

IDENTIFICATION

Product Code: Digital 8-12-S
Product Name: ODT - II Revised
Date Created: December 7, 1966
Maintainer: Software Service Group

2. ABSTRACT

ODT-II (octal debugging tape) is an aid used in debugging a PDP-8 program by facilitating communication with the program being run. The communication between operator and program is via the ASR-33 Teletypewriter.

ODT-II features include register examination and modification, binary punchouts, control transfer, word searching, octal dumping and instruction trapouts to ODT-II control (break-traps). ODT-II makes no use of the program interrupt facility and is restricted to one memory module.

3. REQUIREMENTS

3.1 Storage

ODT-II requires 600 (octal) registers and resides in memory between 7000 and 7577 (or 1000 - 1577). In addition, ODT-II uses register 0005.

3.3 Equipment

Standard PDP-8.

4. USAGE

4.1 Loading

ODT-II is loaded by means of the Binary Loader (see Digital-8-2-U).

4.1.1 Place proper ODT-II tape in reader (either high or low version).

4.1.2 Set the SWITCH REGISTER to 7777 and press LOAD ADDRESS.

4.1.3 If using the 750 Reader, put switch 0 down and press START. If using the 33-ASR Reader, put switch 0 up and press START.

4.1.4 Read the binary tape of the program to be debugged into core as above. (Make sure that the program to be debugged does not overlap ODT-II and vice versa).

4.4 Start-Up

The starting address of ODT-II is 7000 (1000 is low version).

4.4.1 Set ODT-II starting address in the SWITCH REGISTER (1000 or 7000).

4.4.2 Press LOAD ADDRESS, and press START. ODT-II is now running and is awaiting commands from the 33-ASR Keyboard.

5. RESTRICTIONS

5.1 Status Active Registers

Register 0005 (page 0) is used by ODT-II when executing a break-trap. This register is filled by ODT-II with an address within ODT-II. When the operator requests a break-trap, ODT-II saves the instruction at the location of the trap and replaces it with the instruction, JMP I 5, thus returning control

to ODT-II. ODT-II also uses 600 (octal) locations from 1000 - 1577 (low version) or 7000 - 7577 (high version). The programmer must be certain that his program to be debugged does not overlap ODT-II.

5.2 Status Core

ODT-II will not operate outside of the core memory module in which it resides.

6. DESCRIPTION

ODT-II is an octal debugging program for the PDP-8. Operator communication occurs on the Teletype (33-ASR). A number is a string of digits taken as octal. Other characters are either control characters or are illegal.

One of ODT-II's most useful features is the "break-trap". When debugging a program, it is often desirable to allow control to flow up to a certain point, at which time the programmer would like to examine the contents of the accumulator, the link, and various other registers of his program. To facilitate this, ODT-II will insert in the user's program a jump instruction into ODT-II, which will cause the C(AC) and C(L) to be preserved. ODT-II will then print out the break location and the contents of the accumulator at that point. It is then possible to move the break-trap and continue the program, restoring the C(AC), C(L), and the trapped instruction. The trapped instruction is executed and control is returned to the user's program.

6.1 Discussion

6.1.1 Commands

6.1.1.1 Slash (/)

The register examination character / causes the contents of the register addressed by the preceding expression to be typed out in octal. The register is then opened for modification and the location sequence reset to the address.

Example:

1/ 0027

(In all examples, everything underlined is typed by ODT-II; everything not underlined is typed by the user.)

6.1.1.2 Carriage Return ()

The value of any expression that has been typed by the user immediately preceding the carriage return will replace the contents of the open register, if any register was open, and the register will be closed. If nothing has been typed by the user, the contents will not be changed.

Example:

1/ 0027) Register 0001 is unchanged
1/ 0027 14) Register 0001 will contain 0014

6.1.1.3 Line-Feed (\downarrow)

The line-feed has the same effect as the carriage-return, but, in addition, the next sequential register is opened and its contents typed.

Example:

1/ 0027 \downarrow
0002/ 0077 Register 0002 is opened; C(0001) is unchanged.

1/ 0027 14 \downarrow
0002/ 0077 Register 0002 is opened; C(0001) is 0014.

6.1.1.4 Any Illegal Character

Any character that is neither an octal digit nor a valid control character causes the current line to be ignored.

Example:

19 ?

6.1.1.5 Single Quote (')

Go to the location specified before the ('). All indicators and registers will be restored and the break-trap, if any, will be inserted. Typing (' alone is an error.

6.1.1.6 Double Quote (")

Conditions ODT-II to insert a break-trap at the location specified before the ("). If no such location was specified, ODT-II removes the previous breakpoint. A breakpoint is actually inserted only when a goto (') or proceed (!) is executed. ODT-II will remove the instruction at the break location and will save it for future restoration. The instruction at the break location is only executed after the proceed is given.

6.1.1.7 Exclamation (!)

Proceed from a break-trap. The C(AC) and C(L) are restored, the trapped instruction is executed, and control transfers to the user's program.

NOTE: If a trap set by ODT is not encountered by the object (user's) program, the breakpoint instruction will not be removed.

Restarting ODT-II at its starting address (1000 or 7000), however, will restore the original instruction. The breakpoint will be reinserted when goto (') or proceed (!) is executed. When a break-trap is encountered in the user's program, ODT-II will type out the location of the trap, a right parenthesis, and the contents of the accumulator. The accumulator and link are saved in registers 7350 and 7351, respectively, (1350 and 1351 for low version). These registers may be examined (and modified) in the normal manner by use of the slash (/) control character.

6.1.1.8 M

Typing the letter M causes the contents of the mask register (see 6.1.1.11) to be printed and opened for modification. Initially, this value is set to 7777.

6.1.1.9 < (Left Angle Bracket)

Typing this character causes the contents of the lower search limit register (see 6.1.1.11) to be printed and opened for examination. Initially, this value is set to 0001.

6.1.1.10 > (Right Angle Bracket)

Typing this character causes the contents of the upper search limit register (see 6.1.1.11) to be printed and opened for examination. Initially, this value is set to 6777.

NOTE: The registers named M, <, and > are within ODT-II and have addresses 7550, 7551, and 7552, respectively.

6.1.1.11 Word Searches

Word searching using ODT-II is similar to word searching using DDT (see Digital-8-4-S). The searching operations are used to determine if a given quantity is present in any of the registers of a particular section of memory. The search is initiated by the command:

KW

Where K is an octal expression. ODT-II will perform a word search and print the address and contents of every register in the desired section of memory whose contents, when masked by C(M) are equal to the value of the expression K. The search continues between the limits specified by C(<) and C(>).

Example: Search locations 2000 to 4000 for registers with an op-code of 2 (ISZ).

M	7777	7000	Set the mask
<	0001	2000	Set the lower limit
>	6777	4000	Set the upper limit

2000W Initiate the search

2021/ 2467

...
...

etc.

6.1.1.12 Punch Commands

K;l

(ALT MODE)

Where K is the initial address (octal) and L is the final address (octal) of the register to be punched. The computer will halt, allowing the operator to turn the Teletype off line and to punch leader codes. Putting the Teletype back on line and depressing CONTINUE initiates the actual punching of the block. The punching terminates without punching a checksum to allow subsequent blocks to be punched and an all inclusive checksum to be punched at the end. This procedure is optional, however, and the user may punch individually checksummed blocks. See 6.1.1.13 for a description of the checksum command. If subsequent blocks are desired, turn off the punch before typing the pertinent commands.

6.1.1.13 * (Checksum Command)

Typing an asterisk (which does not print) causes the program to punch out the accumulated checksum. The program will then halt to allow the operator to turn the Teletype off line, produce trailer code (200), and remove the tape. The operator should then turn the punch off, turn the Teletype on line and depress CONTINUE to resume ODT-II operation.

6.1.2 Command Summary

Slash (/)	Register examination
Carriage Return	Register is closed
Line-Feed	Closes register and opens the next one
Illegal Character	Current line is ignored. ODT-II types ?
Single Quote (')	Goto
Double Quote (")	Set a break-trap.
Exclamation (!)	Proceed from trap.
M	Examine (and modify) mask
<	Examine (and modify) lower search limit
>	Examine (and modify) upper search limit
W	Word search command
Semicolon (;)	Separate punch arguments
Alt Mode	Punch in binary
Asterisk (*)	Punch checksum

6.2 Examples and/or Applications

6.2.1 Symbols for representing "invisible" Teletype actions:

(CR)	= Carriage Return
(LF)	= Line Feed
(ALT)	= Alt Mode
(*)	= Nonprinting Asterisk
(H)	= Halt
(C)	= Continue
(PON)	= Punch On
(POF)	= Punch Off
(TON)	= Teletype On
(TOF)	= Teletype Off
(LEAD)	= Production of Leader
(TEXT)	= Punching of Text
(CKSM)	= Punching of Checksum

6.2.2 Underlined functions and expressions are those produced by ODT-II.

6.2.3 Brackets enclose comments local to this description.

6.2.4 Examples of Register Examination and Modification

100/ <u>1234</u> 7777 (CR) (<u>LF</u>)	[Examine and modify.]
100/ <u>7777</u> (CR) (<u>LF</u>)	[Examine only.]
(LF) (<u>CR</u>)	[Examine next.]
0101/ <u>7564</u> (LF) (CR)	[Examine next.]
<u>0102/ 4567</u> 2222X? (CR) (LF)	[Illegal character.]
<u>102/ 4567</u> 2222? (CR)(LF)	[More than four integers typed.]
102/ <u>4567</u> 2222 (<u>CR</u>) (<u>LF</u>)	[Correct modification.]
102/ <u>2222</u> (CR) (<u>LF</u>)	[Success at last!]

6.2.5 Word Searching Example

M <u>7777</u> (CR) (LF)	[Examination of parameters.]
< <u>0001</u> (CR) (<u>LF</u>)	
> <u>6777</u> (CR) (<u>LF</u>)	
<u>7200W</u> (CR) (LF)	[Search for all CLA instructions.]
0150/ <u>7200</u> (CR) (LF)	
<u>1216/7200</u> (CR) (LF)	
M <u>7777</u> 600 (CR) (LF)	[Set mask for indirect and page bits.]
< <u>0001</u> 200 (CR) (LF)	[Set limits to exclude page 0.]
<u>400W</u> (CR) (LF)	[Search for all indirect references to page 0 from other pages.]
<u>1426/3403</u> (CR) (LF)	
<u>1627/5400</u> (CR) (LF)	
<u>1705/7402</u> (CR) (LF)	[HLT looks like an indirect ref. to page 0.]
(LF)	[Searching completed.]
M <u>0600</u> 7777 (CR) (LF)	[Reset mask]
< <u>0200</u> 6777 (CR) (LF)	[and lower limit.]

6.2.6 Octal Dump Example

[We wish to dump core between 1000 - 1005]	
M <u>7777</u> 0 (CR) (LF)	[Set mask to 0.]
<u>0001</u> 1000 (CR) (LF)	[Set limits to encompass dump area.]
<u>6777</u> 1005 (CR) (LF)	
W (CR) (LF)	[Since W is typed alone, the word searched for is 0. The result after masking each register with 0 is, of course, 0 so all comparisons appear to the program equal and hence all unmasked contents are typed, constituting a dump.]
<u>1000/0001</u> (CR) (LF)	
<u>1001/0002</u> (CR) (LF)	
<u>1002/0003</u> (CR) (LF)	
<u>1003/0004</u> (CR) (LF)	
<u>1004/0005</u> (CR) (LF)	
<u>1006/0006</u> (CR) (LF)	
M <u>0000</u> 7777 (CR) (LF)	[Reset parameters.]
< <u>1000</u> 0001 (CR) (LF)	
> <u>1005</u> 6777 (CR) (LF)	

6.2.7 Example of Paper Tape Punchout (Binary Format).

[The punchout of page 1 and the first 50g registers of page 3.]
[The TTY is assumed to be on line and the punch turned off.]

200;377(ALT) (H) (TOF) (PON) (LEAD) (TON) (C) (TEXT) (POF) (CR) (LF)
600;647(ALT) (H) (PON) (C) (TEXT) (*) (CKSM) (H) (TOF) (LEAD) (POF) (TON) (C) (CR) (LF)
[ODT-II now running in input mode waiting further commands.]
[Remove tape from punch.]

6.2.8 Examples of Using Break-Trap Mode

241" (CR) (LF)	[Set break to occur at user's loc. 241.]
241/7200 (CR) (LF)	[Examination shows that trap inst. not yet inserted.]
200' (CR) (LF)	[Enter user's program at location 200.]
0241)0324 (CR) (LF)	[Break-trap occurs, and C(AC) typed.]
7351/0001 (CR) (LF)	[Examination of state of link at break.]
241/7200 (CR) (LF)	[Any amount of DDT functions may be indulged]
.....	[in at this point before continuing.]
253" (CR) (LF)	[Break-trap reset at loc. 253.]
! (CR) (LF)	[User's program continued from point of last break (241).]
0253)0000 (CR) (LF)	[Second break occurs.]
" (CR) (LF)	[Break-trap mode deactivated.]
! (CR) (LF)	[User's program continued without any traps set.]

[To reenter ODT-II, user would have to start at 7000.]

7. METHODS (Not Applicable)
8. FORMAT (Not Applicable)
9. EXECUTION TIME (Not Applicable)
10. PROGRAM