23 characters allowed, you do not specify a filename extension, EDTASM-PLUS appends the default extension /ASM for Editor files.

Exiting Disk EDTASM-PLUS Programs

Refer to **The Editor Commands** (pp. 22-23)

Once you are finished with editing, assembling, or debugging, you will want to exit EDTASM or Z-BUG. Under the cassette tape version of EDTASM-PLUS, you would return to Radio Shack Level II BASIC. Under Disk EDTASM-PLUS, you want to exit to TRSDOS. Two commands allow you to exit to TRSDOS. One command exits from the Editor; the other command exits from Z-BUG.

Program	Exit Command
----------------	---------------------

EDTASM	K ENTER
--------	---------

Z-BUG	SHIFT ^ (escape-K)
-------	--------------------

TRSDOS will boot and display the message:

DOS READY

Pseudo-Operations

Refer to **A Review of Pseudo-Operations** (pp. 45-48)

Three additional pseudo-operations are supported by Disk EDTASM-PLUS.

Pseudo-Operation	Description
PAGE	Start a new output page. PAGE allows you to specify where page breaks are to occur in the listing. This pseudo-op is effective only if the listing is sent to a line printer.
TITLE string	Print a running title as part of the header for each page of an assembly listing. The string is used as the text of the title and may be a maximum of 32 characters. This pseudo-op is effective only if the listing is sent to a line printer.
INCLUDE filespec	<p>Include the specified disk file in the source code file at the position of the INCLUDE statement. The source code file may be either a disk file or a cassette file, but filespec must specify a disk file only. The filespec must conform to the TRSDOS rules governing disk file specifications, but may not be more than 23 characters. (Refer to additional comments about file specifications under the LD command, above.) The file specified must contain valid source statements.</p> <p>The INCLUDE pseudo-op provides a means of handling long programs more easily. See below, A Hint for Handling Long Programs.</p>

The INCLUDE pseudo-op is also used in conjunction with the SYSMACRO/ASM program included on your Disk EDTASM-PLUS diskette. See below, **Disk Version Utility Programs.**

Assembling Source Files

Refer to **Assembler Switches** (pp. 49-50)

The basic command to start assembly has been altered to **reflect** support for both cassette and disk operations.

A switch has been added that allows you to save the symbol **table** with the object code on disk.

Command	Description
AC filename/sw	Assemble the named file to cassette. The /sw represents switches requested by the user. All switches are optional. More than one switch may be given, but each switch must be preceded by a slash mark (/). For example: AC filename/IM/WE AC may be entered without a filename. Then EDTASM-PLUS defaults to NONAME, as described in the manual. You cannot assemble to cassette unless A and C are entered together. Without the C, A alone defaults to AD.

AD filespec /sw Assemble the specified file to disk. The /sw represents switches requested by the user. The first switch given must be separated from the filespec by a single blank space. All switches are optional. More than one switch may be given, but each switch must be preceded by a slash mark (/). For example.

AD filename /IM/WE

If the A command is given without the D, EDTASM assembles the specified file to disk unless either the /IM or the /NO switch is given.

If the filespec is omitted, EDTASM returns a MISSING OR BAD FILESPEC error message unless either the /IM or the /NO switch is given.

When giving the AD command, you must give a file specification of 1 to 23 characters.

If, as part of the 23 characters allowed, you do not specify a filename extension, EDTASM-PLUS appends the default extension /CMD for Assembler files.

Symbol Table Switch

Refer to **Assembler Switches** (pp. 49-50)

Switch	Description
/WS	When the assembly is sent to disk (either an AD filespec or an A filespec command was given), entering the / WS switch will append the symbol table to the object code disk file.

Note, however, that to load an object file with symbol table appended, you will need to use the Z-BUG Load command designed for these files (filename *S addr SHIFT ^ LD*, where *addr* is the start address of the symbol table). See the next section below, Z-BUG Commands, for a full description.

Z-BUG Commands

Exit Command

Refer to **Basic Z-BUG OPERATION** (p.86)

After you finish a debugging session, you may want to return to TRSDOS command level rather than to the Editor. To return to TRSDOS, enter:

SHIFT ^ K

Your TRS-80 will return to TRSDOS and display the message:

DOS READY

Load and Punch Commands

Refer to Loading and Saving System Format Tapes (pp. 106-107)

The Z-BUG Punch and Load commands have been altered to support both cassette tape and disk files.

NOTE

The Z-BUG Punch and Load commands use the same file format as the TRSDOS Dump and Load commands. (The exception is when a symbol table has been appended to the disk object file with the new /WS switch described in the previous section of this document.) Therefore, files created by the TRSDOS Dump command can be loaded by the Z-BUG Load command, and files created by the Z-BUG Punch command can be loaded by the TRSDOS Load command.

In the following descriptions, the dollar sign (\$) is used to indicate the escape characters shift up-arrow (SHIFT ^).

Load Commands	Description
filename\$LC	Load the named file from cassette. The C must be included. If not, Z-BUG will return a BAD COMMAND error message.
filespec\$LD	<p>Load the specified file from disk. The D must be included. If not, Z-BUG will return a BAD COMMAND error message.</p> <p>The filespec may contain characters that are also Z-BUG command characters, such as slash mark (/), colon (:), and period (.). This is permitted. However, the filespec must be enclosed in double quotes (") if any of these Z-BUG command characters appear in the filespec. For example:</p> <p style="padding-left: 40px;">"filename/ext.password:d"</p>
filespec Saddr\$LD	<p>Load the disk from the specified file with symbol table appended. <i>addr</i> is the start address of the symbol table. If <i>addr</i> is omitted, Z-BUG defaults to the last specified symbol table start address. If no start address was specified since start up, Z-BUG returns a NO SYMBOL TABLE LOAD ADDRESS error message.</p> <p>When the file and symbol table are loaded, Z-BUG displays the first and last addresses of the space in memory where the symbol table is stored.</p>

CAUTION

Be sure that the symbol table does not write over your program, EDTASM-PLUS, or TRSDOS. TRSDOS occupies addresses 4200H-5200H. Z-BUG requires approximately 7.5K bytes. You should know the size of your program. Add 5200H + 7.5K + program size to calculate the minimum address for loading the symbol table safely. (Approximately 6F4C + program size: 5200H+ ID56H = 6F4CH)

Use this version of the Punch command to load very large files which you assembled to disk because of space restrictions. See the discussion below under **A Hint for Handling Long Programs**.

Punch Commands

Description

first last start filename\$PC Punch the named file to cassette. The C must be included. If not, Z-BUG will return a BAD COMMAND error message.

first is the first address of the program.

last is the last address of the program.

start is the address where you want program execution to begin.

If a filename extension is not included in the filename, Z-BUG will append the default extension /CMD.

first last start filespec\$PD Punch the specified file to disk. The D must be included. If not, Z-BUG will return a BAD COMMAND error message.

first is the first address of the program.

last is the last address of the program.

start is the address where you want program execution to begin.

If a filename extension is not included in the filespec, Z-BUG will append the default extension /CMD.

The filespec may contain characters that are also Z-BUG command characters, such as slash mark (/), colon (:), and period(.). This is permitted. However, the filespec must be enclosed in double quotes (") if any of these Z-BUG command characters appear in the filespec. For example:

"filename/ext. password:d"

Loading Stand-Alone Z-BUG

Refer to **Loading Stand-Alone Z-BUG** (p. 107)

When you are finished with the debugging session **using** Stand-Alone Z-BUG, return to TRSDOS by entering:

SHIFT ^ K

TRSDOS will boot **and** display the message:

DOS READY

—

A Hint for Handling Long Programs

Refer to **Using the Quash Command to Obtain More Memory for Edits**, pp. 37-38.

Disk EDTASM-PLUS provides easier handling of long programs. The Quash command was provided under the cassette version to obtain more memory. Besides the Quash command, several features of the Disk version provide additional assistance. These features are the Write to Disk command, the INCLUDE pseudo-op, the Assemble to Disk command, the symbol table assembly switch, and the Stand-Alone Z-BUG Load Disk file command.

Here is one procedure we can recommend for handling long programs.

1. Quash the Assembler and Z-BUG. Enter:

QA

As described in the manual, this allows the maximum amount of memory for the Edit buffer.

2. Enter your source program.
3. Save your source program with the WD command. Enter:

WD filespec

4. Create a second source program consisting of the INCLUDE pseudo-op and an END statement. For example:

```
0100    INCLUDE    filename
0200    END
```

We recommend that any file to be called by an INCLUDE statement *not* contain an END statement. Rather, place the END statement, as above, in the file which contains the INCLUDE statement. If you INCLUDE a file that has an END statement, the program will END when the statement is encountered. This causes a problem if you want to INCLUDE more than one file.

5. Assemble this INCLUDE program with the AD assembly command and the /WS switch. Enter:

```
AD filespec /WS
```

(The space between the last character of the filespec and the slash mark (/) is required.)

This will send the assembled object code file to disk with the symbol table appended. This assembly method requires only enough working memory to assemble the current line plus the symbol table. Object code is written to disk as soon as a line is assembled.

6. Load the assembled file for debugging using the Stand-Alone Z-BUG load command. From TRSDOS command level, enter:

ZBUG

Z-BUG will load and display the messages:

```
MICROSOFT EDITOR/ASSEMBLER-PLUS  
COPYRIGHT (C) 1981 BY MICROSOFT  
> > > STAND ALONE Z-BUG < < <  
DISK VERSION 1.00 — FOR TRSDOS  
Z-BUG
```

#_

Then enter:

filespec S *addr* SHIFT ^ LD

(where *addr* is the start address of the symbol table. If *addr* is omitted, Z-BUG defaults to the last specified symbol table start address. If no *addr* was specified since start up, Z-BUG returns a NO SYMBOL TABLE LOAD ADDRESS error message. The easiest and most sure method is to specify *addr* whenever you enter the Z-BUG Load command.)

Your large program will be loaded and ready to debug.

Disk Version Utility Programs

Disk EDTASM-PLUS provides three additional programs not related to any topics covered in the EDITOR/ASSEMBLER-PLUS Manual. All three program files are supported utilities; that is, they do not run on their own. Rather, each utility is called from another program, just as Z-BUG can be called from the EDTASM program.

The three utilities are:

1. SYSMACRO/ASM - a file that contains macro definitions for TRSDOS system I/O calls.
2. LPTSET/BAS - a BASIC program used to change line printer parameters.
3. PATCHER/BAS - a BASIC program used to install patches (corrections) in EDTASM-PLUS.

SYSMACRO/ASM — TRSDOS System Macros

SYSMACRO/ASM is an assembly language file (just like the ones you write using the Editor portion of EDTASM-PLUS). The file contains macro definitions for 14 TRSDOS system I/O calls. These system calls are described in Section 6, TRSDOS Technical Information, of the *TRSDOS & Disk BASIC Reference Manual*.

The macro definitions are called from a source program, using standard macro calls, as described in Chapter 4 of the EDTASM-PLUS manual. However, before you can call any of the 14 macro definitions from SYSMACRO/ASM, you must **INCLUDE** the SYSMACRO/ASM file into your source program. To do this, enter:

INCLUDE SYSMACRO

This **INCLUDE SYSMACRO** statement must appear in your program *before* any calls are made to the macro definitions.

The following list gives the macro call name and the parameters the user can specify for each I/O function when a particular macro call is given. The parameters are indicated by these abbreviations:

- BUFADR The Address of the Buffer. For the TIME and DATE macros, BUFADR defines an 8 byte buffer where TRSDOS returns the time or date. For the INIT and OPEN macros, BUFADR defines the Disk buffer.
- DCBADR The Address of the DCB.
- ERRADR The Address of the Error Routine.
- LRBADR The Logical Record Buffer Address.
- LRL The Logical Record Length.
- RECNUM The Record Number.

The macro values will be displayed in the symbol table in the order listed below.

I/O FUNCTION	MACRO NAME	PARAMETERS
Close	CLOSE	DCBADR,ERRADR
Return Date	DATE	BUFADR
initialize File	INIT	DCBADR,ERRADR,BUFADR,LRL
Kill File	KILL	DCBADR.ERRADR
Open File	OPEN	DCBADR,ERRADR,BUFADR.LRL
Position File	POSN	DCBADR,ERRADR,RECNUM
Read Logical Record	READLR	DCBADR,ERRADR
Read Physical Record	READPR	DCBADR.ERRADR
Return Time	TIME	BUFADR
Return to TRSDOS	TRSDOS	(none)
Verify Logical Record	VERFLR	DCBADR,ERRADR,LRBADR
Verify Physical Record	VERFPR	DCBADR,ERRADR
Write Logical Record	WRITLR	DCBADR,ERRADR,LRBADR
Write Physical Record	WRITPR	DCBADR,ERRADR

LPTSET/BAS

LPTSET/BAS is a BASIC program used to change the line printer parameters.

If you are using 8.5 inch by 11 inch paper in your printer, the parameters are already set appropriately. The default parameters are:

Parameter	Default Value
Page Length (of paper)	66 lines (at 6 lines per inch)
Page Width (of print)	80 columns (at 10 columns per inch)
Lines printed per page	58 lines

It is the user's responsibility to enter reasonable parameters. As each parameter must fit within one byte of memory, the range of possible values would be 1-255. The physical dimensions of the paper used, the size constraint imposed by the line printer itself, and the desired appearance of the listing determine the reasonable ranges for the parameters.

To Run LPTSET

1. From TRSDOS command level, enter:

BASIC

2. When Disk BASIC returns the prompt HOW MANY FILES?, press ENTER. When the prompt MEM SIZE? appears, press ENTER. Disk BASIC will display the messages:

```
RADIO SHACK DISK BASIC VERSION 2.2  
READY  
>
```

3. Enter:

RUN "LPTSET/BAS"

LPTSET/BAS will load and display the messages:

EDTASM+ LINE PRINTER CONFIGURATION
PROGRAM — VERSION 1.0

HAVE YOU MADE A BACKUP OF EDTASM/CMD (Y
OR N)?

You should always make a backup of your diskettes.

Response	Result
----------	--------

N	LPTSET/BAS will return the message: PLEASE MAKE A BACKUP OF EDTASM/ CMD BEFORE RUNNING THIS PRO- GRAM READY >
---	--

You are now at Disk BASIC command level.
You should enter CMD"S" to return to
TRSDOS. Then enter BACKUP, respond to
the prompts, and, when the BACKUP is
finished, begin LPTSET/BAS again at step 1,
above.

Y	LPTSET/BAS will return the following series of prompts. As you enter an answer for each, the next prompt will appear automatically.
---	---

WHICH DRIVE IS EDTASM/CMD ON?

Usually, you will have the diskette in drive 0, although this is not necessary. Simply enter the number of the drive, without colon.

PAGE LENGTH (IN LINES)?

The length is the size of the paper you are using. To convert inches to lines, multiply the length in inches by 6 (6 lines per inch).

HOW MANY LINES TO PRINT PER PAGE?

Enter the number of lines you want printed before a page break is inserted.

PAGE WIDTH (IN COLUMNS)?

The width is the number of columns to be printed, not the width of the paper. Calculate the number of columns at 10 columns per inch.

DOES YOUR PRINTER AUTOMATICALLY START A NEW LINE WHEN THE PRINT HEAD REACHES THE RIGHT MARGIN (Y OR N)?

Response	Result
Y	No change will be made to EOT ASM-PLUS.
N	Enter N only if your line printer does not fold automatically when the print head reaches the right margin. If you enter N, EDTASM-PLUS will insert a carriage return when the print head reaches the right margin (the last column you specified for the PAGE WIDTH parameter). Therefore, if you enter N but your printer does fold automatically, your files will print out double spaced all lines that are as long as or longer than the column parameter you set.

When you have entered Y or N, LPTSET/ BAS will write the new parameters to EDTASM-PLUS then exit to Disk BASIC, giving the message:

READY

>

4. Enter:

CMD"S"

to return to TRSDOS command level.

Whenever you enter the Editor commands H or T, or the Assembler switch /LP, EDTASM-PLUS will use the parameters you just set.

PATCHER/BAS

PATCHER/ BAS is a BASIC program used to install authorized patches (corrections) in EDTASM/CMD or ZBUG/CMD (Stand-Alone Z-BUG). PATCHER/BAS should be used only with patches supplied to you by Microsoft. Otherwise, you will not need this program. The following description is provided in case Microsoft does issue patches in the future.

NOTE

PATCHER/BAS does not create any files. The program only installs patches. Because of the possibility of damaging the EDTASM-PLUS files, you should first make a backup copy of the program to be patched. We recommend that you give the backup copy a filename extension which is the version number of the program. For example:

EDTASM/1.00

Procedures

1. From TRSDOS command level, load Disk EDTASM-PLUS by entering:

EDTASM

2. Enter the patch data supplied by Microsoft.

3. Save the data in a disk file. Enter:

WD EDTASM/PAT (for EDTASM patches)

or

WD ZBUG/PAT (for Z-BUG patches)

4. Enter:

K

5. Enter:

BASIC

6. Enter:

RUN "PATCHER/BAS"

PATCHER/BAS will load and display the messages:

EDTASM+ PATCH INSTALLATION PROGRAM —
VERSION 1.0

WHICH FILE TO PATCH (EDTASM OR ZBUG)?

7. Enter either:

EDTASM (for patches to EDTASM/CMD)

or

ZBUG (for patches to ZBUG/CMD)

PATCHER/BAS will display either of the messages:

HAVE YOU MADE A BACKUP OF EDTASM/CMD(Y
OR N)?

or

HAVEYOU MADE A BACKUP OFZBUG/CMD(Y OR
N)?

Enter the appropriate response:

Response	Result
N	<p>PATCHER/BAS will display either of the messages:</p> <p>PLEASE MAKE A BACKUP OF EDTASM/ CMD BEFORE RUNNING THIS PRO- GRAM</p> <p>or</p> <p>PLEASE MAKE A BACKUP OF ZBUG/ CMD BEFORE RUNNING THIS PRO- GRAM</p> <p>Then PATCHER/BAS will exit to Disk BASIC and display the message:</p> <p>READY</p> <p>></p> <p>Make a backup copy of your EDTASM- PLUS diskette, then begin PATCHER/BAS at step 1, above.</p>
Y	<p>PATCHER/BAS will display either of the messages:</p> <p>HAVE YOU PREPARED THE EDTASM/ PAT PATCH FILE (Y OR N)?</p> <p>or</p> <p>HAVE YOU PREPARED THE ZBUG/PAT PATCH FILE (Y OR N)?</p> <p>The /PAT Patch File is the file you prepared during steps 1-3, above.</p>

9. Enter the appropriate response:

Response	Result
----------	--------

N	PATCHER/ BAS will display the messages: YOU MUST CREATE THE PATCH FILE BEFORE RUNNING THIS PROGRAM. PLEASE CONSULT THE PATCHING IN- STRUCTIONS IN THE EDTASM+ DOCU- MENTATION.
---	---

READY

>

You are now at Disk BASIC command level.
Prepare the Patch File according to the instruc-
tions supplied by Microsoft for the patch.
When the patch file is created, begin PATCHER/
BAS at step 1, above.

Y	PATCHER/BAS will display either of the messages:
---	---

WHICH DRIVE IS EDTASM/CMD ON?

or

WHICH DRIVE IS ZBUG/CMD ON?

10. Enter the appropriate drive number (without colon).
PATCHER/BAS will display either of the messages:

WHICH DRIVE IS EDTASM/PAT ON?

or

WHICH DRIVE IS ZBUG/PAT ON?

11. Enter the appropriate drive number (without colon). Both
the / CMD and the / PAT files may be on the same diskette.
PATCHER/ BAS will now attempt to patch the / CMD file.

If unsuccessful, PATCHER/BAS will display an appropriate error message, followed by the message:

PATCH PROCEDURE TERMINATED, NO PATCHES INSTALLED

READY

>

You are now at Disk BASIC command level. Enter CMD"S" to return to TRSDOS. Enter EDTASM. When EDTASM is ready, enter either:

LD EDTASM/PAT:d

or

LD ZBUG/PAT:d

where :d is the number of the drive with the disk that contains the PAT file.

When the file is loaded, examine the Patch File for accuracy, make any corrections necessary, then run PATCHER/BAS again, beginning at step 1, above.

If successful, PATCHER/BAS will display the message:

ALL PATCHES SUCCESSFULLY INSTALLED

Then, PATCHER/BAS will exit to Disk BASIC which displays the prompt:

READY

>

