===

     FILESEARCH is a utility program, consisting of two
independent modules TAPESEARCH and DISKSEARCH, which
provides useful information about programs stored on tape or
disk. Although the data appears on the screen, selection of
the printer option enables a printout to be obtained.

The initial letter identifies the type of program:

   T: Basic Program
   B: Machine Code Program
   D: Data Files
   W: Editor Assembler Source Code

     TAPESEARCH reads the leader of programs and files stored
on tape and prints the information in this form:

   T: NAME OF PROGRAM 7AE9 8EA4

     Following the program type and name, the start and end
addresses of the program will be printed, in hex. Only Data
Files do not have a start and end address.

     TAPESEARCH will also identify Data Files saved using the
D.S.E. Word Processor, which cannot be identified during
normal tape operations. When the printer option is selected
only Data Files which have a name are output to the printer
although all files found are printed on the screen.

     DISKSEARCH reads the disk directory, and prints the data
in this form:

   T:FILENAME ØA Ø4 7AE9 8EA4

     Following the filetype and filename the track and sector
numbers, indicating where the data is stored on the disk,
are displayed in hex form. Immediately after this the start
and end addresses of the program are printed, also in hex.

     If the directory contains more than 13 entries, as the
data is being printed, the screen will scroll. Pressing the
SPACE bar will hold the displayed data. If the printer.
option has been chosen, you can select, each time a
directory is displayed, whether to output the data to the
printer or not.

     If a disk error occurs during reading, control will
return to DISKSEARCH. Should the program not return from
reading a disk, such as if an attempt is made to read an
unformatted disk, control can be regained by pressing the
BREAK key alone.

     FILESEARCH can be safely used by non-disk owners as a
check is made for the presence of the Disk Operating System
before entering DISKSEARCH. Pressing the CONTROL and BREAK
keys together, at any time, will enable a return to the
start of the program.

===

one disk Drv   *two disk Drv*

↓   ✓

CD 1 & ( )

```
###############################################################
#                C O P Y P R O T E C T                       #
############################    ###############################
# 1. B R E A K P R O O F #    # Copyright (C) L. Taylor 1986 #
########################## #    ###############################
# 2. F I L E C O P I E R #    #  Distribution by VSOFTWAREZ  #
###############################################################
```

This program locates itself according to available memory and is actually two programs in one. By following a simple, menu driven format, programs can be loaded and saved with ease.

BREAKPROOF enables error free Basic programs to be loaded from tape or disk, merged with a machine code routine and then resaved to either tape or disk. Once a Basic program has been breakproofed it will, on loading autorun. If the BREAK key is hit during program execution, the normal BREAK message will not appear, instead the program will automatically restart.

FILECOPIER allows the transfer of Basic or Machine Code programs, that is, T or D type files, from tape or disk in any of the following combinations or formats.

* Tape to Tape   * Tape to Disk
* Disk to Tape   * Disk to Disk

The utility makes it possible to produce single or multiple disk or tape backups of all programs quickly and easily.

```
###############################################################
```

with Breakproof

Do not inverse & one right of utility is File to

Breakproke

space to move
Ret Select

FASTDISK

This program allows a 25% improvement in disk speed. It
achieves this by initialising a disk in a different way (see
below), and this means the extra speed is possible without
first loading a special program, and is available from any
software, not just Basic.

When this program is loaded, it will display a warning message,
then wait for you to insert the disk to be initialised, and to
hit <RETURN>. As with INIT, the program will take about one and
a half minutes, and will destroy the previous content of the
disk (if any). When it has finished, the "Insert disk, hit
<RETURN>" message will reappear. If you want, another disk can
be initialised, but otherwise typing <BREAK> will return you to
Basic.

HOW IT WORKS

On each track of a disk, DOS's INIT command will arrange the
sectors in this order:

0, 11, 6, 1, 12, 7, 2, 13, 8, 3, 14, 9, 4, 15, 10, 5

There is a gap of two sectors between any two (numerically)
consecutive sectors. Some interleaving is necessary, to give
time for each sector to be processed, but two sectors is much
longer than necessary. FASTDISK initialises the disk with only
one extra sector between two consecutive ones, using this
arrangement:

0, 8, 1, 9, 2, 10, 3, 11, 4, 12, 5, 13, 6, 14, 7, 15.

The result is that DOS spends much less time waiting for each
sector to come up.

EXT16 PRELIMENARY INSTRUCTIONS :-

COMMANDS FORMAT   LABEL   VOL   REL   CHA   EXA   DIS   DISX   BREAK ON   BREAK OFF

FORMAT - SYNTAX - FORMAT - ENTER DISK LABEL - ENTER TODAY'S DATE

ENTER DISK LABEL - EG :- UTILITY DISK # 5

ENTER TODAY'S DATE - SYNTAX - XX/XX/XX OR XX/XX/XXXX

AFTER DATE ENTRY DISK WILL BE FORMATTED AND A FILE CALLED
S:DISKINFO PLACED ON DISK WHICH CONTAINS ENTERED INFORMATION.

LABEL - AS ABOVE BUT NO DISK FORMAT

IF DISK WAS PREVIOUSLY LABELLED THEN THE MESSAGE LABEL
ALREADY EXISTS WILL BE DISPLAYED.

VOL - DISPLAYS CONTENTS OF DISKINFO FILE

IF NO DISKINFO FILE ON DISK THEN THIS DISK HAS NO LABEL
MESSAGE IS DISPLAYED.

```
REL - SYNTAX  - REL"FILENAME",XXXXH
ORIGINAL      - BASICROM,0000,3FFF
EXAMPLE       - REL"BASICROM",8000
RESULT        - BASICROM,8000,EFFF
```

RELOCATED PROGRAMS WILL NOT RUN, BUT FILES LIKE ABOVE CAN NOW
BE COPIED TO ANOTHER DISK USING DCOPY"FILENAME".

```
CHA - SYNTAX - CHA"FILENAME",X
ORIGINAL - B:EXT16
EXAMPLE  - CHA"EXT16",P
RESULT   - P:EXT16
```

THE COMMAND 'CHA' ALLOWS YOU TO CHANGE THE FILETYPE ON DISK
ON ANY FILE, BUT PROGRAM MAY NOT LOAD OR RUN ANYMORE.

EXA - SYNTAX - EXA,XXXX,XXXX - EXA,TRACK NUMBER, SECTOR NUMBER

EXA ALLOWS YOU TO EXAMINE CHOSEN TRACKS & SECTORS ON DISK.
PRESS SPACE FOR NEXT SECTOR. USE CTRL + BREAK TO RETURN TO
BASIC.

DIS & DISX - SYNTAX - DIS - DISX (X=FILETYPE)

```
DIS - B:EXT16      01 02 D33B DB1A   - PRESS SPACE TO CONTINUE
      S:DISKINFO   02 02 7220 72A0
      W:EXT INST   02 08 CA49 D000      LIKE   DIR/P on MSDOS
      T:DIR#WIDE   02 04 7AE9 7D0A
```

```
DISB - DISPLAYS ONLY 'B' FILETYPE FILENAMES
DISS - DISPLAYS ONLY 'S' FILETYPE FILENAMES
DISW - DISPLAYS ONLY 'W' FILETYPE FILENAMES
DIST - DISPLAYS ONLY 'T' FILETYPE FILENAMES
DISX ETC
```

BREAK ON & BREAK OFF - I'M NOT QUITE SURE WHAT THEY DO EXCEPT
THEY'RE USED IN PROGRAMS, I THINK.

INTRODUCTION TO DISK EDITOR ASSEMBLER :-

THIS MANUAL IS DESIGNED AS AN INSTRUCTION BOOKLET FOR THE DISK EDITOR ASSEMBLER; IT IS NOT DESIGNED TO BE A TUTORIAL IN Z80 ASSEMBLY LANGUAGE PROGRAMMING. IN ORDER TO USE THE EDITOR, THE USER WILL HAVE TO KNOW THE STANDARD FORMAT FOR A Z80 ASSEMBLY LANGUAGE PROGRAM. THE USER MUST ALSO APPRECIATE THE MEANING OF 'SOURCE CODE' AND 'OBJECT CODE'. THIS PACKAGE IS DESIGNED FOR VZ200/300 (VZ200 WITH 16K RAM) WITH DISK DRIVE SYSTEM.

THE PACKAGE OFFERS SEVERAL FEATURES NOT FOUND ON OTHER ASSEMBLERS. ALL LOADING AND SAVING IS TO AND FROM DISK. SOURCE CODE IS SAVED WITH A 'S' PREFIX, OBJECT CODE WITH A 'B' PREFIX IN ORDER THAT IT CAN BE BRUN. AN EXIT TO BASIC IS OFFERED, AND IN MOST CIRCUMSTANCES, THE USER CAN RETURN TO THE EDITOR. THE PACKAGE AUTOMATICALLY MAKES USE OF WHATEVER MEMORY IS AVAILABLE.

GETTING STARTED ...

TO USE THE DISK EDITOR ASSEMBLER, TYPE BRUN"ASSEM". YOU WILL BE PRESENTED WITH THE TITLE SCREEN. WHILE THE EDITOR ASSEMBLER IS RUNNING, THE BOTTOM LINE OF THE SCREEN IS ALWAYS THE COMMAND LINE, WHETHER IN COMMAND MODE OR EDIT MODE.

NOTICE THE WORD COMMAND - THIS INDICATES THAT YOU ARE IN THE COMMAND MODE. AT ANY TIME, YOU RETURN TO THIS MODE BY PRESSING CTRL BREAK.

THERE ARE FIVE BASIC COMMANDS FOR EDITING, ENTERING AND DISPLAYING TEXT; INSERT, EDIT, DELETE, LIST AND FIND. ONLY THE FIRST LETTER OF EACH IS USED. A FULL DESCRIPTION OF EACH FOLLOWS.

LET'S WRITE A SMALL PROGRAM JUST TO ILLUSTRATE THEIR USE. FROM THE COMMAND MODE, WE ENTER I (FOR INSERT), AND HIT RETURN. THE LINE NUMBER 001 IS PRESENTED, AND WE HAVE A CURSOR. WE NOW ENTER THE FIRST LINE OF OUR PROGRAM:

```
001 CLS   CALL 01C9H
```

AND HIT RETURN AT THE END OF THE LINE. NOTE THE -H SUFFIX WHICH MUST BE USED WITH ALL HEX NUMBERS. THE USUAL RUBOUT AND INSERT CONTROLS CAN BE USED WHILE EDITING. WE NOW HAVE LINE 002, AND WE ENTER OUR SECOND LINE:

```
002  JP 1A19H
```

NOTE THAT THERE ARE TWO SPACES BETWEEN THE LINE NUMBER AND THE OPCODE - THIS TELLS THE EDITOR THAT THERE IS NO LABEL ON THIS LINE. WHENEVER WE ENTER A LINE WITHOUT A LABEL, WE MUST HIT THE SPACE BAR BEFORE ENTERING THE OPCODE.

AFTER HITTING RETURN AGAIN, AND HAVING LINE NUMBER 003 DISPLAYED, WE RETURN TO THE COMMAND MODE WITH CTRL BREAK. WE CAN LIST OUR PROGRAM WITH THE L (LIST) COMMAND. WE CAN DELETE LINE 002 WITH THE COMMAND D2. ALTERNATIVELY, WE CAN DELETE BOTH LINES WITH D1:2 (THE COLON INDICATES A RANGE OF LINE NUMBERS). THIS FORMAT CAN ALSO BE USED WITH THE L AND E COMMANDS (SEE DETAILS OVERLEAF). IF WE WANT TO ADD TO OUR PROGRAM, WE RE-ENTER THE EDIT MODE BY ENTERING AN I. IF WE HAD WANTED TO INSERT LINES BETWEEN LINE 1 AND 2, WE WOULD USE I1 (INSERT AFTER LINE 1). LINE RE-NUMBERING IS TAKEN CARE OF BY THE EDITOR.

LABELS

LABELS MUST OCCUPY THE FIRST FOUR CHARACTER POSITIONS IN A LINE. THE FIRST CHARACTER MUST BE ALPHABETIC, BUT THE OTHERS CAN BE NUMERIC.

COMMENT LINES

UNLIKE OTHER ASSEMBLERS, THIS ASSEMBLER CANNOT ACCEPT COMMENTS ON THE SAME LINE AS THE OPCODE (DUE TO THE 32 COLUMN WIDTH OF THE VZ). COMMENTS MUST BE PLACED ON A LINE OF THEIR OWN, BEGINNING WITH A SEMI-COLON (;). FOR EXAMPLE:

004 ;THIS IS A COMMENT LINE

SAVING, LOADING & MERGING

THE FOLLOWING COMMANDS ARE AVAILABLE FOR DISK OPERATION:

TS:FILENAME - SAVE THE SOURCE FILE TO DISK. ONLY THE FIRST EIGHT CHARACTERS OF THE FILENAME WILL BE USED. THE FILE WILL BE SAVED WITH AN 'S' PREFIX, AND CAN BE RELOADED BY THE ASSEMBLER. YOU MAY USE DIGITS IN THE FILENAME, BUT IF YOU DO, THE FILE CANNOT BE DCOPYED OR ERASED FROM BASIC.

TL:FILENAME - LOADS THE SPECIFIED SOURCE FILE FROM DISK. INSERT THE DISK AND CLOSE THE 'DOOR' BEFORE PRESSING RETURN.

TO:FILENAME - SAVES THE CURRENT OBJECT CODE TO DISK. THE FILE IS SAVED WITH A 'B' PREFIX SO THAT IT CAN BE BRUN FROM BASIC. DO NOT USE DIGITS IN THE FILENAME.

TM:FILENAME - MERGES THE SPECIFIED SOURCE FILE FROM DISK WITH THE SOURCE FILE CURRENTLY IN MEMORY. LINE RENUMBERING OCCURS AUTOMATICALLY.

TB:EXIT TO BASIC

IF AT ANY STAGE YOU WANT TO USE THE OTHER STANDARD DOS COMMANDS, SUCH AS DIR OR ERA ETC, YOU CAN EXIT THE EDITOR AND RETURN TO BASIC BY SIMPLY ENTERING THE TB COMMAND. IN FACT YOU CAN DO WHAT YOU LIKE WHEN IN BASIC, HOWEVER, IF YOU WANT TO RETURN TO THE EDITOR, YOU MUST NOT ALTER MEMORY IN ANY WAY. SIMPLY ENTER X AND RETURN FROM BASIC TO GET BACK TO THE COMMAND MODE OF THE EDITOR.

DISK CONVERSION FOR THE DICK SMITH EDITOR ASSEMBLER

This software package allows you to convert your old tape-based Disk Smith Editor Assembler to full disk operation. After being converted, your Editor Assembler (EDASM) will use NO MORE MEMORY than usual, and will be capable of loading source files from, and saving both source and object files to disk. Source codes are saved with an 'S' prefix, object codes with a 'B' prefix - they can be BLOADed or BRUN as usual. The conversion process can be split into two stages, where the EDASM is converted initially for Tape Load/Disk Save. Use this version to copy your old source files on to disk (do not attempt to save object files at this stage). The EDASM is then converted for full disk use, which includes a command to exit the EDASM if you need to do a DIRectory, ERAsure or any other DOS command! You may then return to the EDASM with a single command.

USING THE DISK EDASM :-

When fully converted to disk operation, the EDASM will still use the T- prefix commands for loading/saving/merging. Each command is explained below:

TS FILENAME - Save a source file to disk. Only the first eight characters of the filename will be used. The file will be saved with a 'S' prefix, and can only be reloaded by the EDASM.

TO:FILENAME - Saves the current object code to disk. The file will be saved with a 'B' prefix, so that it can be loaded and run from BASIC. Do not use any digits in the filename or a 'FILE NOT FOUND' error will occur.

TL:FILENAME - Loads the specified source file from disk. Insert the disk and close the door before pressing RETURN.

TM:FILENAME - Merge the specified filename from disk with the source code in memory. Same proceedure as TL.

TB - Exit the EDASM to BASIC. Use this command to enter any other DOS commands, such as DIR or ERA etc. You can return to the EDASM by simply entering: X <RETURN>. Any source code or assembled object code will remain intact, providing you have not altered any memory.

Note: The TV command is no longer supported.

CONVERTING YOUR EDASM :-

Firstly, you will need a copy of the DSE Editor Assembler on disk. If you have it on tape, start at step 1, otherwise start at step 3.

STEP 1 - Insert Conversion disk. Enter: BRUN"BICOPIER". Insert your EDASM tape in your datassette and press PLAY.

STEP 2 - After tape EDASM has loaded, remove disk from drive and insert a blank, initialised disk. At the top of the screen you should see a command to save the EDASM to disk. Move your cursor up and type you own name (such as EDASM) over 'filename', and then type over the hex numbers which follow. Hit return and the EDASM will be saved to disk. You now have a disk copy of it, but the tape commands still need to be changed.

STEP 3 - To convert your EDASM to tape load/disk save. Insert the disk with your EDASM and enter:

BLOAD"(insert its name here)"

Now insert the conversion disk, and enter :

BLOAD"PATCH A": BLOAD"PATCH B": BLOAD"PATCH C"

Replace your EDASM disk, and enter :

BSAVE"(insert a new name here)",7AFD,A2A1

You have now saved a version of the EDASM for copying your old source files from tape to disk. Do this before proceeding to step 4.

STEP 4 - Converting a tape/disk EDASM to full disk operation. Insert the disk on which the new EDASM was saved. Enter :

BLOAD"(the filename you chose)"

Now insert the conversion disk and enter :

BLOAD"PATCH D"

Replace your disk and (finally) enter:

BSAVE"(another new name eg: DISKEDAS)",7AFD,A2A1

You have now saved a final copy of the EDASM with full disk functions.

If you have any problems, don't hesitate to contact us at:
P.O Box 154, DURAL NSW 2158