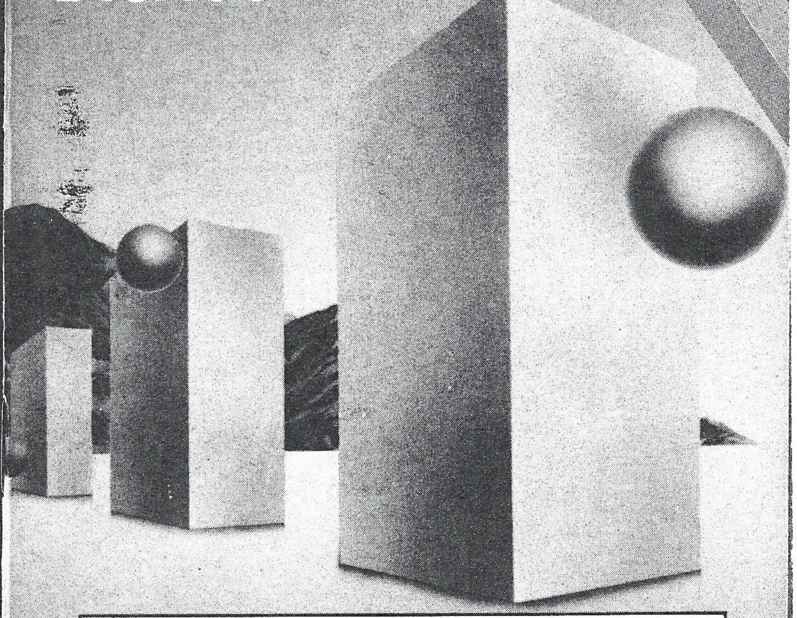
sinclair®

ZX Spectrum®

# MONITOR AND DISASSEMBLER



SOFTWARE BY Crystal Computing

CASSETTE
16K/48K RAM

# USER MANUAL FOR MONITOR AND DISASSEMBLER

First published in 1983 by
Sinclair Research Ltd
25 Willis Road Cambridge CB1 2AQ England

ISBN 0 85016 005 8

Printed in UK

## Contents

## Loading and running your machine-code monitor and disassembler

1. Rewind the cassette to the beginning.
2. Type LOAD "monitor 16K" (16K version)
   or LOAD "monitor 48K" (48K version)
   followed by ENTER
3. When loaded the program prints the message LOADED O.K. PRESS ANY KEY. Stop the tape. When you press a key, you will return to BASIC, leaving Monitor and Disassembler safely above RAMTOP.
4. to enter Monitor and Disassembler, type:
   PRINT USR 29369 (16K version)
   or PRINT USR 62137 (48K version)
5. You will see the title page appear, along with the READY message, and flashing cursor. Pressing almost any key now will result in the command token word being displayed. Some of the commands will work without parameters (PRINT, EDIT, DISASSEMBLE), but most will return with the message PARAMETER ERROR. To find out how to use all the commands please read the next section.

## Assign type 'A'

The ASSIGN command is used to alter the values which will be loaded into the CPU registers when, using the GOTO command, a machine-code routine is next executed.

ASSIGN takes two parameters. The first is the register you wish to assign a value to. This may be any one of the following:

A,B,C,D,E,H,L,BC,DE,HL,B',C',D',E',H',L',BC',DE',HL'.

The second parameter is the value to be given to the specified register and, like all numerical parameters, may be entered in either hexadecimal or decimal form (decimal numbers must be preceded by a '£' character). The two parameters must be separated by an equals sign.

GENERAL FORM OF THE COMMAND:

ASSIGN register=value

example: ASSIGN BC=£16514

## Break type 'B'

The BREAK command sets a breakpoint at the address in RAM specified as its first parameter.

A previously set breakpoint may be removed using the command BREAK O. A breakpoint is removed automatically when it is executed.

example: BREAK 5000

On encountering a breakpoint, the monitor displays the CPU registers along with the menu:

'Q'-QUIT 'C'-CONT 'M'-MONITOR

Entering:

'Q' will return control to the monitor
'C' will continue execution of the machine code routine from the BREAKPOINT
'M' produces the following reply: READY
?

In this mode, the machine code run may be continued from the BREAKPOINT even after memory and/or registers have been inspected and perhaps modified. When you wish to continue the run (or return to normal monitor mode) type 'X' (Exit). The register display is presented along with the Q/C/M menu. You may now proceed as though the BREAK had just been executed:

'Q'–Return to monitor
'C'–Continue run
'M'–Re-enter special monitor mode to inspect/modify before continuing run.

## Copy type 'C'

The COPY command moves a block of bytes from one location in memory to another.

COPY takes three parameters:

1 the start address of the block to be copied from,
2 the finish address the the block to be copied from,
3 the start address of the block to be copied to.

The three parameters must be entered in the order given above, separated by full stops. Decimal and hexadecimal parameters may be mixed if required, and the two blocks may overlap.

GENERAL FORM OF THE COMMAND
COPY start address 1.end address 1.start address 2

| | | |
|---|---|---|
| examples: | COPY 0.10.5000 | copies 0–10 to 5000–5010 |
| | COPY 5100.5200.5000 | copies 5100–5200 to 5000–5100 |
| | COPY 0.£500.£16514 | copies 0–1F4 to 4082–4176 |

## Disassemble type 'D'

The DISASSEMBLE command converts hexadecimal code into standard Zilog Z-80 mnemonics. It takes one parameter, the address to start disassembling from. Initially, 15 instructions are displayed, along with the original hexadecimal bytes and the address of the instruction. To return to command level, press 'break'; any other key continues disassembly.

example: DISASSEMBLE 02BB

## Edit type 'E'

The EDIT command provides a movable 'window' into memory. On entry to EDIT mode you will see displayed twenty addresses, and their contents in both hexadecimal (right hand column) and character form.

The contents of the address pointed to by the cursor may be altered simply by typing the new hexadecimal value.

Other EDIT mode commands are:

| | |
|---|---|
| newline | –scroll forwards by one byte |
| 'J' | –jump backwards one byte |
| 'Q' | –quit editing and return to command level |

EDIT takes one parameter, this being the address to start editing at.

example: EDIT 5000
EDIT £16530

## Find type 'F'

The FIND command finds all occurrences of the string given as its third parameter, in the area of memory specified in the first two parameters.

The first parameter is the address at which to start looking for the string, the second is the address at which to stop. The third parameter may be any one of the following:

1 a single hexadecimal byte,
2 a string of up to seven hexadecimal bytes separated by full stops,
3 a single decimal number in the range 0–255 preceded by a '£' sign,
4 a string of up to four decimal numbers, each of which must be preceded by a '£' sign, separated by full stops,
5 a string of up to twenty characters, enclosed in quotes,
6 any combination of the above.

GENERAL FORM OF THE COMMAND:
FIND start address.finish address.string

examples: 1 FIND 0.100.C9
2 FIND 0.100.2A.0C.40
3 FIND 0.100.£201
4 FIND 0.100.£42.£12.£64
5 FIND 0.100."HELLO"
6 FIND 0.100."CR".3E.£51."TAL"

## Goto type 'G'

The GOTO command loads the CPU with the values as displayed by the REGISTERS command (see below), then executes machine-code from the address given by its parameter.

Return to command level is accomplished by the use of a RET instruction at the end of the machine-code routine.

If a breakpoint has been set, and is encountered, execution will be halted and the current register values displayed. For details, please refer to 'BREAK' command.

example: GOTO 5000

## Jump relative calculator type 'J'

The JR command calculates the correct offset byte (the byte immediately following a JR or DJNZ instruction), given two parameters as follows:

1 the address of the instruction
2 the address of the instruction to be jumped to.

example:

Given the program below, the offset byte would be given by:

JR 5012.5000

the answer being OFFSET BYTE=EC

```
5000    LD A.18          3E 18
5002    CALL 2BB         CD BB 02
5005    INC L            2C
5006    CALL Z,4082      CC 82 40
5009    LD (6000),HL     22 00 60
500C    SBC HL,DE        ED 52
500E    RES 7.(HL)       CB BE
5010    BIT 1.L          CB 4D
5012    JR NZ,5000       20 ?
```

## Convert type 'K'

Converts between hex and decimal. CONVERT takes as its parameter any hex number in the range 0-FFFF, or any decimal number in the range 0-65535, giving as its result the corresponding decimal or hex value.

example: CONVERT 5000
=£20480

## Message type 'M'

MESSAGE takes two parameters, the first being the address at which the string is to be inserted. The second is the string itself, which may be up to nine characters in length, enclosed in string quotes.

GENERAL FORM OF THE COMMAND:

Example: MESSAGE address "string"
MESSAGE 6000 "CRYSTAL"

## Print type 'P'

The PRINT command displays in character form the contents of any 64 consecutive addresses in memory. The address of the first character is given by the first parameter.

example: PRINT 2BB

## Registers type 'R'

The REGISTERS command displays the contents of the CPU registers in easy to read form.

The registers displayed are:
A,BC,DE,HL,BC',DE',HL'IX,IY,SP,(SP),(HL).

The FLAGS register is displayed separately, allowing the inspection of each flag. Any breakpoint set is also displayed.

## Substitute type 'S'

The SUBSTITUTE command replaces all occurrences of one byte within a block with another byte. The block to be searched is specified by the first two parameters following the command. These are, respectively, the start address of the block and the finish address. The third parameter is the byte to be replaced. The byte to replace it with must be given as the fourth parameter.

Example:

SUBSTITUTE 5000.5100.CD.CC

replaces all occurrences of the bytes CD in the block 5000 to 5100 with the byte CC.

## Tabulate type 'T'

The TABULATE command displays the contents of any 64 consecutive addresses as hexadecimal bytes. The address of the first byte displayed is given by the first parameter.

example: TABULATE 7E

## Verify type 'V'

The VERIFY command compares two blocks in memory, returning with the message 'O.K.' if they are equivalent. If two bytes are found which are not equal, the message 'ERROR AT', followed by the address in the second block at which the error was found, is printed, and further comparison halted.

VERIFY takes three parameters, the first and second are the start and finish address of the first block. The third is the start address of the second block.

GENERAL FORM OF THE COMMAND:

VERIFY start address 1.finish address 1.start address 2

example: VERIFY 5000. 5100. 5200

## Exit type 'X'

The EXIT command causes a return to the BASIC system. You may re-enter Monitor and Disassembler at any time using the BASIC command PRINT USR 29369 (16K version)
or PRINT USR 62137 (48K version)

## Zero type 'Z'

The ZERO command sets all bytes in the specified block to zero. Its two parameters are the start address and finish address of the block to be zeroed.

example: ZERO 5000.510

## Appendix 1–List of commands

| KEY | COMMAND | PARAMETERS |
|---|---|---|
| A | –ASSIGN | REGISTER=VALUE |
| B | –BREAK | ADDRESS |
| C | –COPY | START ADDRESS 1.FINISH ADDRESS 1.START ADDRESS 2 |
| D | –DISASSEMBLE | address |
| E | –EDIT | address |
| F | –FIND | START ADDRESS.FINISH ADDRESS. string to be found |
| G | –GOTO | ADDRESS |
| J | –JUMP RELATIVE | address from. address to |
| K | –CONVERT | NUMBER |
| M | –MESSAGE | "message" ADDRESS |
| P | –PRINT | address |
| R | –REGISTERS | no parameters |
| S | –SUBSTITUTE | "name" START ADDRESS.FINISH ADDRESS. BYTE 1 . BYTE 2 |
| T | –TABULATE | address |
| V | –VERIFY | start address 1.finish address 1.start address 2 |
| X | –EXIT | no parameters |
| Z | –ZERO | START ADDRESS.FINISH ADDRESS |

Notes Parameters in capital letters **must** be entered.

Parameters in lower case may be omitted, and the last parameter input will be assumed.

## Appendix 2–ZX Printer

The ZX Printer may be driven by entering CAPSHIFTed"1" after each of the following commands (and their parameters if required) and then pressing ENTER:

DISASSEMBLE
FIND
PRINT
REGISTERS
TABULATE

The keyword LPRINT will appear after the command.

Notes After a block has been disassembled to the printer, press ENTER for the next block or BREAK to return to the monitor.

If BREAK is pressed while the printer is operating, you will return to BASIC. Re-enter the monitor as usual.