

DISASM: A TRS-80 DISASSEMBLER

What follows is a condensation of the DISASM instruction manual. The complete 12-page manual is available at a cost of \$3, which may be deducted from a future order.

I. INTRODUCTION AND POWER-UP

DISASM is a machine language disassembler for the TRS-80, Level II, whose primary purpose is the translation of Z-80 machine language object code into assembly language instructions using standard Zilog mnemonics. DISASM offers the essential features of a standard disassembler plus the ability to use its source code output with Radio Shack's Editor/Assembler. This compatibility gives DISASM the effective power of a disassembler plus an assembler, multiplying its usefulness many times.

DISASM features full table building with synthetic or user-assigned symbols. All addresses referenced in the object code are stored during disassembly and displayed on command. All addresses within an object code block referenced by instructions within that block are labeled. If no user-assigned symbol exists, then DISASM assigns synthetic symbols which will be recognized as labels by the Editor/Assembler. Thus, relocation of object code becomes as easy as changing the origin address in the assembler.

Besides relocation of object code, DISASM permits simplification of program editing. Neither a listing nor a source tape is necessary in order to modify an object program. Simply disassemble it in DISASM, load the source tape into the Editor/Assembler, edit and reassemble.

In addition to disassembly and table support functions, DISASM contains several utility functions. Among them: a sophisticated search routine that searches memory for text tables (search may be specified as "coarse" or "fine") and displays contents and location of found tables; the ability to transfer CPU control to any location in memory with one user-defined default value permitted; and a routine that loads object code from tape with automatic display of file name, starting address, ending address and entry address.

DISASM requires 3469 bytes of RAM, not including space for the symbol table whose size and location are user-specified (and may be omitted if desired). A Level II ROM must also be resident in the computer in order to operate the program.

DISASM loads under the Level II SYSTEM command. Upon entry into the program you are asked where you wish to locate the symbol table. The program accepts the first eight hex digits input, the first four being the symbol table origin and the last four being the address of the last byte allocated to the table. A space is printed by DISASM to separate the two addresses.

II. INPUT CONVENTIONS

In every case where a specified number of characters is expected it

is not necessary to press ENTER. All addresses are input and output in hexadecimal form. Backspacing is permitted until the final character of an address has been entered. Entry of the final character initiates execution of the command. The BREAK key may be used at any time, even during a disassembly or save, to return immediately to command mode.

III. SCROLL COMMANDS

Since DISASM is written primarily with the video user in mind, the program features great flexibility in control of the display. After the first page of a display has been completed, DISASM waits for one of four scroll commands before continuing: ENTER scrolls one line at a time. LINE FEED scrolls one page at a time. CLEAR clears the screen and displays the next page. Depressing the character Z continues the scroll for as long as it remains held down. As usual, BREAK signals an immediate return to command mode.

IV. DISASSEMBLING OBJECT CODE WITH THE D COMMAND

Upon entry of the D command DISASM asks for start and end locations of object code to be disassembled. The program normally makes two passes during a disassembly: the first builds the symbol table and the second displays the source listing. You may skip the table-building step and go right to the display by entering a starting address without an ending address. Disassembly continues until the BREAK key is pressed.

Two-byte immediate data may be expressed in the source code in one of three ways: a) As a symbol, if one was assigned to the data via the A command. b) As a synthetic symbol, if no symbol was assigned but the address formed by the data falls within the range of the block of code being disassembled. Synthetic symbols take the form "L" followed by the address followed by the "H" designation (e.g., 7ABD becomes L7ABDH). c) As a constant, if no symbol has been assigned and the address falls outside of the boundaries of the block of code being disassembled. All one-byte data are treated as constants.

The source code display consists of seven fields, four of which are saved to tape for use by the Editor/Assembler. The fields consist of the object address of the instruction at that particular line; a hex dump of the object code; the source line number, assigned by DISASM in increments of ten up to 65530; the label if the object address was found in the symbol table; the mnemonic, according to Zilog standards; the operands, if any; and the ASCII dump of the object code, treating bit 7 as zero.

At the conclusion of the display the source code may be saved to tape for later use by the TRS-80 Editor/Assembler for editing, relocation, reassembly, output to printer, etc. Two source lines are saved but not displayed: line 10 contains the ORG statement with the hex address and the last line contains the END statement.

V. MANIPULATING THE SYMBOL TABLE: COMMANDS A, S, R, M

The symbol table allows DISASM to index all lines referenced in a block of object code. The A command supports three operations on

the table: assign a label to an existing table entry, create a new entry and assign a label, or change a previously assigned label. Upon receiving the A command DISASM asks for the address (table entry) you wish to create or assign. Depending on whether the entry was found and whether it already has a label, the messages LABEL:, NEW LABEL:, or OVERWRITE LABEL: may be displayed. The label may contain up to six characters. Defaulting on an entry or hitting the BREAK key aborts the command without changing the table.

Once a label has been assigned to an address, it will be inserted in place of the address at every point where the address appears as an operand in the source display. The symbol appears in the label field at the line of the address itself.

The S command displays the entire table, four entries to the line, with assigned label immediately following the entry. Synthetic labels are temporary only and are not stored in the table.

The R command re-initializes the table, effectively deleting all entries without changing the location of the table itself.

The M command allows you to move the table to a new location, adjusting its size as desired. Execution of the command automatically re-initializes the table.

Each entry in the table requires three bytes of storage plus one byte per character in the label. Thus, the minimum required per entry would be three bytes; the maximum is nine.

VI. USING THE T COMMAND TO LOCATE AND DISPLAY TEXT TABLES

DISASM features a sophisticated text search routine that allows the quick location and review of text tables stored in memory. The routine searches the memory one byte at a time, ignoring non-alphanumeric characters and displaying all others as ASCII characters. At the start of each display line, the current address of the search pointer is displayed in hex form. The "coarse" search simply fills the display one line at a time. The "fine" search begins a new line if the end of the last line has been reached or if a non-ASCII byte intervenes between ASCII bytes. Thus, in searching and displaying a text table, each message is displayed on a separate line along with its starting address. Use of the T command is illustrated in the instruction manual.

VII. USER PROGRAMS AND ROUTINES: THE U AND J COMMANDS

Entering the J command causes the message JUMP TO: to appear. DISASM will accept a four-digit hex address and transfer CPU control to the input location. A default value for this command may be pre-set by entering the command U followed by the hex address. Once a value has been set you may respond to JUMP TO: with ENTER and control will go to the address last entered under the U command.

DISASM also provides space in its command table for the addition of two user commands, allowing calls to user routines by entering a

single character from command mode.

The B command returns control to Level II BASIC.

VIII. LOADING OBJECT CODE FROM TAPE WITH THE L COMMAND

Entering the L command causes the tape to start and the next file to be loaded. Upon encountering the file, the file name is displayed next to the L. After the file has been loaded, the first address loaded, last address loaded and entry address as specified when the file was saved are displayed. If a loading error is detected the letter E appears next to the file name and control returns to command mode.

IX. ERROR MESSAGES

DISASM allows three error messages: symbol table overflow, line number overflow (if line number exceeds 65536) and invalid object code. Symbol table overflow halts expansion of the table but does not otherwise affect disassembly. Line number overflow has no effect on disassembly (the line counter resets to zero). These two messages are displayed before and after the source code display. Invalid object code terminates disassembly and returns to command mode with the address of the error displayed.

* * *

Also included in the ~~included in the~~ instruction manual are sections on memory allocation and sample use.

DISASM COMMAND SUMMARY:

A Assign or overwrite label
B Go to MEMORY SIZE? in Level II BASIC
D Disassemble object code, create symbol table, save to tape
J Jump to an input hex address or pre-defined default address
L Load next file from tape. Display file name, start, end and entry addresses
M Move the symbol table to input locations.
R Re-initialize symbol table
S Display symbol table
T Search for and display ASCII text in "coarse" or "fine" mode
U Store input default address for J command
Others: space provided in look-up table for two user commands.

HOW TO ORDER:

DISASM occupies the top 3469 bytes of any 4K block of RAM. Thus, the final three digits, in hex, of its location will always be 292 to FFF. In a 4K system the first digit must be 4; a 16K system would allow the first digit to be a 4, 5, 6 or 7, etc. The program is mailed first class on certified cassette. Use the enclosed order form, and be sure to specify starting location in RAM.