

**ZEP 100 User Guide**

# **TORCH**

**COMPUTERS**



---

---

# **ZEP100 User Guide**

# **TORCH**

**COMPUTERS**

© TORCH COMPUTERS LTD  
Abberley House, Great Shelford, Cambridge CB2 5LQ

TORCHMAIL Telephone (0223) 840238

Password: Secret.

---







---

# **Contents**

|   |               |
|---|---------------|
| <b>ASSEMBLY INSTRUCTIONS</b>                      | <b>1</b>      |
| <b>BUILT-IN COMMANDS</b>                          | <b>7</b>      |
| <b>FILES AND FILENAMES</b>                        | <b>35</b>     |
| <b>UTILITY PROGRAMS</b>                           | <b>37</b>     |
| <b>TORCH DISC SYSTEM</b>                          | <b>59</b>     |
| <b>PREPARING COMMAND FILES</b>                    | <b>65</b>     |
| <br><b>APPENDIX A Disc Filing System Commands</b> | <br><b>67</b> |
| <b>Glossary</b>                                   | <b>73</b>     |
| <b>Index</b>                                      | <b>75</b>     |

---

---

# TORCH

COMPUTERS

---



---

# INTRODUCTION

Your TORCH Z80 Extension Processor transforms your BBC micro into a powerful twin processor business machine. The Z80 processor, together with the MCP operating system allows you to run the full range of CP/M compatible software, such as language compilers, (PASCAL, BCPL, FORTRAN, etc.) Word Processors, (PERFECT WRITER, WORDSTAR, MAILMERGE, etc.) and a wide range of business software.

Supplied with the ZEP100 package are:-

- ★ The processor card
- ★ The TORCH CP/M compatible operating system (2 ROMs)
- ★ 64K RAM (located on the Z80 card)

As the 6502 processor in the BBC machine handles all the peripheral and I/O devices, this means that nearly 63K of RAM is available when running CPN programs.

This User Guide is intended to give you enough information to connect the Z80 Extension Processor to your BBC micro, and exploit the capabilities of this powerful combination to the full. A description of the TORCH built-in commands and the free utility software is provided, together with a brief outline of the TORCH disc system and how it may be used for storing your BBC BASIC programs.

Normally your TORCH dealer will carry out the installation of the Z80 card and check that the drives you will use it with are of suitable quality and performance.

If for some reason you are going to fit the Z80 card yourself then read ALL of the assembly instructions carefully before connecting the Z80 card to your BBC micro. If you are in any doubt, contact your dealer.

The Operating System supplied with your ZEP100 allows you full TORCHNET networking. This means that if the BBC micro you are using with your Z80 Extension Processor is connected to a TORCHNET network, you will be able to access disc drives on other TORCH computers or BBC computers with a ZEP100 or a Z80 Disc Pack.

For further information on TORCHNET or to purchase a TORCHNET USER GUIDE, please contact your dealer.

Acorn is a registered trade mark of Acorn Computers Limited.

---



---

# TORCH

## COMPUTERS

---

### INTRODUCTION

The purpose of this document is to provide a comprehensive overview of the Torch Computers system. This document is intended for users who are new to the system and need to understand the basic concepts and components. It covers the hardware, software, and network configuration of the system.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance. The system is designed to be easy to use and to provide a high level of performance. The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance. The system is designed to be easy to use and to provide a high level of performance. The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance. The system is designed to be easy to use and to provide a high level of performance. The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

The system is designed to be easy to use and to provide a high level of performance.

---

# ASSEMBLY INSTRUCTIONS

NOTE: Before connecting your Z80 Extension Processor (ZEP) to the BBC machine, it is important to confirm that the BBC machine has been upgraded properly from model A to B, and that all the modifications (where necessary) have been carried out correctly. Your Acorn dealer should upgrade your BBC machine for you.

## UPGRADING THE BBC MACHINE

The following components are the disc interface components which must be fitted to the BBC machine before connecting the Z80 Extension Processor. All of the following items are available as a kit from TORCH computers. If you already have disc drives attached to your system, move straight to the section – INSERTING THE ROMS as your machine will already have been upgraded.

### Disc Interface Kit

| Circuit Location. (BBC PCB) | Type number.  |
|-----------------------------|---|
| IC78                        | 8271  |
| IC79                        | 7438  |
| IC80                        | 7438  |
| IC81                        | 74LS393   |
| IC82                        | 74LS10  |
| IC83                        | CD4013B   |
| IC84                        | CD4013B   |
| IC85                        | CD4020B   |
| IC86                        | 74LS393   |
| IC87                        | 74LS123   |
| IC88                        | Disc Filing System (DFS) ROM  |
| IC51                        | V1.2 Machine Operating System (MOS) carrier.<br>(Butterfly PCB)<br>(or ROM) |

The following points must also be checked on the BBC machine before connecting the Extension Processor:

1) IC76 (74LS163), and IC77 (74LS00), should have been fitted when the machine was upgraded from model A to B.



2) If the BBC printed circuit board is issue number 3, the following link positions must be confirmed:

IC27 pin 9 (14 pin IC type SN 7438) is connected via a wire link to IC78 PIN 11. (To make the link bend the pins up).

Links S12 and S13 are broken.

3) If the BBC printed circuit board is issue number 4

Links S9, S12, and S13 are broken.

NOTE: (The issue number of the PCB is printed to the right of the ACORN symbol.)

Having checked that your BBC machine has been upgraded to a model B and that the Disc Interface kit has been fitted correctly, (i.e. no bent under pins etc) you can now connect your Z80 Extension Processor by carefully following the procedures given below.

## **INSERTING THE ROMS (Refer to Figure 1)**

Please carry out all these instructions carefully, and do not rush them. If you have any doubts at any stage do not hesitate to contact your dealer.

**DISCONNECT THE MAINS POWER SUPPLY AND COMPLETELY REMOVE ANY SOURCE OF MAINS ELECTRICAL CONNECTION FROM THE BBC MACHINE.**

Carefully remove the lid from your BBC micro computer.

Remove the three keyboard retaining bolts, and remove the keyboard to reveal the main PCB and ROMs. The following procedure describes how to fit the ROMs on your BBC machine:

NOTE: If the BBC machine already has a disc interface then ignore A and B below.

A) The MOS carrier PCB (part of the Disc Interface Kit) should be plugged into the fifth socket from the right hand side of the PCB (first remove the MOS ROM already in this socket). If you already have a MOS version 1.2 then this need not be changed.

B) The DFS ROM should be plugged into the socket immediately to the right of the BBC BASIC interpreter ROM.

C) Fit the MCP ROM (supplied with the Z80 Extension Processor) in the first socket from the right hand side of the PCB. (Checking that the notch on the ROM is facing the rear of the BBC Computer.)



D) Check that the links S32 and S33 (in front of the MCP ROM) are linked towards the centre of the board.

E) Fit the CCCP ROM onto the Z80 Eurocard with the notch facing the connector.

Great care should be taken when inserting or removing ROMS.

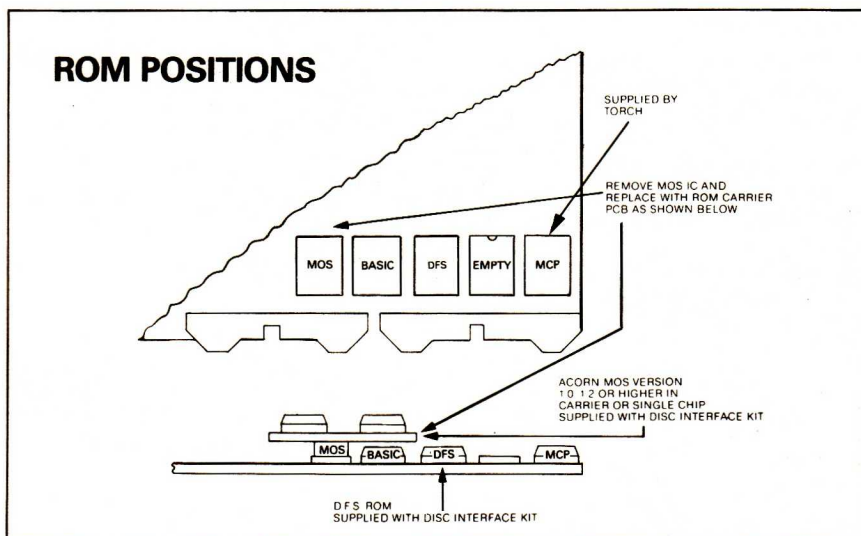


Fig. 1

## INSTALLING THE Z80 EUROCARD

(Refer to Figure 2)

The Z80 Eurocard should be mounted on the lid of the BBC box using the self-adhesive mounts provided. It should be situated above the RAM area with its connector at the right hand side of the box. It is important to mount the Z80 card in this manner so that the 200mm cable provided will be of sufficient length to connect the Z80 card to the TUBE interface on the BBC machine. The following procedure should then be followed.

- A) Connect one end of the Z80 ribbon cable to the TUBE connector on the BBC micro. (The TUBE connector is the nearest to the right hand side of the BBC computer, it is marked PL12.)
- B) Re-attach the keyboard using the three bolts removed previously. Make sure the anti-scrape card provided is protecting the ribbon cable from any chaffing by the underside of the keyboard.
- C) With the lid of the BBC box almost fitted connect the Z80 ribbon cable to the Z80 card, and screw down the lid of the BBC box.

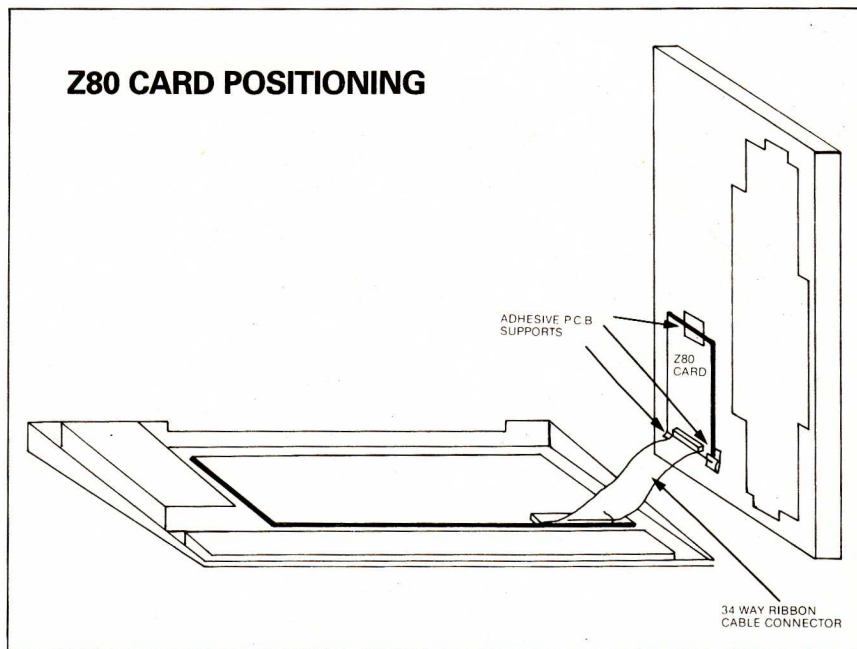


Fig. 2

---

## **TROUBLESHOOTING**

This section tells you what to do if your BBC computer and Z80 Extension Processor do not function properly after assembly. Given below are fault symptoms and possible causes.

### **No screen display and no response to keyboard**

Check that the power lead from the BBC machine is connected to a suitable mains supply. (250 Volts AC 50 HZ in Europe, 110 Volts 60 HZ in North America.)

### **Continuous 'beep' and no response to keyboard**

Check that the ROM carrier PCB has been plugged in correctly and that all the ROMS have been fitted in the correct positions and the right way round.

### **Single 'beep' and a flashing cursor, but no prompt**

Check that the links S32 and S33 are linked towards the centre of the main PCB.

### **Double 'beep' and a continuous display of 'No Z80' on the screen**

Check that the CCCP ROM has been correctly installed and that the legs of the ROM are not bent.

### **CPN operating system o.k. but Disc Drives not working**

A) Confirm all link positions are correct for that particular issue of BBC printed circuit board.

B) Check that the Disc Interface components have been fitted in the correct location, and that all ICs have been inserted the right way round. (If any are found to be the wrong way round then they should be replaced as they have probably been damaged).

D) Check that the connector from the Disc Drives to the BBC Disc interface socket has been plugged in correctly.



---

# TORCH

## COMPUTERS

---

---

# BUILT-IN COMMANDS

## INTRODUCTION

The TORCH operating System, CPN, controls the manipulation of data and all the operations that the Z80 Extension Processor can perform. Commands are words that the operating system will recognise as definite instructions and act on immediately. Several commands are 'built into' the operating system so that as they are typed in at the keyboard, they are understood and acted on by the computer. Apart from these 'built-in' commands there are also a number of other commands, these will be dealt with in the following section 'FILES AND FILENAMES', which may be read in conjunction with this section.

The following pages describe the 'built-in' commands and how they are used. Each description is split into three parts:

- 1) The explanation of the command giving an outline of where and when the command may be used. It may also include a table of options and or a list of exceptions and special cases.
- 2) The command syntax together with a list of any additional arguments that may be used with the command.
- 3) Examples of uses of the command. These are some examples of how the command is used. Once you have copied your master discs and stored the originals for safe keeping, it is a good idea to practise using the commands with some of the examples given.

To enter a command, type the appropriate word on the keyboard together with any additional information the command may require and press **return** (upper or lower case letters may be used when entering a command). Note that several commands have an alternative shorter form.

The following abbreviations are used in this section to show the parameters necessary with each command. Some of these parameters may be optional such as the drive letter or the filename extension.

When the 'filespec' argument is used, the drive or surface letter (D:) and the extension (.ext) are usually optional.

| Parameter | Description   |
|-----------|---|
| D:        | Disc drive or surface letter e.g. A:, B:, C:.   |
| filename  | <p>Any valid filename (max. 8 characters). The following characters can be used in filenames:</p> <p>A-Z 0-9 ! @ \$ % &amp; + - /</p> <p>Any other character is treated as a separator.</p> |
| .ext      | <p>Any valid extension (3 characters).</p> <p>The characters listed above are valid in filename extensions.</p>   |
| filespec  | {D:}filename{.ext}  |
| <code>    | A code representing a number from a table or list.  |
| <N>       | Any number (A range may be given).  |
| { }       | Anything enclosed in curly braces in a Command Syntax is an optional parameter.   |
| *         | Any number of characters in a filename or extension.  |
| ?         | A single character in a filename or extension.  |



---

# BACKGROUND

## EXPLANATION

The BACKGROUND command is used to change the background colour of the screen. The command may be abbreviated from BACKGROUND to B and the colour is indicated by a number between 0 and 15. The choice of colours are as follows:

|           |                           |
|-----------|---------------------------|
| 0 Black   | 8 Black/white flashing    |
| 1 Red     | 9 Red/cyan flashing       |
| 2 Green   | 10 Green/magenta flashing |
| 3 Yellow  | 11 Yellow/blue flashing   |
| 4 Blue    | 12 Blue/yellow flashing   |
| 5 Magenta | 13 Magenta/green flashing |
| 6 Cyan    | 14 Cyan/red flashing      |
| 7 White   | 15 White/black flashing   |

## COMMAND SYNTAX

BACKGROUND <code> **return**

B <code> **return**

## EXAMPLES

B 0 **return**

This would set the background colour to black (default).

B 10 **return**

This would set the colour of the background to green and magenta flashing.

NOTE: If, when you change the colour, the screen becomes a blank, you have probably set the foreground and background colours the same.

## BASIC

### EXPLANATION

The BASIC command is used to run the BBC BASIC interpreter provided with your BBC Micro. To return the computer to the CPN command line, use the \*MCP command (see the explanation of the \*MCP command on page 22).

If you intend to use BBC BASIC to run programs, you should hold the 'B' key down and press the BREAK key. This will clear all MCP information from the BBC programming space and allow you to run larger programs.

If you have used this, then you should return to the CPN command line by holding the CTRL and SHIFT keys down whilst pressing BREAK.

NOTE: It may be necessary to hold the CTRL key down and press BREAK before the Acorn DFS commands may be used.

### COMMAND SYNTAX

BASIC **return**

### EXAMPLE

BASIC **return**

This will allow you to use the BBC BASIC interpreter.

---

## COMMAND

### EXPLANATION

The COMMAND statement (which can be abbreviated to C) is used to cause the contents of a file to be treated by CPN as a series of commands. Comprehensive facilities are available for the substitution of parameters when the commands are obeyed. If, when the filename is specified the extension is omitted, a default of .SUB is assumed.

There is a more detailed description of how to prepare command files on page 65.

### COMMAND SYNTAX

C filespec **return**

COMMAND filespec **return**

### EXAMPLES

COMMAND A:BEGIN.TXT **return**

This would output the file BEGIN.TXT from the disc in drive A as a series of commands.

C TEST **return**

The operating system would add the extension .SUB to the file TEST and then output the contents of the file as a series of commands. The computer would assume that the file is stored on the logged on disc.



## COPY

### EXPLANATION

COPY is used to make a copy of a file (or a group of files) onto another disc or disc surface, or to make another copy of the file on the same disc.

The COPY command is particularly useful for making backup copies of single files rather than all the files on a disc. Another feature of the COPY command is the ability to copy a group of files using ambiguous filenames. This facility makes use of the following two characters; '\*' and '?'.

### COMMAND SYNTAX

COPY filespec TO filespec **return**

### EXAMPLES

COPY A:SYS.MAN TO B:SYS.MAN or COPY A:SYS.MAN TO B:

The above two commands perform the same function. They will both copy the file SYS.MAN from the disc in drive A to the disc in drive B.

COPY SY\*. \* TO B: **return**

This command will copy all files on the logged on disc drive with SY as the first two letters to the disc in drive B. The following files would be copied:

SYSTEM  
SYS.DOC  
SY001

COPY SYS.COM TO SYS1.COM **return**

The above command would copy the file SYS.COM to SYS1.COM and leave the original file intact.

---

## DELETE

### EXPLANATION

The DELETE command removes files from a disc or disc surface once they are no longer required. Before the DELETE command removes the file, it will ask:

Do you want to delete D:filename.ext ?

to which you must reply:

Y to delete the file,  
N to not delete the file,  
Q to not delete the file and to stop the command.

An option available with the DELETE command is the [u] option. If the filename is followed by this option, then the file will be deleted Unconditionally, that is, you will not be asked if you want to delete the file. The DELETE command can be abbreviated to DEL.

### COMMAND SYNTAX

DELETE filespec **return**

DEL filespec {[u]} **return**

### EXAMPLES

DEL A:SYS.MAN **return**

This will cause the command to ask you if you wish to delete the file SYS.MAN from the disc in drive A. If you answer Y, the file will be deleted otherwise the file will not be deleted and the command will end.

DEL SY\*. \* [u] **return**

This will automatically delete all the files whose first two letters are SY from the logged on disc.

## DIR

### EXPLANATION

The DIR command is used to display a table containing certain Directory information about the files stored on a particular disc. There are 4 options available with the DIR command, these are:

[L] will display the length of the specified file.

[P] will display the protection associated with the file or files specified. See the description of the PROTECT command.

[S] will display the space left on the disc.

[u] where u is the user number, will display all the files on the disc with the user number specified.

NOTE: Any number of options may be enclosed inside the square brackets.

### COMMAND SYNTAX

DIR {filespec} {[options]} **return**

### EXAMPLES

DIR B:**return**

The above example would list all the files on the disc or disc surface in drive B.

DIR SYSTEM.\* [lp]**return**

This would list all the files with the filename SYSTEM and any extension, that are stored on the logged on disc drive or surface. The [lp] options would also cause the lengths of the files and the protections allocated to the files to be listed.

DIR \*.DOC **return**

This will list all the files that are stored on the logged on disc drive or surface with the extension .DOC.



---

# DUPLICATE

## EXPLANATION

The DUPLICATE command will copy the contents of the disc in drive B to the disc in drive A. Before the contents of the source disc (B:) is copied to the destination disc (A:) the DUPLICATE command formats the destination disc.

When the command is working, it is looking on the disc for valid information to copy. If there are some tracks on the disc without any data on them, the command will ignore these tracks. There is a special option available with the command called FULL. If this option is used, the command will copy the whole of the disc without checking to see what data it is copying.

NOTE: the DUPLICATE command can be abbreviated to DUP.

## COMMAND SYNTAX

DUPLICATE {FULL} **return**

## EXAMPLES

DUP **return**

This will format the disc in drive A and then copy the contents of the disc in drive B to the disc in drive A.

DUP FULL **return**

This will copy the whole of the disc in drive B to the disc in drive A without looking to see what it is copying.

## FOREGROUND

### EXPLANATION

The FOREGROUND command is used to change the foreground colour of the screen. The command may be abbreviated from FOREGROUND to F and the colour is indicated by a number between 0 and 15. The choice of colours are as follows:

|           |                           |
|-----------|---------------------------|
| 0 Black   | 8 Black/white flashing    |
| 1 Red     | 9 Red/cyan flashing       |
| 2 Green   | 10 Green/magenta flashing |
| 3 Yellow  | 11 Yellow/blue flashing   |
| 4 Blue    | 12 Blue/yellow flashing   |
| 5 Magenta | 13 Magenta/green flashing |
| 6 Cyan    | 14 Cyan/red flashing      |
| 7 White   | 15 White/black flashing   |

### COMMAND SYNTAX

FOREGROUND <code> **return**

F <code> **return**

### EXAMPLES

F 7 **return**

This would set the foreground colour to white (default).

F 11 **return**

This would set the colour of the foreground to yellow and blue flashing.

NOTE: If, when you change the colour, the screen becomes a blank, you have probably set the foreground and background colours the same.

---

# FORMAT

## EXPLANATION

The FORMAT command is used to set up the necessary tracks and sectors on a floppy disc so that it may be used.

NOTE: Re-formatting a disc will cause all the data on that disc to be destroyed.

## COMMAND SYNTAX

FORMAT **return**

## EXAMPLE

FORMAT **return**

This will format a disc in one of the floppy disc drives depending on which letter you enter when the computer asks:

Enter drive to Format (type space to abandon)



## HELP

### EXPLANATION

The HELP command is used to display a list of the 'built in' commands and the arguments that are needed to use the commands. An extract from the help screen is:

|               |                     |
|---------------|---------------------|
| B, BACKGROUND | Colour number.      |
| BASIC         | Enter BBC BASIC.    |
| C, COMMAND    | Filename arguments. |

There are two ways of accessing the help screen:

- 1) Type HELP or H followed by a **return**.
- 2) Press the second red function key from the left.

### COMMAND SYNTAX

HELP **return**

H **return**

### EXAMPLE

HELP **return**

This command would cause the computer to display the HELP file on the screen.

---

# INPUT

## EXPLANATION

The INPUT command is used to enter a text file into the computer and then store it on a disc or disc surface.

You should type the command and then enter the text. Once all the text has been entered, move to a blank line and then press the END key or type CTRL-Z followed by a carriage return. The data you have typed, will now be saved in the file you specified in the command line.

There are some basic editing facilities available with the INPUT command, these are:

| KEY        | FUNCTION                                       |
|------------|--|
| Lower case | Move the cursor down.                          |
| Upper case | Move the cursor up.                            |
| Move left  | Move the cursor to left.                       |
| Move past  | Copies the character above the editing cursor. |
| Move right | Move cursor to right.                          |

NOTE: As soon as one of these screen editing keys has been pressed, a second cursor known as the editing cursor will appear on the screen. This will disappear from the screen every time a **return** is pressed but will reappear when one of the edit keys are pressed again. The delete keys on the right hand keypad, may be used to delete characters when using the INPUT command.

## COMMAND SYNTAX

INPUT filespec **return**

## EXAMPLE

INPUT A: SYS.DOC **return**

This document is to show  
how to enter a 3 line text  
file.

**^Zreturn**

If you entered everything in the above example you would create a three line file called SYS.DOC which would be stored on the disc in drive A, (or on the disc surface A).

## KEY (\*KEY)

### EXPLANATION

The KEY command is used to program the red function keys along the top of the keyboard. Any of these keys may be programmed to represent a given set of characters with a maximum length of 58. An example of this might be a company's name and address, or any piece of text that might have to be typed or printed frequently.

To use the command, enter KEY followed by the key number and the string of characters you wish to program into the key. To include control characters in the string, you should use the bar key which is situated above the oblique key on the keyboard.

It is also possible to type \*KEY instead of KEY.

NOTE: Keys 11 to 13 cannot be programmed unless the command \*FX4,2 is used first (refer to the BBC USER GUIDE for further information on this).

On 'power on', or reset, the following keys are defined:

|   |       |    |    |
|---|-------|----|----|
| 0 | QUIT  |    | IM |
| 1 | HELP  |    | IM |
| 2 | NO    |    | IM |
| 3 | YES   |    | IM |
| 9 | BEGIN | IX | IM |

Function key 9 has a CTRL-X before the word BEGIN, this clears the data to the beginning of the line before typing BEGIN and a carriage return.

### COMMAND SYNTAX

KEY <N> {<string of characters>} **return**

where <N> is a number in the range 0 to 13

### EXAMPLES

KEY 2 TORCH **return**

The above example would be used to program the third key from the left to equal 'TORCH'.



---

KEY 6 VDU 12 I M **return**

This command is slightly more advanced in that it includes the control character 'I M' which represents a carriage return. If you now pressed the function key number 6, the VDU command 12 (clear text area) would be typed on the screen and the command would then be processed.

KEY 10 ' 1234 **return**

This command would program function key 10 to have 4 spaces before the numbers 1234, and 4 spaces after it. This is the best way of including spaces in a function key.

## **\*MCP**

### **EXPLANATION**

The \*MCP command returns control to the MCP operating system from BBC BASIC. It may also be used when running MCP to re-enter the system.

### **COMMAND SYNTAX**

\*MCP **return**

### **EXAMPLE**

\*MCP **return**

This would re-enter the operating system or return control to MCP from BBC BASIC.

---

## MODE

### EXPLANATION

The MODE command allows you to change the format of the screen. This means that you can increase or decrease the character size. There are seven different modes available and the codes that represent these modes are:

- 0 80\*32 text (or graphics).
- 1 40\*32 text (or graphics).
- 2 20\*32 text (or graphics).
- 3 80\*25 text.
- 4 40\*32 text (or graphics).
- 5 20\*32 text (or graphics).
- 6 40\*25 text.
- 7 40\*25 teletext display.

NOTE: In the above table 80\*32 means 80 columns by 32 lines.

The teletext display (mode 7), is the mode used for the British Telecom PRESTEL system.

When the computer is first switched on the screen appears in mode 0 and once a MODE command has been issued it will stay in this mode of operation until:

- a) A new mode command is given.
- b) An application program changes the screen mode.
- c) The BREAK key is pressed (this will default back to mode 0).

NOTE: Changing the mode clears the screen, selects a white foreground on a black background and displays the system prompt.

### COMMAND SYNTAX

MODE <code>**return**



---

# TORCH

## COMPUTERS

---

### EXAMPLES

MODE 0 **return**

This is the default mode that allows 32 rows of characters and 80 columns.

MODE 7 **return**

The above example would set the screen to the mode used by the British Telecom PRESTEL system.

---

# PRINT

## EXPLANATION

The PRINT command is used to print out information from files stored on disc onto a printer. A printer may be attached to your computer via the parallel or serial ports.

NOTE: \*FX followed by the correct code to select a printer must be entered before the PRINT command. This is to specify what type of printer is connected (refer to the BBC USER GUIDE for more printer information).

To abort the printing, press the escape key. The computer will ask you if you wish to stop the printing or carry on. You should respond to this by pressing the Y key for yes and any other key for no.

## COMMAND SYNTAX

PRINT filespec **return**

## EXAMPLES

PRINT TORCH.DOC**return**

This would send the file TORCH.DOC from the disc in the logged on disc drive or surface to the currently selected printer port.

PRINT B:ALEWIS **return**

This would send the file ALEWIS from drive or surface B to the printer.

NOTE: It is up to you to ensure that the printer is fully operational (e.g. there is enough paper and the printer is switched on).

## PROTECT

### EXPLANATION

The PROTECT command is used to protect files in various ways depending on the option/s selected from the following table (if no option is specified then the command will display the current protection for the files).

s: File is invisible to the DIR command.  
u: This option cancels the s option.  
r: File is read only.  
w: File is read/write, cancels r option.

### COMMAND SYNTAX

PROTECT filespec {[options]} **return**

### EXAMPLES

PROTECT TORCHTEL.COM [r] **return**

The command above will make the file TORCHTEL.COM read only. This means that it cannot be deleted or modified until another PROTECT command has been used.

PROTECT B:\*.COM [s] **return**.

This makes all files on drive B with extension .COM (program files) into system files, which means in effect they do not appear on the directory, but may still be used.



---

# RENAME

## EXPLANATION

The RENAME command is used to rename a file (or a group of files), which may be useful if you want to regroup information or put outdated information into storage and release the filename or names to use again. This command can also make use of the ambiguous characters, \* and ?. The command can be abbreviated from RENAME to REN.

NOTE: It is possible using RENAME to give a file the same name as an already existing one, but the operating system will then not be able to differentiate between the two files.

## COMMAND SYNTAX

REN filespec AS filespec **return**

## EXAMPLES

REN SYS.MAN AS DISC.MAN **return**

This will rename the file SYS.MAN on the logged on disc drive or surface as DISC.MAN.

REN AL.\* AS RA.\* **return**

This will rename all files with AL as the first two letters to RA with the first two letters e.g.

| Old Filename | New Filename |
|--------------|--------------|
| AL.DOC       | RA.DOC       |
| ALSYS.MAN    | RASYS.MAN    |

## SAVE

### EXPLANATION

This is a specialised command used to save data stored in the TPA (Transient Program Area). The command will save the data starting from the memory address 0100H in pages of 256 bytes. The number specified in the command line is the number of pages to be saved.

### COMMAND SYNTAX

SAVE <N> filespec **return**

### EXAMPLES

SAVE 16 DATA.COM **return**

This will save the first 16 pages (4K) of memory, starting from 0100H into the file DATA.COM on the logged on disc drive or surface.

SAVE 64 A:PROGRAM.COM **return**

This will save the first 64 pages (16K) of the TPA into the file PROGRAM.COM.

---

# TIME

## EXPLANATION

The TIME command is used to read the time from when the computer was switched on or from when the time was last set. The time is displayed in the format

The time is 06:25:30

To set the time a utility program called SETTIME.COM is normally required. This program is included on your system disc.

## COMMAND SYNTAX

TIME **return**

## EXAMPLE

TIME **return**

This would display the time from when the computer was turned on, or from when the time was last set.



## TYPE

### EXPLANATION

The TYPE command is used to display the contents of a file on the screen.

When looking at the contents of a file pressing the escape key will interrupt the display, and you will be asked if you wish to stop the display or carry on. You should respond to this by pressing the Y key for yes or the N key for no.

NOTE: It is possible to stop the screen scrolling every 22 lines and allow you to view the data (refer to the VDU commands 14 and 15 on page 33).

### COMMAND SYNTAX

TYPE filespec **return**

### EXAMPLES

TYPE REPORT.DOC **return**

This will display the contents of the file REPORT.DOC.

TYPE A:FILE.BAK **return**

This will display the contents of the file FILE.BAK.

---

## USER

### EXPLANATION

The USER command is used to allocate a number (as a label) to a group of files on a disc. The reason one might do this, is that if there are many files on a disc, it is possible to select only the files required, by specifying the user number of that group of files. If the user number is between 1 and 31 and a DIR command is executed, then the files with a user number 0 will not be displayed.

NOTE: The files with a user number 0 are always available for use although they might not always appear on the directory.

### COMMAND SYNTAX

USER <N> **return**                      where <N> is in the range 0 to 31

### EXAMPLE

USER 5 **return**

This will only allow access to the files with a user number 5 or 0.

## VIEW

### EXPLANATION

The VIEW command is used to display a picture file on the screen. It takes a file that uses the VDU control functions, and outputs them to the screen driver which displays them on the screen. To stop the display of a file, press the escape key. Files that are displayed with the VIEW command usually have the extension of .PIC and if no extension is given, the computer assumes the extension of .PIC.

NOTE: Some picture files terminate with the VDU code 21 (disable screen) so that commands can be issued without corrupting the display. VDU 6 will restore that output.

### COMMAND SYNTAX

VIEW filespec **return**

### EXAMPLE

VIEW BGRAPH.PIC **return**

This will display the contents of the file BGRAPH.PIC on the screen.



---

## VDU

### EXPLANATION

The VDU command may be used to send given control codes to the screen driver (The screen driver is part of the operating system software, and is used to print and draw on the screen). There are various functions which are available with this command which are listed below with their appropriate number codes.

To use the VDU command enter VDU followed by a space and the number code required.

| Number<br>Codes |   | Function                                |
|-----------------|---|---|
| 0               |   | Null (does nothing)                     |
| 1               | 1 | Send character to printer               |
| 2               |   | Enable printer (switch<br>printer on)   |
| 3               |   | Disable printer (switch<br>printer off) |
| 4               |   | Separate cursor                         |
| 5               |   | Join cursors                            |
| 6               |   | Enable VDU driver                       |
| 7               |   | Ring bell                               |
| 8               |   | Cursor left                             |
| 9               |   | Cursor right                            |
| 10              |   | Cursor down                             |
| 11              |   | Cursor up                               |
| 12              |   | Clear text area                         |
| 13              |   | Move cursor to start of line            |
| 14              |   | Page mode on                            |
| 15              |   | Page mode off                           |
| 16              |   | Clear graphics area                     |
| 17              | 1 | Define text colour                      |
| 18              | 2 | Define graphics colour                  |
| 19              |   | Define colour relationship              |
| 20              |   | Reset colour relationships              |
| 21              |   | Disable VDU driver                      |
| 22              | 1 | Select mode                             |
| 23              | 9 | Define character                        |
| 24              | 8 | Define Graphics window                  |
| 25              | 5 | Plot                                    |
| 26              |   | Reset windows                           |
| 27              |   | Escape ( ASCII 27 )                     |

| Number Codes |   | Function                            |
|--------------|---|-------------------------------------|
| 28           | 4 | Define text window                  |
| 29           | 4 | Define Graphics origin              |
| 30           |   | Home text cursor                    |
| 31           | 2 | Position text cursor                |
| 127          |   | Delete character                    |
| <N>          |   | Any ASCII printable character code. |

The second column of numbers give the number of parameters (in bytes) needed with the code.

## COMMAND SYNTAX

VDU <N>,<N>,<N>.....<N> **return**

## EXAMPLES

VDU 65 **return**

This sends the character with the ASCII value 65 to the screen. (i.e prints the letter A).

VDU 7 **return**

This produces a short 'beep' on the speaker.

VDU 12 **return**

This command would clear the current text area and position the cursor in the top left hand corner of the screen.

VDU 2,1,<N>,3 **return**

This command would cause the following actions; the printer is enabled, the character <N> is sent to the printer, and the printer is then disabled.

NOTE: Some number codes have following characters to complete the command.

VDU 22,2 **return**

This will set screen MODE 2, but will not switch on the printer.

---

## FILES AND FILENAMES

When a piece of text or data is to be taken from the computer and stored on disc, it has to be put in a 'file' on the disc. This file has to have a name so that the computer can identify it at a later date, retrieve its contents, and load it back into the computer.

Filenames are made up of:

(1 to 8 characters). (0 to 3 characters,) where a character is defined as: a letter from A to Z, a number from 0 to 9, or any of the following: @, %, \$, \, /.

Examples of filenames:

GALLIVAN.DES  
DISC.TXT  
JMB/RA  
DEMO-1  
Z.X  
LETTER.001  
REM.

NOTE: In the above example GALLIVAN.DES, GALLIVAN is the filename, and .DES is known as the extension. If the extension is omitted, part of the CCCP operating system, or the current program may add one. (This is explained further below).

Within these limits the user may name his/her own files according to their own system. Any filenames that do not conform to the above requirements would be invalid and would not be accepted by the system.

## AMBIGUOUS FILENAMES

Where it is desirable to include a file within a certain group, without giving it a precise name you can use what is called an 'ambiguous' filename. This makes use of two characters.

? and \*.

These characters may be used to represent whichever character the user desires them to. Thus, ? in any position in a filename means 'any character'.

EXAMPLE: SYS.??? means SYS. followed by any characters.



---

# TORCH

## COMPUTERS

---

What this means is that a file starting with SYS. may have any optional extension, and will still be recognised by the operating system.

The \* character in any position has the same effect but it can mean any number of characters.

EXAMPLE: \*.COM means all files with a .COM extension.

\*.\* means all files on the default disc drive.

The default disc drive (A), is the drive the computer will access on 'power on', or after a reset, unless the logged on drive is changed. (See section: TORCH DISC SYSTEM.)

NOTE: When using the COPY command, (see section on COMMANDS) the disc drive name A or B becomes part of the filename.

EXAMPLE: A:SYS.MAN

Where A: is the disc drive, SYS is the filename, .MAN is the extension.

The name of the disc drive becomes part of the filename so that the operating system knows exactly on which disc drive it will find the file called SYS.MAN.

## COMMAND FILES AND SUBMIT FILES

If a filename is entered at the beginning of the command line without an extension, the TORCH operating system will add the extension .COM and then look for this on the disc. If it finds it then it will load it and run it. If the operating system adds the .COM extension to a filename, looks for it, and doesn't find it on the disc, then it will add the extension .SUB; it will then look for this on the disc. This is known as a SUBMIT file. Such a file will have in it a sequence of commands written as text. When it finds it, the operating system will use this file as its input rather than the keyboard. The operating system will then execute this file as a series of commands.

EXAMPLE: A file called SYS.COM may be executed by entering:

SYS **return**

A file called SYS.SUB may be executed as a command file by entering SYS **return** provided the file SYS.COM does not exist.



---

## UTILITY PROGRAMS

Your Z80 Extension Processor is supplied with a system disc containing utility programs. The utility programs provided are listed below with a brief description of each program and its function. To use any of the utility programs, put the system disc in drive A and type the name of the utility followed by **return**.

NOTE: Backup copies should be taken of all system programs before use.

As it is TORCH policy to upgrade and improve existing software, some versions of the Utility programs described below may not have all the features of future versions. However, all present software will be supported in future upgrades. (Software upgrades will be available from your dealer for a small handling charge.)

The following programs are described in this section:

- 1) FONT.
- 2) EXEC.
- 3) TORCHBUG.
- 4) RS423.COM
- 5) MUSIC.
- 6) MAPDISC.
- 7) POKEDISC
- 8) RDACORN/WRTACORN
- 9) FIND
- 10) COPIER
- 11) SETTIME

### FONT

This program allows the user to design character sets for special use or just for personal preference. The program is menu driven and should be self explanatory.

When FONT is run it will ask the user if he wants to modify an existing FONT file, or to create a new one from the current character font. Either enter the name of the file to be modified, or press **return** to create a new FONT file. Font will then display a list of options from which the user may design his own character font.

When selecting a character to be re-defined, the ASCII character code (in Hex), should be entered. Alternatively, enter = followed by the character to be selected.

The selected character is defined on an 8x8 grid around which a cursor is moved and is used to re-define the character. The cursor movement is controlled by the keys on a numeric key pad. Key 5 reverses the field of the pixel (picture element) under the cursor, all the keys around key 5 i.e. 4,6,8 and 2, are used to move the cursor left, right, up, and down, respectively. The character to be defined is shown in parentheses above the grid box in mode 0,1, and 2 sizes. At the bottom of the screen is a box showing the current character font for reference. While editing the character, a number of commands are available to alter the grid quickly; all of these commands are listed on the screen when the program is run.

The W command allows the user to allocate a filename to the new character font and save it on disc.

If a filename is given the extension .COM then this file may be saved and run as a command file.

EXAMPLE: To set up a font called GOTHIC.COM enter GOTHIC followed by **return**, and the operating system will add the .COM extension itself and load and run the file. (For an explanation of the syntax used for naming files, see the section on FILES AND FILENAMES.)

NOTE: FONT is not limited purely to producing character sets, it may also be used to produce graphic fonts to produce pictures or circuit diagrams for example. This is achieved by redefining the characters of the current font as graphic characters.

---

## **EXEC**

The EXEC command allows the user to specify up to approximately 120 characters that will be 'typed' after execution of a SUB file. The effect of any such command is exactly as though the characters in question had been read from the keyboard buffer.

NOTE: If SUPERVDU or other terminulators are to be loaded with a command (SUB) file, then the EXEC command should be used.

### **OPERATION**

The characters following the EXEC command are programmed into function key 8 and, if it is sufficiently long also into function key 7. The codes for key 8 (and 7 if necessary) are added to the keyboard 'typeahead' buffer. When the operating system next tries to read input from the command line in interactive mode the soft key definitions will be read and expanded. The format for the parameter given to EXEC is the same as that used by the function keys (i.e. vertical bar is an escape character).

### **RESTRICTIONS**

Only one EXEC command may be included in any SUB file or sequence of such files. This limits the total command length that may be EXECed.

The keyboard buffer is not cleared before execution of the program so it is advisable to include a control-X (entered as `IX`) as the first character in the buffer. Similarly a carriage return should be included at the end of the string if it is intended that the command should be terminated by one. This may not, of course, always be the case (see below).

The keys 8 and sometimes 7, are used by EXEC and therefore may not be set by the user when EXEC files are used. The key definition resulting from EXEC will of course remain in keys 8 and 7 and may be reused as necessary. If the string involved is sufficiently short only key 8 will be affected.

### **HELP SCREEN**

If the EXEC command is invoked without any parameters, then a screen of information describing the command will be displayed.



### EXAMPLES

Set up a SUB file which may be easily invoked by a user with little or no experience of computers, which will set up a printer and a yellow foreground on the screen and will leave the user in an MBASIC program called ACCOUNTS.

Since the program is to be run as a 'turnkey' system, (i.e. automatically runs a specific program on power on), the file will be named 'BEGIN.SUB'. This allows it to be executed by simply typing the CTRL+BEGIN keys together, with the disc in drive A.

The contents of this file might be as follows (lines beginning with a semicolon are comments):

```
; Set up a parallel printer
*FX5,1
; Set up screen
Mode 0
F 3
; Now run the accounts package
EXEC |XMBASIC ACCOUNTS |M
; Note that the |X clears the line of any unnecessary letters
; while the SUB file is being obeyed and the |M is
; necessary to ensure that the command will be
; entered
; Also note that while EXEC need not be the last command in a
; file there may only be one EXEC and it will not take effect
; until all other commands have been executed.
```

This file (with or without the comments) will have the desired effect

A programmer wishes to have a simple method of editing and assembling programs using WordStar, M80 and L80. The following submit file (called UPDATE.SUB) will set up function keys 4 and 5 to invoke the assembler, key 8 to invoke WordStar and leave the programmer in WordStar at the end of the file. It is assumed that the program under development is on drive B with the assembler etc. on drive A. Further, this programmer always begins files with a 'ruler line' on the second line of the file. Tab stops are set up by a sequence of control keys whenever the file is altered

```
; Set up key 4 to assemble the program.
KEY 4 M80 = B:$1
; and key 5 to do the link
KEY 5 L80 B:$1,b:$1/n/e
; now invoke WordStar
EXEC WS B:$1.MAC |M |X |OF |OV |OT
```

This file is invoked by a command of the form UPDATE EXEC to allow editing of the file EXEC.MAC.



---

# **TORCHBUG** (monitor and debugging aid)

## **AIM OF PROGRAM**

To provide a simple debugger program capable of loading files (either object code or Intel Hex), modifying and testing the code, and then saving the resulting program.

## **OPERATION**

TORCHBUG is invoked as an ordinary COM file. It loads and relocates itself automatically at the top of the free store. If a filename is given on the command line as a parameter then that file will be loaded at 0100H but not executed. From TORCHBUG the following command keys may be used.

**HELP** (Function key 1) This key will always reveal some information about what options are available from the part of the program in use. It usually leaves the user in the same mode as if no key had been pressed, but an exception to this is the Modify command where the program will return to the TBUG> prompt if help is requested.

**QUIT:** (Function key 0) This key returns the user to the 0A> command line. A cold boot is performed, removing TORCHBUG from the system but preventing SAVE being used. N.B. Because the save command will not work after TORCHBUG has been run, always use the S command from TORCHBUG. (A 'cold boot' refers to a condition whereby the 6502 is reset.

This is the same effect as pressing the BREAK key on the BBC keyboard.

**G GO.** This command allows the user to execute a program from any memory location. The address to go from should be entered as a hexadecimal number terminated by a carriage return.

**I Load Intel hex file.** The name of a hex file will be prompted for and should be entered in the format outlined below. Errors detected in the standard hex format will cause the file to stop loading with an error message. The file will be displayed as it loads.

**L Load COM file.** The program will prompt for a filename and then the first location in memory from which the program should be loaded. No checks are made to prevent overwriting of the CPN etc.

**M Modify memory.** This command allows dumping and modifying of blocks of memory. The first address will be prompted for as in G above. memory is displayed in blocks of 16 bytes in HEX format followed by the sixteen characters made from those numbers with the MSB set to zero and non printing characters displayed as '.'.

---

# TORCH

## COMPUTERS

---

To change memory locations copy the new values onto the line with the cursor and press **return**. If fewer than 16 hex numbers are entered then only the first few bytes of that block may be modified. The line will then be redisplayed and may be modified further if necessary. If '.' is entered on the line then the program will return to the TBUG prompt. To display the next 16 bytes of memory simply press **return** without modifying anything. Holding **return** down will allow large areas of memory to be cleared to 0.

**R** Register display and modify. After invoking this command the user is requested to enter the name of a register. Registers AF, BC, DE, HL, IX, IY, SP, and PC, may be modified by this command and except of IX, IY the first letter is sufficient to identify the register. Upper or lower case may be used. If register name X is entered the main register set and the alternative register set will be exchanged in the register buffer.

Pressing CR on its own will display the current value of all registers in the register buffer.

**S** Save file. This command allows areas of memory to be saved to disc. The user is prompted for filename and also for the first and last addresses to be saved. This means that programs need not be saved from location 0.

All filenames entered when using TORCHBUG (except the optional one on the command line) have the following format. Optional drive name A or B followed by '.', up to 8 character name, then optionally '.' and 3 character extension. If the extension is omitted it always defaults to '.COM'. The filename is taken to be as entered without any uppercasing. This last point allows access to any files consisting of printing characters only.

All hexadecimal numbers except those used by modify are of the following format: any number of characters consisting only of 0-9 and A-F or a-f, terminated by any other character. Numbers entered to modify a block of memory must contain exactly 2 digits.

The command line prompt in TORCHBUG is always TBUG>.

### EXAMPLE OF USE

A software package is available to allow customisation of WordStar for the CPN operating system. This consists of the files WSPAT.HEX (an Intel Hex file) and WSCUSTOM.COM. (For an explanation of WSCUSTOM.COM see the separate document 'Patching Wordstar for the Torch.') To produce a copy of WordStar for the Z80 Extension Processor the following procedure should be used:

Install WordStar without selecting any terminal but installing whichever printer drive is appropriate to your own installation. Assuming that the disc in drive A now contains the files mentioned above, TORCHBUG.COM and WordStar file (here called WS.COM) enter the following:

---

## TORCHBUG WS.COM

The TBUG> prompt should appear. WordStar is now in memory so the patch file should be loaded by pressing I. The prompt 'Filename:' will appear and WSPAT.HEX should be entered. N.B. If no extension is given then the file type will default to '.COM'. The hex file will load and be linked automatically into the existing program. The next stage is to save the patched version of WordStar by pressing S and then a new filename such as WS1.COM to save the patched WordStar. The start address should be given as 0100 and the end address as 3f00. The name under which WordStar is saved should not already exist on the disc as TORCHBUG will never delete an existing file.



### RS423.COM

RS423.COM allows data to be received through the serial port on the BBC machine. This facility allows the Z80 Extension Processor and BBC machine to communicate with other computers which have an RS423 interface.

This program is run by typing: RS423 'filename' **return**. The file 'filename' is opened, and the program waits for characters to come in on the RS423 port. The TORCH reads the characters as they come in and puts them in the file, using the RTS line to prevent the sender going too fast. If no characters are sent for more than 5 seconds, the program checks that the end of file has been reached, if so closes the file and returns to the command line.

The RS423 program running on a TORCH can be treated as a printer which will record output sent to it onto disc. Text files are therefore easily sent to the TORCH by simply printing them to it. Object files (.COM files) should be sent using a utility which sends binary characters to the serial port on the other computer.



---

## MUSIC

The TORCH music system consists of two programs, one of which plays music files, whilst the other is used in creating them.

**MUSIC:** The MUSIC program plays standard TORCH music files, the format of which is described below. TORCH music files usually have the extension .MUS.

**EXAMPLE:**   MUSIC B:CHARIOTS.MUS  
              MUSIC B:CHARIOTS

either of the above commands will play the music file name:

B:CHARIOTS.MUS

**NOTE:** The default extension is .MUS  
(For explanation of filename syntax see section on FILES AND FILENAMES.)

### MUSIC COMPRESSOR (MUSCMPX)

This program compresses a TORCH music manuscript file, (these normally have the extension .MSC) to give a TORCH music file, (normally with the extension .MUS) for use with music.

**EXAMPLE:**   To compress the file B:CHARIOTS.MSC into the file B:CHAR.MUS  
              enter the command:

MUSCMPX B:CHARIOTS B:CHAR

**NOTE:** The default extension is .MSC Both programs will give full instructions if run without specifying a filename.

### DESCRIPTION OF MUSIC FILES

A music file is an ordinary text file. With the exception of comments (see below), it is free format, with spaces and newlines being used to separate items. The program MUSCMPX will remove all unnecessary characters from a music file.

The file consists of a mixture of note descriptors and special items. These are described below. The end of file is detected automatically.

## NOTE DESCRIPTORS

A note descriptor is a sequence of four items. These are:

- (1) Channel Code.
- (2) Dynamic Mark.
- (3) Note name.
- (4) Length.

The channel code controls the channel and range that are used. There are three channels and two ranges, as shown in the table. A channel can only produce one note at a time, and the notes indicated in the file for a channel are automatically queued.

| Code | Channel | Range  |
|------|---------|--------|
| *    | 1       | Bass   |
| 1    | 1       | Normal |
| 2    | 2       | Normal |
| 3    | 3       | Normal |

The dynamic mark controls the volume of a note. There are 8 dynamic marks permitted:

|     |          |
|-----|----------|
| ff  | loudest  |
| f   | .        |
| mf  | .        |
| mp  | .        |
| p   | .        |
| pp  | .        |
| ppp | quietest |
| r   | silent   |

The note name is specified as an uppercase letter from A to G, followed by # or b if required to indicate sharp or flat (names such as Cb OR F # will be accepted). These names indicate the octave from middle C to the B above. Following the name by + or - signs will respectively raise or lower the note one octave. The range of notes available is:

Normal Range B-- to D++++  
Bass Range C---- to D#

If the dynamic mark is 'r', then the note name should also be so. The length is a number from 1 to 255 indicating the length of the note in units of 50 ms, i.e. 20 indicates one second.

Whilst the operating system will automatically queue notes on a channel, there is a limit to the amount that can be queued, and it is recommended that notes be kept in their 'real' order. The file can be conveniently formatted to aid this. For example, here is the start of 'Chariots of Fire'.

|         |    |         |    |         |    |
|---------|----|---------|----|---------|----|
| 1 f Db+ | 8  | 2 r r   | 32 | 3 r r   |    |
| 1 f Gb+ | 8  |         |    | 32      |    |
| 1 f Ab+ | 8  |         |    |         |    |
| 1 f Bb+ | 8  |         |    |         |    |
| 1 f Ab+ | 24 | 2 f Ab- | 4  | 3 f Db- | 48 |
|         |    | 2 f Db  | 4  |         |    |
|         |    | 2 f F   | 4  |         |    |
|         |    | 2 f Ab  | 4  |         |    |
|         |    | 2 f Db+ | 4  |         |    |
|         |    | 2 f Ab  | 4  |         |    |
| 1 f F+  | 24 | 2 f F   | 4  |         |    |
|         |    | 2 f Db  | 4  |         |    |
|         |    | 2 f Ab- | 4  |         |    |
|         |    | 2 f Db  | 4  |         |    |
|         |    | 2 f F   | 4  |         |    |
|         |    | 2 f Ab  | 4  |         |    |
| 1 f F+  | 16 | 2 f Db+ | 4  | 3 f Db- | 48 |
|         |    | 2 f Ab  | 4  |         |    |
|         |    | 2 f F   | 4  |         |    |
|         |    | 2 f Db  | 4  |         |    |
| 1 f Db+ | 8  | 2 f Ab- | 4  |         |    |
|         |    | 2 f Db  | 4  |         |    |
| 1 f Gb+ | 8  | 2 f Bb- | 4  |         |    |
|         |    | 2 f Db  | 4  |         |    |
| 1 f Ab+ | 8  | 2 f Gb  | 4  |         |    |
|         |    | 2 f Ab  | 4  |         |    |
| 1 f Bb+ | 8  | 2 f Bb  | 4  |         |    |
|         |    | 2 f Db+ | 4  |         |    |

The program will indicate if it detects any item in the wrong place, but will not attempt to rectify the fault.

### SPECIAL ITEMS

The following special items are available:

- (1) Comments
- (2) Repeat marks
- (3) Debug mode

Comments are indicated in one of two ways. Firstly, a semicolon indicates that all text until the end of the line containing the semicolon is a comment. Secondly, anything within braces {} is also a comment (except within the first kind of comment). Braces may not be nested, but may contain new lines. This second kind of a comment is known as a 'debug remark', as it is handled specially in debug mode.

Repeat marks are used to allow the repetition of part of a tune. up to ten independent repeat systems may be active at any one time. The number of the repeat system (0 to 9) is indicated by £ in the following description:

Four items are used to control repeats. These have the following effects when found whilst playing the tune:

- |     |  |
|-----|--|
| 7#  | All previous information about this repeat system is forgotten, the system is marked 'first time', and this location is noted.                                     |
| 8#  | If this system is marked 'first time', it is changed to 'second time', and a jump is made to just after the location of the noted 7#. Otherwise it is ignored.     |
| 10# | If this system is marked as 'second time', then a jump is made to just after the location of the first 11# beyond this point in the file. Otherwise it is ignored. |
| 11# | Ignored (see 10#).   |

The first item found for any repeat system MUST be a 7#.

Debug mode is provided to allow the notes being played to be viewed on the screen. It is entered whenever the item 121 is found whilst a tune is being played, and left whenever a 120 is found. Debug mode has two effects. Firstly, all debug remarks (see above) are displayed as they are reached during the playing. secondly, whenever a note is played, details about it are displayed on the screen.

N.B. It has been found that the extra processing involved in debug mode may have the effect of delaying some notes slightly (normally by about 1 length unit).



---

## MAPDISC

Each disc used on the TORCH system has an allocation map, which shows exactly which parts (sectors) of the disc are being used.

MAPDISC has two functions:

- 1) To allow the user to view the allocation map of the disc and look at the status of the sectors on the disc.
- 2) To allow the user to check that the map is correct, and if necessary attempt to correct it against the disc.

When MAPDISC is run it will first ask the user to enter the drive to be scanned and then display an allocation map of that disc showing the sectors that are allocated. For a full map of the disc, option f should be selected; this looks at the directory on the disc, and for each file it notes which sectors that file uses. It then sets up a full map on the screen of the disc. The symbols shown on a full map of the disc may be interpreted as follows:

A cell completely shaded in grey: this is a data sector that has been allocated and is used.

A completely empty cell: this is an unused and unallocated sector.

A cell with a grey horizontal line: this means there is an allocated L3 block in that sector.

A cell with a grey vertical line: this means there is an allocated L2 block in that sector.

A cell with a grey x: this is a cell that has been allocated but is not used. (garbage).

A cell with a grey u: this means that this sector is un-allocated but is in use.

A cell with a grey dot in the centre: this indicates that this cell has got multiple uses. i.e. Two files 'think' they own this sector.

As well as the full map of the disc (containing the above symbols) being displayed with the F option, there is also a full option menu displayed. All the options available are displayed on the menu.

The 12 options available on the full menu are as follows:

G, M, U, 2 and 3 display their specific sectors, (as described on the screen). Note that options 2 and 3 display blocks L2 and L3 respectively.

A Re-displays the main map of the disc.

N Allows scanning of another disc.

Q Quits the program.

W Writes the allocation map back to the disc (after it has been modified).

T Tidies the map, that is, it erases all garbage sectors from the disc, and fixes all un-allocated sectors.

NOTE: After using T, use the W option to put back the 'fixed' map. When the T option is selected, the message 'Tidied' is displayed on the screen.

H Displays a file handle. That is, it displays all the information associated with a particular file.

When the H option is selected, the number displayed following the H is the handle number of the file. i.e. Its position in the directory.

The following are additional symbols that may appear on the map after the H option (described above) has been selected:

A completely blue cell: this is a data sector for that file.

A cell with a blue horizontal line: this sector is an L3 block for that file.

A cell with a blue vertical line: this sector is an L2 block for that file.

A blue cell with a white U: this sector is part of the file but is unallocated on the map (irrespective of use).

A blue cell with a white dot in the middle: this is a sector used in that file and is also used by another 1 or more files.

NOTE: the above symbols only appear on the map when the H option (described above) is selected.

P This option will allocate the entire map as if used. Again to implement this on the disc, the W (write) option described above, should be selected. Note that when the p option is selected, the message 'Protected' is displayed on the screen.

---

## TYPE AHEAD

If MAPDISC is in the middle of drawing a map of a disc, and a key is pressed, which is a valid option, (as described above), then MAPDISC will break off from drawing the map of the disc, and execute the option selected. A message is displayed on the screen telling the user if MAPDISC is drawing a map, or if it is interrupted whilst doing so. Note that when a particular option is selected, a message may be displayed on the top left hand side of the screen. This message is: 'working' or 'drawing map'.



## POKEDISC

POKEDISC allows the user to scan a disc and change the contents of the sectors directly, without using the normal file structure.

When using POKEDISC, the typeahead facility is suppressed, in that, it will only 'remember' the first letter that is typed.

At any time POKEDISC is displaying 1 sector of the disc. this sector may be displayed in one of four modes, each of which is selected by its relevant letter key. These four modes are as follows:

D This selects the DIR mode which displays the 16 directory entries.

K This mode displays the sector as a set of 256 characters. If the character is not a character that may be on the screen, then a hexadecimal code will be displayed instead. For instance, a carriage return would be displayed as 0D. A yellow character indicates that the most significant bit of the character is set. A space with the most significant bit of it's byte set would be represented by the symbol of a square bracket on its side.

H This mode sidplays the entire sector as 256 pairs of hexadecimal digits.

M or N Either of these letter keys will select mixed mode, this displays one eighth of the sector at a time. Thie particular part of the sector being displayed is shown at the top right hand of the screen, and also down the left hand side. Example BYTES 00 to 1F. The 32 bytes being displayed are treated as 16 words, (each of two bytes) each word is displayed as sixteen bits, 4 hex digits, 2 characters, and possibly a sector address if valid. Note: because of the way the Z80 processor handles words, the least significant byte is the first character.

## MOVING AROUND THE DISC

A number of keys may be used to move around the disc. The letter keys A and B are used to select disc drive A OR B. If a hard disc is used, then B will cause the user to be asked to enter the surface required. Some of the keys listed below will have different functions depending on which issue of POKEDISC is being used.

The 0 key selects: cylinder 0, side 0, sector 0 (0:0:0)

The S key moves to the other side of the disc.

The R key moves to the other side of the disc.

The C key moves up 1 track.

The E key moves up 1 track.

The X key moves up 1 sector.

The E and X keys may be used to move to another track if necessary.



---

When in mixed mode (i.e. when option M or N is selected), the W key is used to move down 2 Bytes.

The J key (followed by 4 digits) is used to move to a particular sector, whose location is determined by the 4 digits that follow the J. The J followed by the cylinder number and a colon, the side number and a colon, and the sector number, are entered, to 'jump' to a specific part of the disc. The delete key may also be used.

EXAMPLE: J5:1:6

This would move the user to cylinder 5, side 1, sector 6.

The QUIT key, which is the first red function key at the top left of the keyboard, (key f0), is used to quit a particular option, or to quit the program entirely.

The COPY key, will display a further menu with more options, all of which are selected by using the appropriate control code as described below.

Control R: This option allows the user to define specific sector addresses to the red function keys along the top of the keyboard. Control R followed by the QUIT key, (red function key f0) will bring the user back to the menu.

Control T: save the current sector in a special buffer.

Control S: This option is used to copy the contents of the buffer (referred to above) back to the screen.

Control U: Any editing that has just taken place will be cancelled and the sector re-read from the disc.

Control B: This option allows the user to actually start editing the sector. once in this mode, none of the above options will be operational except the cursor control keys.

Control E: This finishes the editing operation, and returns the user to the normal mode of operation for POKEDISC.

Control L: This moves the cursor the the start of the current line.

**return:** This moves the cursor to the start of the next line.

---

# TORCH

## COMPUTERS

---

NOTE: When editing, it is possible to change anything in the sector that is displayed on the screen by typing any valid character at that point. Whilst editing a character, the escape key may be used to enter a hex code. Also the " key and the ' key may be used to set, and clear, respectively, the most significant bit of a character.

The valid keys for the 'entry' field of a directory display are 2, 3, D, and N. Those for attributes of a file are: R, W, S, U, E, and G.

The program will interrogate the user before it writes back to the disc any editing that may have taken place on a particular sector.

---

## **RDACORN AND WRTACORN**

These utility programs are provided to enable the user to convert files from Acorn DFS format to TORCH CPN format and vice versa. There may be numerous occasions when manipulation of files between the two operating systems becomes a useful facility. A typical example of this is where a test file prepared in BBC BASIC could be written into a TORCH CPN file and edited using a word processor such as WordStar. If the user also has access to a TORCH microcomputer then BBC files could be written into CPN and transmitted to a remote TORCH using the communications package TORCH MAILPLUS. (Once received at the remote end the file could be converted back into BBC BASIC format.)

### **WRTACORN**

WRTACORN is used to convert files from TORCH CPN disc format to Acorn DFS disc format. When run, WRTACORN asks the user to enter the name of the CPN file, and the name to be assigned to it when it is written into Acorn format. After a file has been written into Acorn format from CPN it is stored on the disc as a text file, therefore the DFS command \*EXEC should be used to load and run it. (For information on DFS commands see Appendix A.)

NOTE: It is essential to put the Acorn DFS format disc in disc drive A, and the TORCH CPN format disc in disc drive B. If this procedure is not followed then one or both of the discs may be corrupted.

### **RDACORN**

RDACORN is used to convert files from Acorn DFS format to TORCH CPN disc format. Before the Acorn file is converted to CPN format, it should first be converted to a text file using the \*SPOOL command. (See Appendix A for information on DFS commands.) RDACORN is used in the same way as WRTACORN. i.e. by entering the name of the file to be converted, and the new name to be assigned after being written into CPN format.

NOTE: As in the case of WRTACORN the Acorn DFS format disc must be placed in drive A, and the CPN format disc in drive B or the discs may be corrupted.



### FIND

Find is a utility program used to search for files on a disc or a group of discs. It is primarily intended for hard disc surfaces although it can be used with floppy discs just as well. To run the program type:

FIND drivelist:filename **return**

Where:

drivelist Is any number of disc surfaces in the range A to P.

filename Is any valid filename which may include ambiguous characters.

NOTE: If no filename is given, the program will list all the files on the disc surface/s specified.

### EXAMPLES

FIND BCD: \*.COM **return**

This example would list the files with an extension .COM which are stored on the disc surfaces B, C and D.

FIND KL: **return**

This example would list all the files on the disc surfaces K and L (These will probably belong to another computer on a network).



---

## COPIER

This utility program is intended, primarily, to copy a 40 track disc using 80 track disc drives. It can also be used to copy 80 to 80 track and 40 to 40 track discs. The program is menu driven and very easy to use. To run COPIER type:

COPIER: **return**

and the program will display the main menu which includes a list of the different options available. These are:

### OPTIONS

- |      |   |
|------|---|
| C    | Change the disc type from single to double sided or vice-versa.   |
| S    | This option is to change the source disc from A to B when using TORCH format discs and to change the source disc from 0 to 1, 2 or 3 when using ACORN format discs.     |
| D    | This is the same as the S option except that it changes the destination disc drive.   |
| F    | The F option allows you to format the destination disc before the source disc is copied to it.  |
| V    | This option lets you choose whether to have the data verified after it has been copied to the disc. If the verify option is chosen, the copying may be a little slower. |
| T    | This option allows you to toggle between ACORN and TORCH disc format.   |
| Z, X | The Z and X options change the source and destination discs to either 40 or 80 track respectively.  |

To copy both sides of a disc automatically, you should choose TORCH format discs. Note that this will not change the disc size on ACORN format discs. (To do this, use the RDACORN and WRTACORN programs described below.)

NOTE: If you require more detailed information on the differences between ACORN and TORCH disc formats, study the BBC User Guide and consult your dealer if necessary.

### OPERATION

When the program is first used, the options will be at their default settings. It is possible to change any of these options by pressing the relevant key on the keyboard. Once the options have been set, B should be pressed to begin. The screen will then clear and the program will give another prompt before the copying starts. This prompt will be:

Insert discs and Press Any Key  
to continue

NOTE: When this prompt appears; or at any other time in the program it is possible to press the escape key to return to the main menu.

## SETTIME

This utility program is used to set the time of the TORCH clock. To read the time, type:

TIME **return**

the computer will display the time in the following format:

The time is 12:42:36

To set the time, type:

SETTIME HH:MM:SS **return**

where HH are the hours.

MM are the minutes.

SS are the seconds.

The time will be set when the space bar is pressed. If an invalid time is entered, the program will reply with one of the following two error messages.

This is not a valid time

or

Format should be HH:MM:SS

NOTE: If you type

SETTIME **return**

the time stored in your TORCH will be displayed.

---

## **TORCH DISC SYSTEM**

Data and programs used on the TORCH system are stored on mini floppy discs or hard discs. Floppy discs are constructed from a thin piece of plastic sheet, treated with a magnetic compound which allows the storage and retrieval of data to and from the disc. When the disc is placed in the DISC DRIVE it is spun at high speed, the heads of the disc drive read and write data to and from the disc. Great care should be taken when handling discs as they are easily damaged.

The following points should be remembered when handling discs:

1. Always store the discs in the protective envelopes provided when not in use.
2. Avoid bending the discs.
3. Insert into disc drives carefully.
4. Don't touch exposed area of disc.
5. Don't expose disc to extreme temperatures or magnetic fields.
6. Avoid getting smoke or other contaminants on the surface of the disc.
7. Don't switch the power to the BBC Machine on or off without first removing the disc(s).

To load a program from the system disc, the disc should be inserted into one of the Disc Drives. When using only 1 disc drive, the user will normally use drive A. This is because the CPN operating system automatically 'defaults' to drive A on 'power on.' Disc Drive A is referred to as the logged disc drive: this may be changed if required.



### LOADING A FILE FROM DISC

To load a file from disc, place the disc with your program on, in disc drive A, carefully closing the door of the disc drive, ensuring that the disc is fully inserted into the drive. To load your program type the filename of the program and press **return**. If you want to find out the names of the files (programs) stored on a disc, type DIR and then press **return**. The VDU will then display the contents of the disc. For details of other disc commands, refer to the section on COMMANDS. For details of the Acorn DFS commands used in BBC BASIC see APPENDIX A.

### SAVING A FILE ON DISC

Before a file can be saved on disc the disc has to be 'formatted'. This process prepares the disc to receive data. The disc is divided into 80 tracks, track 0 being at the outside edge of the disc, track 79 being at the inside edge of the disc. The tracks on both sides of the disc are divided into 10 sectors, and each sector can store 256 bytes (characters) of data. It is in these sectors that the files (or data) are stored. The number of sectors required to store any given file will depend entirely on the length of the file.

To format a disc, place it in disc drive A, type FORMAT and press **return**. The computer will now instruct you via the VDU to tell it which disc drive you are using, type A or B and the computer will now format the disc as described above. During this formatting process the operating system automatically checks that no errors have occurred. Should any errors occur during this formatting process the computer will display on the VDU an error message, so you should then check that you have placed a disc in the correct drive, i.e. the one designated above, or try another disc in case you have a faulty or corrupted disc in the machine. For an explanation of error messages see section on DISC ERRORS.

NOTE: A disc may be formatted in CPN and immediately used in BBC BASIC mode for loading and saving BASIC programs.

### IMPORTANT

It should be noted that the formatting process completely clears the disc of any files that are on it. Care should be taken not to put the wrong disc into the machine. The FORMAT command is used for erasing data off old discs and for formatting new discs prior to saving data on them.

To prevent losing important programs, (and a lot of time!) either by loss or damage to the disc, or by accidental erasure of data from the disc, it is recommended that 'backup' copies are made of all important discs. Discs may also be Write Protected to prevent data accidentally being written over an existing file. All TORCH system and utility programs should be backed up, and write protected.



---

## MAKING BACKUP COPIES OF DISCS

A useful exercise in making backup copies, would be to copy the disc that came with your Z80 Extension Processor (SYS 1.7). It is good practice also, to make backup copies of the discs you have been using at the end of each day.

### TO DUPLICATE A DISC

1. Put the disc to be copied into drive B. This is called the 'source' disc.
2. Put the blank disc into drive A. This is called the 'destination' disc.
3. Type DUP and press **return**.

The screen will then display the following instructions:

'press RETURN to continue or any other key to abandon'

When you press **return** the operating system will format disc A and duplicate the contents of disc B onto disc A.

Whilst it is doing this it continuously checks for errors; should an error occur during this process an error message will be displayed on the screen, such as:

Disc A Track 001D Sector 00 Error 0E

In the event of this happening, try taking the destination disc out of the drive and putting it back in again; it could be that the disc was not aligned in the drive properly. Repeat the above procedure to copy the contents of disc B onto disc A. For an explanation of disc errors see section entitled DISC ERRORS.

### WRITE PROTECTING A DISC

To prevent a disc having data written onto it accidentally, or its data accidentally wiped off, a square of adhesive paper should be placed over the small notch or cut out on the side of the disc. This cutout is known as the WRITE PROTECT NOTCH. Adhesive patches to cover the write protect notch are supplied with the blank discs.

## DISC ERRORS

When the computer is reading or writing data to or from the disc, it is continuously checking for errors.

If an error occurs during a read operation, the TORCH operating system attempts to re-read the track up to 5 times.

If an error occurs during a write operation, the operating system will re-format the track, and then attempt to re-write the data up to 5 times.

If no error has occurred during a write operation, the data that has just been written onto the disc is verified for integrity and if any data corruption has occurred, the track will again be re-written up to 5 times. Every second attempt at re-writing the track, the operating system will re-format the track. If after 5 re-tries of either re-reading or re-writing, there is still an error, then an error signal such as the one below will be printed on the screen:

DISC A Track 001D Sector 00 Error 0E

This tells the user the disc drive that failed, the track that failed, and the hex error code, in that order.

NOTE: The track number that is given in this error message is a hexadecimal value, and is actually twice the true track number, plus the side number. In the case of the example above, this would actually be Track 14 Side 01.

Normally all of the above operations of re-reading and re-writing are 'invisible' to the user until, after the fifth attempt at re-reading or re-writing, it has abandoned the operation and printed the error message.

To monitor the above operations as they take place, and determine where and when an error is occurring, type in the following before performing a read or write operation:

[SYSTEM 16

PRESS **return** and continue as normal. If an error occurs now, the operating system will immediately print an error message such as the one below:

DISC A track 08 re-reading

This time the error message is printed to the screen as soon as an error occurs, and informs the user each time that a re-reading re-writing, or re-formatting operation takes place. As before after 5 attempts are re-reading, and re-writing, then the operation will be abandoned. In which case the user will have to try again.

---

---

A similar instruction to [SYSTEM 16 is [SYSTEM 64; in this case the operating system will continue to attempt to re-read, re-write, or re-format data even after 5 attempts have failed. This process will continue unless the following error message is displayed:

DISC A track 08 unrecoverable error 18

This error message only occurs after 5 attempts at re-formatting a track have failed. In this situation, (which is very rare) there is either a serious fault with the disc, or with the disc drive itself, and the user should try another disc before consulting a dealer.

NOTE: To set more than one system function at a time add the two function numbers together. e.g. to have both [SYSTEM 16 and [SYSTEM 64 set, type [SYSTEM 80. Any form of reset will remove these functions (e.g. CTRL C). [SYSTEM 0 may also be used to remove these functions.





---

## PREPARING COMMAND FILES

A command file contains a set of characters which can be presented to the operating system as though they were typed at the keyboard. Command files are prepared using the INPUT command, or a wordprocessor such as PERFECT WRITER.

EXAMPLE: Insert a blank formatted disc into drive A. Enter the following lines of text, each terminated by **return**.

```
INPUT BEGIN.SUB
F 6
B 4
HELP
```

Now enter a CTRL-Z (press control and Z together) followed by **return**. After a slight delay, the prompt 0A> will appear. The file BEGIN.SUB now exists on the disc. check this by using the DIR command or the TYPE command.

To enter the contents of BEGIN.SUB as a sequence of commands, use the command:

```
C BEGIN.SUB
```

BEGIN.SUB should be loaded from disc, and the commands F 6 (Foreground cyan), B 4 (Background blue), and HELP should be executed. The operating system will then be ready for commands from the keyboard.

When using command files, facilities are available for argument substitution, and inserting comments into the file which would then be displayed on the screen when the file is executed. Where a dollar sign (\$) followed by a number appears as part of the text in a command file, the argument corresponding to that number will be substituted in the text of the command file. Where dollar zero (\$0) appears, the name of the command file is substituted. The filename may be followed by up to nine arguments, which are separated by spaces and terminated by the lines end.

Special characters may be inserted into the argument using backslash \ followed by a letter:

|                   |                                       |
|-------------------|---------------------------------------|
| \n :newline       | (moves to a new line).                |
| \s :space         | (inserts a space into that argument). |
| \0 :null argument | (that argument is ignored).           |

EXAMPLE: C TEST XXX TXT RU \sN

In this example TEST is the filename, XXX is argument 1, TXT is argument 2, and RU N is argument 3. Note that to incorporate a space into the argument RU N, \s was used when entering the file, as an actual space may not be used within an argument, but is used to separate arguments.

To copy files of a specified name from drive B to drive A, for example for making backups, enter the file BACKUP.SUB containing:

```
; Backing up $1  
DEL A:$1 [u  
COPY B:$1 TO A:  
VDU 7  
followed by control Z and return.
```

The command file C BACKUP\*.DOC will display the words 'Backing up \*.DOC', delete all files on disc A with the .DOC extension to disc A, and then sound a bell on the speaker, to tell the user that the backing up operation has been completed. (The old version of the files were first deleted from disc a and then replaced with the new version from disc B). Using the above procedures the user may write their own command files for various purposes.



---

# APPENDIX A

## DFS and UTILS COMMANDS

This appendix is a guide to the ACORN DFS and UTILS commands that are available on your BBC micro with the TORCH Z80 Disc Pack.

Listed below is each command and a brief outline of its function.

NOTE: The DFS command parameters given in the examples below have the following meanings:

<fsp> ..... file specification.  
<afsp> ..... ambiguous file specification.  
<src> ..... source.  
<drv> ..... drive.  
<dest> ..... destination.  
<dir> ..... directory.

NOTE: When operating the BBC computer with your Disc Pack in BBC Basic mode, the Disc System operates as a Dual Drive double sides system. This means that the drive number specified will be in the range of 0 to 3. (See below for an explanation of File Specifications.) The drive number given must be preceded by a colon (:). The drives (surfaces) are numbered as shown below:

Drive 0  
Drive 2

Drive 1  
Drive 3

EXAMPLE: To display a catalogue of the files on the left hand drive, surface 2 enter:

\*CAT: 2 **return**

If a drive (surface) number is omitted the current drive is assumed. For an in-depth explanation of the Disc filing System refer to the Acorn manual.

## FILE SPECIFICATION

A file specification consists of:

:drv.dir.filename

EXAMPLE: \*EXEC:1.C.MYTORCH **return**

will EXEC the file MYTORCH on directory C of drive 1. For further information on filename syntax refer to the BBC user guide.

## COMMANDS

**\*ACCESS** <afsp> (L) **return**

Locks or unlocks a file.

EXAMPLE: \*ACCESS MYTORCH L locks the file MYTORCH

\*ACCESS MYTORCH unlocks it.

A locked file cannot be saved to, deleted, wiped, renamed, or destroyed. Although it may be formatted.

**\*BUILD** <fsp> **return**

Creates an ASCII (text) file directly from the keyboard.

EXAMPLE: \*BUILD TORTXT displays line numbers which allow the user to enter data as text. After the ESCAPE key has been pressed the file TORTXT exists on disc and may be accessed via the DFS commands \*EXEC \*LIST or \*TYPE.

**\*CAT** <drv> **return**

Gives a catalogue of a disc on the screen.

EXAMPLE: \*CAT 1 gives a catalogue of drive 1

\*CAT gives a catalogue of the current drive.

**\*COMPACT** <drv> **return**

Tidies the disc in the named drive so that all free space is moved to the end.

---

EXAMPLE: \*COMPACT 1 will tidy the disc in drive 1.

**\*COPY** <src drv><dest drv><afsp> **return**

Allows a file to be copied from one drive to another or from one drive to itself.

EXAMPLE: \*COPY 0 1 SPATS copies SPATS from a source disc on drive 0 to a destination disc on drive 1.

**\*DELETE** <fsp> **return**

Directly removes an unlocked file from the catalogue of a disc. Note that a deleted file cannot be revived.

EXAMPLE: \*DELETE PHILBY **return** will remove PHILBY from the current disc in use.

**\*DESTROY** <afsp> **return**

Removes all unlocked files specified.

The \*ENABLE command must be entered immediately before \*DESTROY or it will not work.

EXAMPLE: \*ENABLE

\*DESTROY \*.\* will remove all files on the disc.

Note that destroyed files cannot be revived.

**\*DIR** <letter> **return**

changes the current directory name.

EXAMPLE: \*DIR G sets the current directory to G. Any saved file will be prefixed with the letter G.

**\*DRIVE** <number> **return**

Selects the current drive in use.

EXAMPLE: \*DRIVE 1 sets the current drive to 1 so any command not specifying a drive will log onto drive 1.

**\*DUMP** <fsp> **return**

Produces a listing in hex of the file on the screen.



---

# TORCH

## COMPUTERS

---

EXAMPLE: \*DUMP SPATS will produce a hex listing of the file SPATS CTRL-N will select paging mode.

**\*ENABLE return**

Allows the irreversible command \*DESTROY to be used. It is a safety lock.

**\*INFO <afsp> return**

Displays the following information on the specified file(s): directory, filename, access, load, address, execution address, length in bytes, and start sector.

EXAMPLE: \*INFO \*.\* displays the above information on all the files on the current disc.

**\*LIB <drv> <dir> return**

Sets the library to the specified drive and directory.

EXAMPLE: \*LIB: 1.C sets the library to drive 1, directory C.

**\*LIST <fsp> return**

Displays an ASCII (text) file created with the \*BUILD command.

EXAMPLE: \*LIST TORTXT will display the contents of the file TORTXT.

**\*RENAME <old fsp><new fsp> return**

Changes the name of a file.

EXAMPLE: \*RENAME ALPHA BETA will rename the file ALPHA as BETA.

**\*TITLE <disc name> return**

Retitles the disc with a new name. (Up to 12 characters).

EXAMPLE: \*TITLE "TORCH DISC" will display the words "TORCH DISC" on future \*CATs of the discs.

**\*TYPE <fsp> return**

Displays a text file on the screen without line numbers.

---

EXAMPLE: \*TYPE TXT will display the contents of the file TXT as a text file. CTRL-N may be used to select paging mode.

**\*WIPE <fsp> return**

Removes specified files from a catalogue after asking permission for each file. It is only possible to remove unlocked files.

EXAMPLE: \*WIPE \*.C\* wipes all files beginning with C.

---

# TORCH

COMPUTERS

---



---

# **Glossary**

The following is a list of some of the terms used in the manual and other TORCH manuals.

## **APPLICATION PROGRAM**

A program written to carry out a specific task such as word processing.

## **BYTE**

A group of digits called bits which the computer treats as one unit.

## **CHARACTER**

Any letter, number or symbol on the keyboard.

## **CURSOR**

The flashing rectangle which shows the current typing position on the screen.

## **DATA**

Information that can be processed by a computer.

## **DISC, DISK, DISKETTE**

A floppy plastic disc used to store data.

## **DISC DRIVE**

The unit into which you insert the discs.

## **FILE**

A unit of storage on a disc which is named and can be accessed.

## **FORMATTING**

Preparing a disc to receive data.

## **HARDWARE**

Physical equipment which makes up a computer.

### **MEMORY**

The part of the computer that can store data.

### **MENU**

A list of the activities a program can perform.

### **PROGRAM**

A set of logically ordered instructions which tell the computer how to perform certain tasks.

### **RAM**

Random Access Memory, also known as read/write memory.

### **ROM**

Read Only Memory.

### **SOFTWARE**

The instructions used with computer hardware, such as application programs, languages and operating systems.

# Index

|                             |           |                  |        |
|-----------------------------|-----------|------------------|--------|
| Allocation map              | 49        | Pokedisc         | 52     |
| Ambiguous filenames         | 35        | Prestel          | 23     |
| ASCII                       | 34,38     | Print command    | 25     |
|                             |           | Protect command  | 26     |
| BBC BASIC                   | 2,10      |                  |        |
| Background command          | 9         | User command     | 31     |
| BASIC command               | 10        | Utility programs | 37     |
| Break key                   | 23        |                  |        |
| British Telecom             | 23        |                  |        |
|                             |           | Rename command   | 27     |
| Copier                      | 57        | Ribbon cable     | 4      |
| CPN operating system        | 5         |                  |        |
| Character set               | 38        | Sectors          | 17     |
| Command command             | 11        | Settime          | 58     |
| Command files               | 36, 65    | Submit files     | 36     |
| Copy command                | 12        | System disc      | 37     |
|                             |           | System functions | 62, 63 |
| Delete command              | 13        |                  |        |
| Dir command                 | 14        | TUBE interface   | 4      |
| Disc backup                 | 61        | Time command     | 29     |
| DISC ERRORS                 | 62        | Torchbug         | 41     |
| Disc filing system (ROM)    | 2, 3      | Troubleshooting  | 5      |
| Disc filing system commands | 67        | Type command     | 30     |
| Disc interface              | 1, 2      |                  |        |
| Disc system                 | 59        | VDU command      | 33     |
| Duplicate command           | 15        | View command     | 32     |
|                             |           |                  |        |
| Error message               | 62,63Exec | Z80 Eurocard     | 4      |
|                             |           |                  |        |
| Find                        | 56        |                  |        |
| Files and Filenames         | 35        |                  |        |
| Font                        | 38        |                  |        |
| Foreground command          | 16        |                  |        |
| Format command              | 17        |                  |        |
| Function codes              | 33        |                  |        |
| Function keys               | 20        |                  |        |
|                             |           |                  |        |
| Help command                | 18        |                  |        |
|                             |           |                  |        |
| Input command               | 19        |                  |        |
|                             |           |                  |        |
| Key command                 | 20        |                  |        |
|                             |           |                  |        |
| Links (PCB)                 | 3         |                  |        |
|                             |           |                  |        |
| Machine operating system    | 1, 2, 3   |                  |        |
| Mapdisc                     | 49        |                  |        |
| MCP                         | 3, 22     |                  |        |
| Mode command                | 23        |                  |        |
| Music                       | 45        |                  |        |



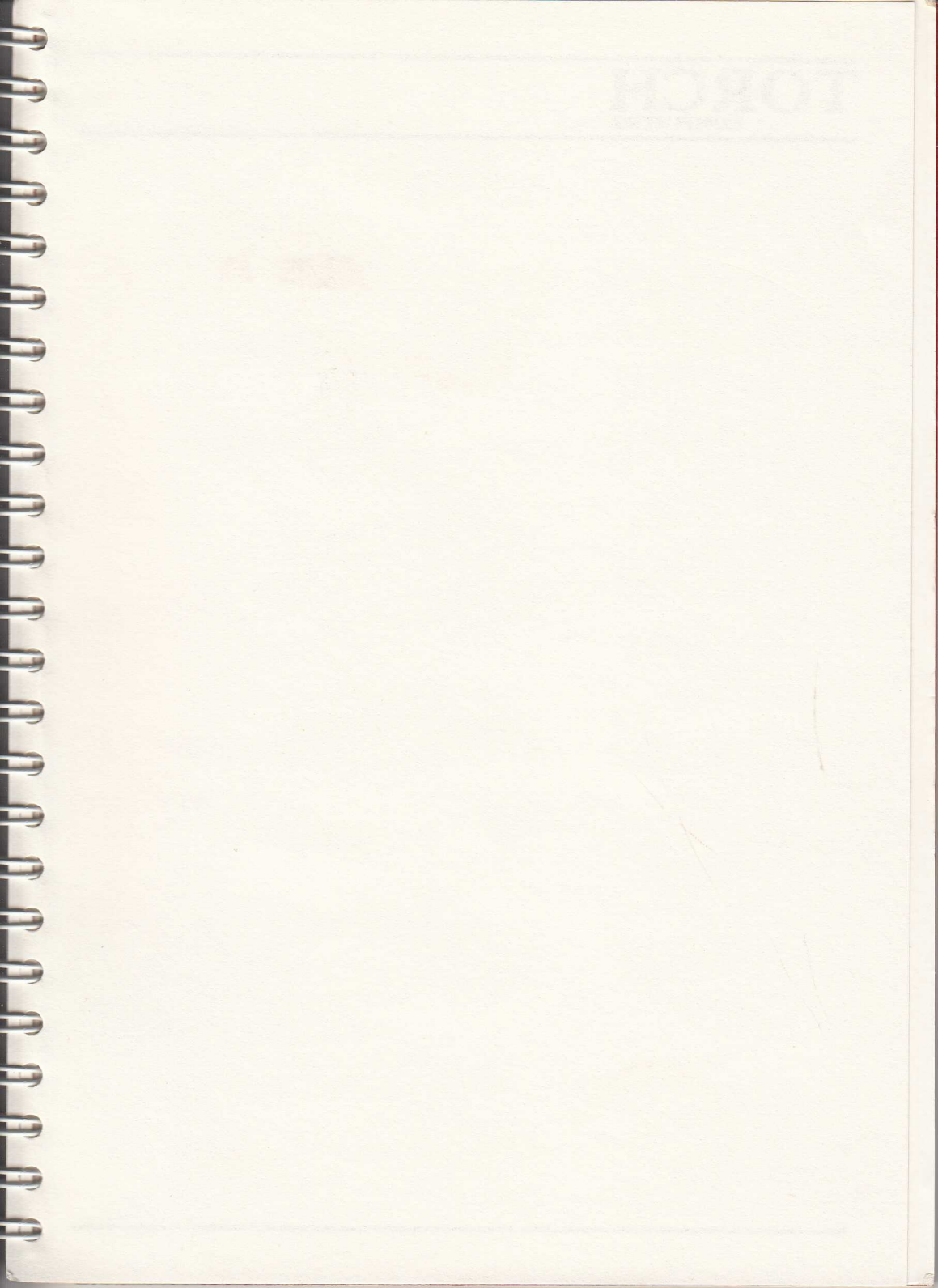
---

# TORCH

COMPUTERS

---







---

# TORCH

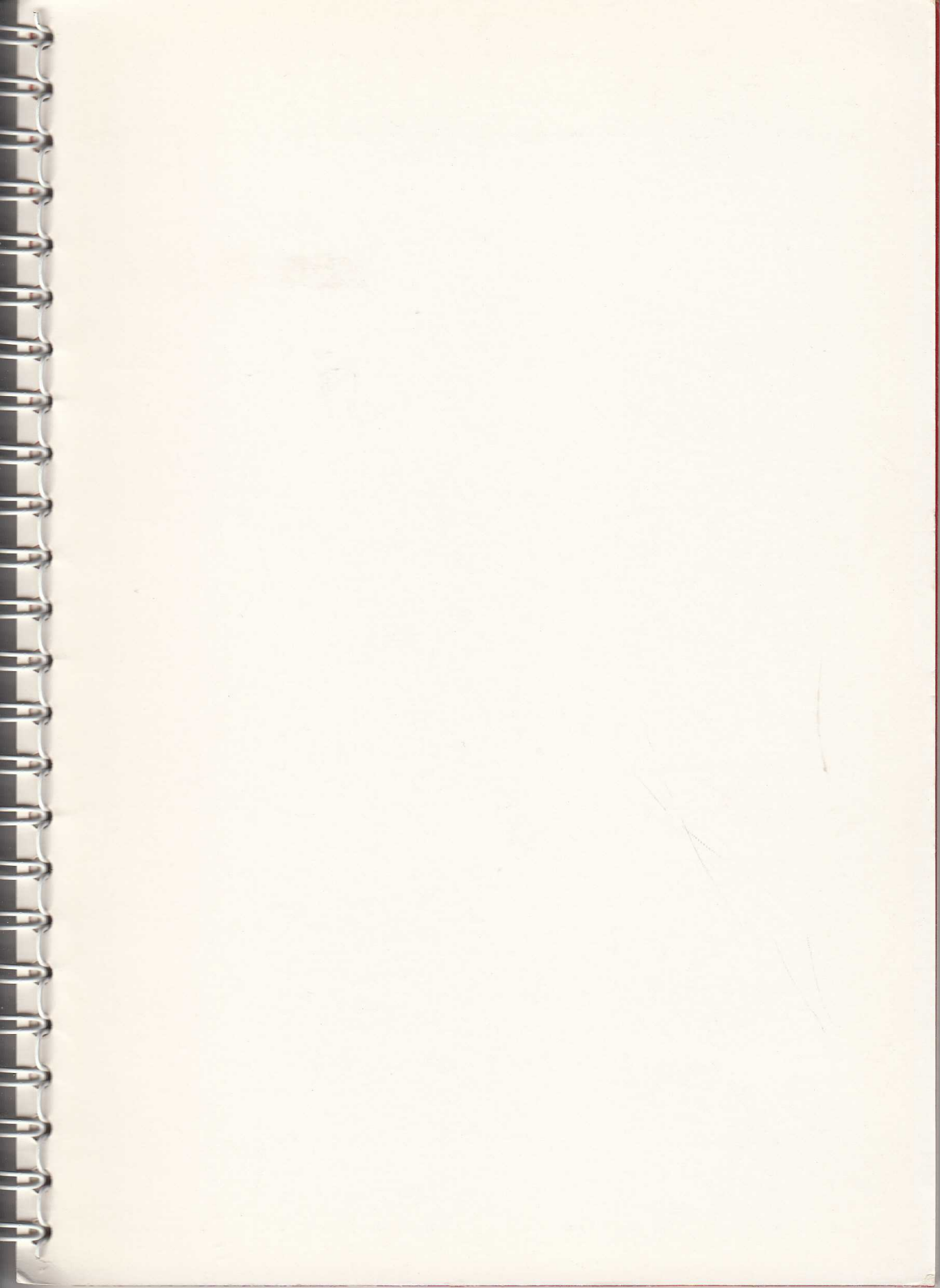
COMPUTERS

---



---





**TORCH**  
COMPUTERS

# **ZEP 100 User Guide**