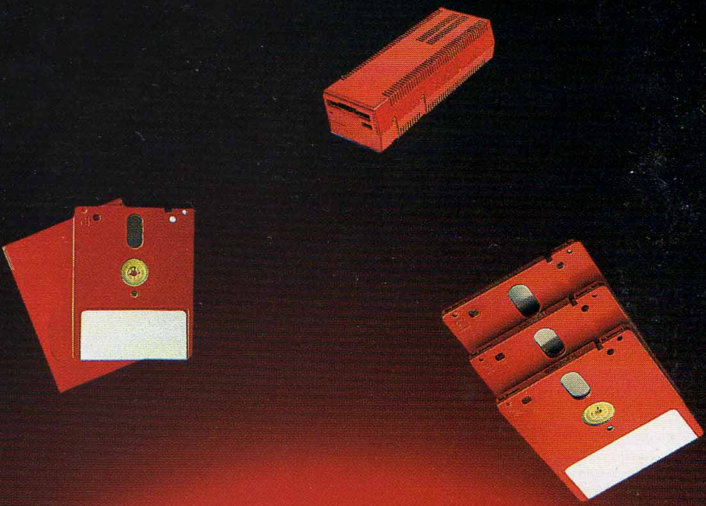
DISK USER'S UTILITIES



SUPER POWER

MANUAL

ROM BASED SOFTWARE FOR YOUR AMSTRAD
FOR MODELS CPC 464 AND CPC 664

CONTENTS

Starting DiscPower

To start DiscPower type |DP (| is obtained by pressing SHIFT/@). This will take you into the DiscPower Main Menu, from which you can select any of the utilities available.

Keys

Within this manual we refer to various keys and combinations of keys. CTRL/E means hold down the CTRL key (to right of the space bar) and press E. SHIFT/ESC means hold down either SHIFT key, and press the ESC key (top left of keyboard). The CURSOR keys (in a cluster above the numeric keypad), are referred to by CL, CR, CD and CU (for left, right, down and up, respectively). In all areas of computing remember that 1, l and I are all different, as are 0, o and O, unless the programmer specifically arranges for some of them to be treated in the same way, e.g. throughout this program commands may be in upper or lower case.

ESC is used to "back up" through a string of entries or to the last menu. E.g. in the SEARCH routine you are prompted for a string and then two addresses. Pressing ESC on the second address takes you to the first, another ESC takes you to the string prompt, and a third takes you back to the menu.

BACKGROUND

This is a foreground ROM designed to access data and programs on the disc directly. In order to make use of this facility, it is necessary to understand something of the way CP/M and AMSDOS store data on the disc. Each disc has 40 "tracks" each composed of 9 "sectors". Each track consists of a circular path around the disc, whilst a sector is part of that path. Each sector is capable of storing 512 bytes of data, giving the disc 512 x 9 x 40 = 184320 bytes = 180 Kbytes. The information about where a particular program is stored, is kept in a "directory". This has one or more entries for each file on the disc, and contains information about it. There are 32 bytes for each directory entry. The first is E5 for a deleted file or the "user number" otherwise. Bytes 2 - 9 are the filename (padded with spaces if necessary); 10 - 12 are the extension. 13 - 16 have special significance to the operating system. The next 16 bytes are the "allocation numbers", and show whereabouts on the disc the file is stored. The first file to be put on a disc is stored in allocation unit (AU) 02. This consists of the first 2 available sectors, giving 1K of space. If the file is longer than 1K unit 03 is used, and so on. Thus for a 4 1/2 Kbyte file, units 02, 03, 04, 05 and 06 would be allocated, though 06 would be only half filled.

Since there are 16 "slots" for allocation units, the maximum size of file for a single directory entry is 16K. Should a file exceed this size another directory entry must be created. Its first 12 bytes will be as before, but byte 13 will contain 01. When a directory command is given (i.e. DIR from CP/M or CAT from AMSDOS) this second entry will be suppressed. If the file is longer than 32K there will be a third entry, in which byte 13 is 02, and so on. Byte 16 in the directory entry says how many "records" long an entry is. In this sense a CP/M record is 128 bytes (the original size of a sector on the disc format for which CP/M was designed) and so in our example of 4 1/2 K above, this byte would be 24. Note that all these values are in hexadecimal notation (hex), so that 24 means 2 x 16 + 4 or 36 in decimal.

When a file is deleted the first byte of its directory entry is set to E5 (which is how the whole disc is set when it is formatted). The entry is otherwise unchanged, but the operating system will now know that this entry is available for use. If you now save another file then CP/M will look for the first free entry in the directory and put the directory data there, as well as putting the contents of the file in those allocation units which are the first available. These may well be the ones you have just released by deleting the first file.

This system allows considerable scope for restoring files which have been erased accidentally. Suppose you have saved files A.TXT, B.TXT, C.TXT, D.TXT, in that order, on your disc. You wish to erase B, C and D and so you use the CP/M command ERA *.TXT. This will erase all files with the TXT extension, and you realise after having done it that this includes A.TXT, which is a very important document. Using DiscPower you read the directory. All the entries are just as they were before the deletion, except that the first byte of each is now E5. If you now edit (for more information on how to edit the display see the EDIT command) this byte to be 00 for A. TXT, then write this back to the disc, you can leave DiscPower and should find your file back in the normal CP/M directory.

Suppose, however, that you save E.TXT on the disc before realising that you still need A.TXT, and, further, that A is 10K long whilst E is 1K. If you look at the directory now, you will see that E has replaced A as the first file. All is not lost!! First save E on another disc. Now edit the directory entry for E. Change the name to A (byte 3), the length from 08 to 50 (decimal 8 x 128 byte to decimal 80 x 128 bytes) and the allocation units at bytes 17 to 25 from 00 to 03 to 0B. this should conform with the old file entry. If you now write this to the disc you will be able to recover most of your file - the first 1K of it will be the contents of E, as this will have been written to AU 02 over the top of A, but the rest of the file should be intact. If it was D.TXT that you wished to recover, it would have been possible to do so completely, provided you had not saved more than 3 files, and that their total length had not exceded that of the original A, B and C. Even with more than 3 files saved it would be possible to salvage D, provided that the same AUs had not been re-used. To do this you would have to reconstruct the directory entry from scratch, including searching the disc to find out what AUs were used by D. With a text file with at least some known contents it would be possible to do this using the SEARCH MEMORY command in Disc-Power.

COMMANDS

1 DIRECTORY

This command will look to see whether this is a CP/M or an AMSDOS disc and load the directory into the buffer. You are then presented with the EDIT screen, headed by "Directory". You may now edit the directory entries in the buffer, saving your changes back to the disc when you have finished.

2 READ DISC SECTOR

This option allows loading of a particular track and sector from the disc to the edit buffer. See the EDIT MEMORY section for the controls then available.

You should note that whilst AMSDOS-CP/M sectors are 512 bytes long, with 9 sectors per track, "interleaving" of these tracks means that you are normally only able to load two tracks at a time, rather than one. Thus, for practical purposes sectors are 1024 bytes long, and there are alternately five and four to a track. In terms of the Tr/Sec numbering system, this means the odd number tracks have only even number sectors, and vice versa. Most of the consequences of this are hidden from the user, but having this background knowledge may help explain some oddities in Tr/Sec numbering. An example of this in action in DiscPower is in loading a non-AMSDOS sector, e.g. Track 2 Sector 6, the EDIT display given will start at 0200, i.e. the middle of the buffer. DiscPower has loaded the two sectors 2/5 and 2/6, but, knowing that you really wanted only the second of these, has started you in the middle of the buffer, at the start of 2/6.

3 READ DISC FILE

The disc directory is displayed in mode 2 (a full directory could not fit on the screen in mode 1). The cursor can be moved around the screen and used to select a file for loading. When the cursor is on the correct file press ENTER to get the first sector of the file displayed, as in EDIT. The normal EDIT commands apply, except for CTRL/CR and CL. These move to the previous and next sectors <u>of this file</u>, following it around the disc (CP/M is liable to scatter bits of file all over a disc). You will be told when you reach the end of the file (in either direction).

4 READ ROM

First the ROMs are scanned and a display of information concerning them shown. This gives the ROM number, type of ROM, version number and name for each ROM found. To select a ROM move the cursor to the appropriate line and press ENTER. A copy of the first 1K of the ROM is made in the buffer, and the EDIT display is given.

5 EDIT MEMORY

This command displays the current contents of the buffer. On first entering DiscPower this will normally contain all 00s. When the commands to load a file, a ROM, a sector, or a directory are used, the resulting data is put into a buffer in RAM, and it is this area which is examined by EDIT: indeed all these commands effectively put you into the same situation as does EDIT. Leaving one of these commands and then using EDIT will redisplay the contents of the buffer.

The EDIT display consists of four areas. At the top of the screen the first area shows information about what kind of data you are editing.

```
File A.TXT  Track 02 Sector 05   or
ROM number 01   GHOULS   C000
```

When looking at a ROM the address at the right shows whereabouts in the ROM you are, and changes as you move the cursor around.

Whatever you are examining there is a single letter H at the end of the top line. This tells you that entry is currently in hex. You may change to ASCII character entry, and back, by pressing \.

Below this at the left-hand side of the line is the address of the start of that line within the edit buffer, which starts at A000H.

In the centre will be the contents of the file or ROM, in hexadecimal, with the ASCII code equivalent to its right (non-printing codes represented by a ".").

In a four colour mode the hex code will be in pen 1, the ASCII and the address in pen 2, and the title in pen 3.

You may now edit the display by typing in the hex code you want, or switch to text by pressing \. If you then want to enter more hex, press \ again first.

The characters you type in will appear in pen 2 (in a four colour mode). When you press enter the changes will be made, and their colour will change to pen 1. Obviously, whilst your changes will affect the buffer immediately, you cannot edit a ROM in this way. Changes to a disc file or directory will only be made when you write to the disc, using option 6 from the main menu.

TAB copies the current screen to the printer.
COPY gives a copy facility. This prompts for start, end and destination addresses, and then performs an "intelligent" copy. If you have a ROM or file in the buffer, and press ENTER in response to the Start Address prompt, then the whole of the remainder of the ROM (or file) will be copied to the destination. Note i) this means that you cannot copy from 0 (though you can from 1); ii) if you press ENTER in response to the Destination prompt as well, you will copy to 0, but this will have the effect of corrupting the computer's system variables, and cause it to crash.

CTRL/A reselects the address of the part of memory being examined, and changes the top line of the display to "examine" mode. In this mode the top line has:
1. The address of the byte over which the cursor is placed (the "current byte"),
2. The current byte.

3. The ASCII character represented by the current byte (the top bit, if set, is ignored),
4. The binary equivalent,
5. A disassembly of the instruction represented by the byte (and subsequent ones if necessary),
6. The Hex/ASCII flag.
Pressing CTRL/A again and entering A000 as the New Address will return you to the normal EDIT display.

CTRL/D starts disassembly on the screen.
CTRL/P starts disassembly on the screen, and sends output to the printer as well. (The CLR key produces the same code as CTRL/P, and could equally well be used.)

CURSOR KEYS move as usual round the screen display. SHIFT/CR and SHIFT/CL move to the right and left ends of the line. SHIFT/CU and SHIFT/CD move to the previous and next "page" of the display, keeping within the buffer. In these cases the cursor will be at the top left of the new page. With discs CTRL/CR, CL, CU and CD will read the previous/next, sector/track, i.e. if you are on Track 4 Sector 5 CTRL/CR takes you to Tr 4 Sec 7, whilst CTRL/CU takes you to Tr 3 Sec 4 N.B. Remember that, because of interleaving, only alternate sectors are available i.e. on odd number tracks you can only read even number sectors, and vice versa, so that the next sector after Tr 4 Sec 5 is sector 7. On the other hand sector 6 was read in with sector 5, and is already part of the buffer. Similarly, when you use CTRL/CD or CU, you are moved by 10 sectors, rather than a track (9 sectors), to give you a readable sector.

With a ROM CTRL/CR and CL move you backwards and forwards by a buffer full, i.e. 1K or 400H. CTRL/CU and CD move by 4K or 1000H. Thus without CTRL you move around the buffer, with it you change the contents of the buffer.

6 WRITE SECTOR TO DISC

WARNING THIS OPTION CAN SERIOUSLY DAMAGE THE HEALTH OF YOUR DISCS USE WITH EXTREME CARE AND ONLY ON A BACKUP COPY !!!!

If you have modified a sector loaded from disc - including the directory - you may wish to write the modified version back to the disc. On selecting this option you are prompted with the current Track and Sector numbers, and asked for confirmation. "Y" writes the sector, any other key abandons the write.

7 DISASSEMBLE/TABULATE

Give the start and end addresses of the code you wish to examine (the first will be A000 if your code is in the buffer as the result of a ROM or file load), followed by the address at which it would normally be located when it was executed, e.g. the start of the upper ROM is always at C000H.

After a check on whether you require printer output, the display shows address, code, disassembled instruction and ASCII equivalent. Selecting a number 1 - 9 will display lines in that multiple of six at a time. 0 returns to single line mode. Any key now diplays more lines of code in the predetermined number, except C, which switches to continuous output, J, which follows jumps and calls, or ENTER, which displays a single line even if the multiple line option (using a number as above) is on. Note that ALL values in the disassembly are in hex.

8 SEARCH MEMORY

This facility allows you to search a file for a string of characters or hex codes. The length of the string is limited to one line of the EDIT data display, i.e. if you are using a forty column screen the limit will be 8 characters, with 16 available if you use the 80 column mode. ? can be used as a "wildcard". If a ROM is selected, or a sector has been loaded by fileload, then you are offered the chance of searching the whole file (or ROM).

To enter the "target" for the search, you may either enter a string of hex values (with or without spaces or commas separating them), or a string of characters. In the latter case you must preceed your string by quotes (") if you wish the search to check on the case of a letter, or slash (/) if the case does not matter. (For the technically minded any character with ASCII code from 21 to 2F hex will do to start the string: if the code is 28 or greater, the search will be case insensitive. If you need to embed one of these characters in the target, simply use a different one as a prefix.)

When a match is found, a line of edit screen is displayed. This is repeated until either all matches have been displayed, or the screen is full. In the latter case CTRL/F will continue the search.

9 UTILITIES

Pressing 9 takes you to another menu.

1. Format Disc

You can use this option to format a disc without going into CP/M. The Data Only, Vendor, and IBM options are available. The standard CP/M format is not, since it would require the presence of a CP/M disc to copy the system tracks from. You may, of course, copy the system tracks onto a Vendor disc at a later stage, using CP/M utiliies.

2. Backup Disc

Again this routine simply allows you to backup a disc without requiring entry to CP/M.

3. Disc Map

This gives a pictorial representation of the type of data stored on each sector of the disc. Each character position represents a sector, each rectangle a track. The numbers relate to the track above; the dots are a guide to the sector.

4. File Map

Effectively this shows what is in the directory, even if the corresponding file has been erased, or if there is more than one entry for the file (in the case of a file using more than one extent). After the filename is: i) the number of the "extent" (nothing if this is the first extent or if there is only one), ii) A or E for "active" or erased, iii) the user number. If you position the cursor over a filename and press ENTER, the Disc Map display is given, and the selected file is highlighted.

5./6. Read/Write File

Using these two options you can read a whole file into memory, or write from memory to a file, in a rather simpler way than from BASIC.

7. External Commands

This will help you to use commands from a background ROM, e.g. |ERA *.BAK, or enter a different foreground ROM directly (remember that the latter will erase all data in memory).

8. Calculations

A facility for making conversions between bases, and performing hex calculations.

You are prompted for input, which is assumed to be in hex unless preceded by a T (for base Ten, or decimal values) or an X (for binary values).

If you enter a single number you are given the hex, decimal, negative decimal and binary equivalents. The binary value is padded with leading zeros to 8 or 16 bits according to whether the initial input was less than 255 (decimal) or not. The hex value is always given as two bytes (four hex digits).

Alternatively you may enter two values, A and B, separated by a comma or spaces. The display then shows B + A and B - A, together with the value required by a jump relative (JR) instruction, from B to A, and its decimal equivalent. If the difference between the values is too great for a JR, then the message "Out of Range" will appear. All these values will be displayed in hex.

0 SCREEN

This option allows you to choose between 40 and 80 column modes, and between 12 and 24 lines of data. You can also choose the inks assigned to the background, and each of the 3 foreground colours.

INDEX

MICROPOWER

SUPERPOWER–SERIOUS SOFTWARE FROM THE MICROPOWER ORGANISATION