

=====

Unlike the MOS R and W Commands this Utility Programm enables You to access any byte of any file on disc - directly without computing where it might be. But, DDE tells You the physical address of the displayed sector as well.

Displayed are 128 (80H) bytes a time, the logical filesector is used for direct access and displayed besides the filename. Every line of code is starting with the relative Byte Address of the first byte in that line, contending 8 Hex-Bytes and then their ASCII-Equivalent (Control-Characters are replaced by a dot).

The two lower bytes of the rBA are also the position bytes.

00 to 70 Position in even sectors (0,2,4...)
80 to FO Position in odd sectors (1,3,5...)

If there was no filename given as a parameter to DDE when starting the program, you will be asked for it. The ESC-Key will enable you to leave the current level, switching back to DISPLAY-Mode or HELP-Mode.

MAIN COMMANDS:

D - Default drive setting
N - New Filename

> - display next sector of 128 bytes
+ - as >
< - display previous sector of 128 bytes
- - as <
, - display first sector of given file
. - display last sector of given file
S - Sectornumber, the logical sectornumber may be given in decimal or hexadecimal (OhhhhH)

A - ASCII-Modification
H - HEX-Modification

Q - Quit , leave the program and return to DOS

Notice:

On the 80-Column-Card only 40 Characters per line are used. On standard colour display devices the backdrop colour will tell you the program status. It is green whilst in display-mode, red during modification and grey anytime else. To improve readability on a home-TV-set text is displayed black on light brown.

No . Com, or whatever
will get you the directory - first.

Modification:

You start modifying the currently displayed sector by typing A for ASCII or H for HEX (depending on which side you want to start modifying). Then you will be asked for the position the cursor is to be located (just typing the ENTER-key will default to the upper left corner).

ASCII-Modification is done by overtyping the displayed text with convenient letters or even graphics. The graphic-key may be used to enter control-characters on the ascii-side, but take care to reset bit 7 before writing back to disc.

Using Hex-Modification you have to remember that a hexnumber consists of two characters (range 0-9,A-F). Any changes made are then displayed.

The arrow-keys are used for positioning the cursor. You are not able to leave the sectordisplay - going down the bottom will cause a reenter at the top.

To make this tool really powerful there are special control-keys defined:

- | | | |
|---------|---|--|
| CTRL D | - | Cursor right |
| CTRL H | - | Cursor left |
| CTRL J | - | Cursor down |
| CTRL K | - | Cursor up |
| INS-key | - | makes room for one character, dropping character at position 127 (7FH) and inserting a blank or 00 |
| CTRL Z | - | same as INS-key |
| DEL-key | - | deletes one character under the cursor and fills position 127 (7FH) with blank or 00 |
| CTRL Y | - | same as DEL-key |
| CTRL A | - | hardcopies the screen to a parallel printer (take care for hangups or use dummy cable) |
| CTRL U | - | toggles bit 5 (upper/lower case) |
| CTRL T | - | toggles bit 6 |
| CTRL S | - | toggles bit 7 (Graphics/non graphics) |
| CTRL F | - | fills the sector from cursor to end with the character at cursor |
| CTRL V | - | fills the sector beginning with position 0 up to the cursor with the character at cursor |
| CTRL C | - | complements the character (Z80-CPL) |
| CTRL L | - | Rotate left. Bitmask of byte is rotated left (Z80-RLC), e.g. bit0 to 1, 1-2, 2-3, , 7-0 |
| CTRL R | - | Rotate right. Bitmask of byte is rotated right (Z80-RRC), e.g. bit7 to 6, 6-5, 5-4, , 0-7 |
| CTRL X | - | Switch to other side of Hex/ASCII-Modification |
| ENTER | - | writes back to disk |
| CTRL M | - | same as ENTER |
| ESC | - | Escape from Modifying - sector is reread and then displayed. All modifications are cancelled. |

=====

PSG is shorthand for Programmable Sound Generator. This highly integrated chip is used in the EINSTEIN-Microcomputer to generate music and noise. But it is also used to provide keyboard input. To access the chip in BASIC would normally cause a hang-up, due to disabling the keyboard. This is one reason why this utility was written. But the main reason to use it is the easy way to play with the facilities of your PSG and the means to store your special noises for later use.

You alter the contents of your PSG by setting the registers bitwise on your registertable displayed on screen with binary, decimal and hexadecimal values for register 0 to 0DH. Move the cursor with the arrow keys to the position to be altered, and then set or reset the bit. Setting is done by typing an I or a 1. Resetting is done by an O or a 0.

As you might already have noticed, some bit fields are blank. They either have no effect on the PSG or they are not to be accessed, to prevent hang-ups.

Programming examples:

To achieve the same effect as MUSIC "C" in XBAS, you would have to program:

Reg. 0 1101 1101
Reg. 1 0001

Reg. 7 1 (bit 0)
Reg. 8 0 1010

The sound of the sea:
(Channel B)

Reg. 7 0 ???? (bit 4)
Reg. 9 1 ???? (bit 4)
Reg. C 0010 1000
Reg. D 1010

and now play with the contents of Reg. 6 (frequency).