LICENSED PROPRIETARY INFORMATION

The designs, information, data, software, or programming herewithin contained are submitted to the licensee in confidence, and are proprietary. These materials and programming are furnished for the one-time license fee, and are subject to the following restrictions:

> These materials shall not be copied, reproduced, lent, rented, nor sold in any form whatsoever without prior, written consent and permission from the Byte Shop of Westminster.

Receipt and use of these materials shall be construed as an act of acceptance of the conditions and restrictions specified herein.

LIMITED WARRANTY

I. Each diskette is checked, and is warranted to be free from errors.

II. The Byte Shop of Westminster warrants said condition for 30 (thirty) days after original license purchase.

III. The obligation incurred under this warranty shall be limited to the replacement of any diskette found to be defective, exclusive of return postage charges.

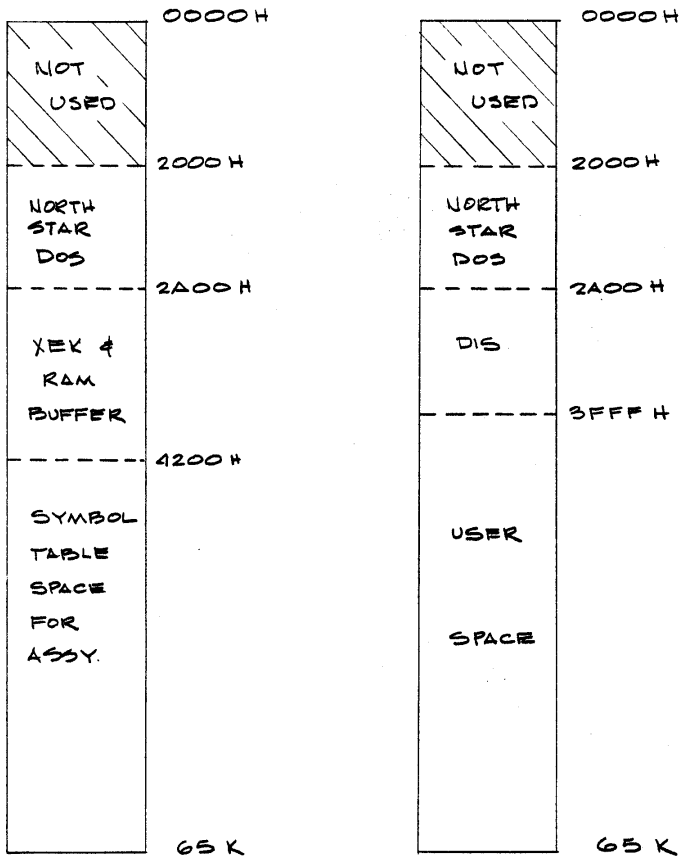
IV. To obtain replacement of such diskette, return it in its original sleeve to the Byte Shop of Westminster, postage paid, with return label.

V. Such replacement will not be made if it is found that the software contained on it has been altered or edited in any fashion, save for the addition of a disk operating system in the location provided for it.

VI. Licensee acknowledges that the Byte Shop of Westminster shall not under any circumstances be liable for any loss of use of the software or any other incidental or consequential costs, expenses, or damages, nor any expenses or damages caused by the use of this software or the supplied diskette.

COPYRIGHT © 1977

MEMORY MAPS



NOTES:

1. THE FIRST 8K IS AVAILABLE AS USER SPACE.
2. EACH DISKETTE HAS AN ASSIGNED FILE FOR THE USERS PERSONALIZED DOS.

XEK COMMAND SUMMARY

COMMAND	ACTION
:	Scans and checks file for proper format; file recovery
=	Displays sum and differences of two hexadecimal numbers
?	Initializes executive RAM space and clears file table
@	Displays data saved by program breakpoint
A	Assembles current source file to object code in memory
B.	Sets breakpoint to display all register data
C.	Concatenates cassette file and renumbers resulting file
D	Displays memory data in hexadecimal format
E	Enters displays and/or modifies memory byte by byte.
F	Creates, assigns, and displays file table data
G	Executes program at supplied address
H	Reads INTEL hexadecimal format data into memory
I	Reads Tarbell format cassette tape
J	Picks up data saved at breakpoint and jumps back to program
K	Kills line or lines from current file
L	Lists current file as originally entered
M	Moves current file or block of memory to new location
N	Renumbers source lines of current file by supplied increment
O	Outputs Tarbell format to cassette tape
P	Prints formatted listing of current file
Q	Quits and exits to North Star D.O.S. at 2Ø28H
R	Reads XEK format object or source from cassette tape
S	Sets autoline to supplied increment value

COMMAND	ACTION
T	Tests selected area of memory and displays errors
U	Unyanks (writes) current file to North Star Diskette
V	Verifies two area of memory and displays differences
W	Writes current file or memory to cassette (XEK format)
X	Displays current file unformatted and without line numbers
Y	Yanks (reads) source file from diskette to current file
Z	Zeroes or fills memory with selected constant

*"RUB" deletes last character and backspaces cursor

*A, D, L. P, and X may be temporarily stopped with the linefeed key and continued with return, or aborted with CONTROL-C.

*1111 four numeric digits supplies the line number for entry of a line in sequence into current file.

*Space supplies auto-incremented last line number for new line.

All commands are executed only after the complete command line is terminated by a carriage return, indicated by) in this document.

Commands refer to current file, which is indicated by the command F.

File names are contained in double quote marks such as "TEST", not more than five characters long.

Words contained in angle brackets (and) are optional, and are given as explanation only. They do not affect command interpretation.

Command lines must contain a space or blank between the actual command, and any parameters supplied.

All values are in hexadecimal notation unless otherwise noted. Leading zeroes are not required, except to enter an edited line into the current file.

"Addr1" and "Addr2" are abbreviations for execution or starting and ending addresses. These parameters must be supplied unless otherwise noted.

Commands A, D, L, P, and X may be temporarily stopped by the line-feed (or CONTROL-J) key. RETURN continues execution and may be aborted by CONTROL-C.

Further information on the assembler used in XEK can be found in the "8080 Assembly Language Handbook" from the INTEL Corp. NOTE: The assembler in XEK does not support the definition of macros, the operations of multiply, divide, SHR and SHL. A colon (:) after a label is not required but may be included. Comment lines may begin with either an asterisk (*) or a semicolon (;). Comment fields after an operand need only a single space following the operand.

NOTE: The INTEL "8080 Assembly Language Handbook" is available from the Byte Shop of Westminster for $5.00 plus $1.50 postage and handling (California resident's sales tax included).

COMMAND EXPLANATIONS

A (SSEMBLE) Addr1 ↵

Starts assembly of current source file, starting at address specified as the origin.

A (SSEMBLE) Addr1 Addr2 ↵

Assembles current source file starting at Addr1 for origin, and locates object code at Addr1 and Addr2.

AE (RROR) Addr1 ↵

Assembles current file, origin at Addr1, and only displays lines containing errors.

AE (RROR) Addr1 Addr2 ↵

Assembles current source file, using Addr1 as origin, storing object code at Addr1 and Addr2. Only lines containing errors will be displayed.

A;D ↵

Displays current symbol table.

A;S ↵

Sets current symbol table to saved status for use with additional assembly

A;K ↵

Sets current symbol table to unsaved status to be cleared for next assembly. This is a default state.

B (REAKPT) Addr1 ↵

Restores any previous breakpoint, and sets new breakpoint to Addr1 Breakpoints are implemented by a RESTART Ø (RST Ø), and by storing a

jump to the breakpoint routine at address zero. The breakpoint is restored upon execution. NOTE: Using the breakpoint without RAM at ØØØØ to ØØØ2 will crash XEK.

C (ONC) INUM ↵

Reads in a source file from cassette and appends it to the current file. The value of INUM sets the line numbers increment for the concatenated file; e.g., if INUM = 1Ø, then whole file is renumbered from ØØ1Ø in decimal increments.

D (UMP) Addr1 Addr2 ↵

Displays in hexadecimal the contents of memory. Format is 16 bytes across, with hex address at start of each line. Default Addr1 is ØØØØ, default Addr2 if FFFF. Display may be controlled using linefeed, return, and CONTROL-C.

E (NTER) Addr1 ↵

Displays the address then the contained data-byte followed by a period. This value may be changed by inputting a value in hex, until the last two digits are correct. A space will display the next byte. A RETURN will display the address and next byte for modification. This command is aborted by CONTROL-C.

F (ILE) ↵

Displays current file parameters in following format:

Name Start-Addr End-Addr Last-Line-Number

F (ILE) "NAME" ↵

Saves parameters of current file into table, gets named file, and displays its parameters.

F (ILE) "NAME" Addr1 ↵

Creates new file at specified address, and saves it as "NAME" in file look-up table. This also makes this the current file. File name must be five characters or less.

F (ILE) "NAME" Ø ↵

Deletes named file from table, and makes NO file current. NOTE: No file may start at location ØØØØ. L (IST) or P (RINT) with no current file causes unpredictable results.

FS (TRUCT) ↵

Displays contents of file table in usual format. All null files shown as a a dash ("—").

G (OTO) Addr1 ↵

Begins execution of program at specified address.

H (EX) ↵

Reads INTEL hex-format paper tape into memory. Load is aborted with CONTROL-C. NOTE: The code at 3735H must be changed to conform with the reader input routine. 3736H and 3737H are the low and high address for this I/O and must match your reader input call address.

I (NPUT) Addr1 LNG ↵

Reads Tarbell format cassette into memory starting at address Addr1, of length LNG. Both values are hexadecimal.

J (UMP) Addr1 ↵

Restores all registers saved by breakpoint, sets new breakpoint to Addr1, and restores execution from last breakpoint.

K (ILL) LN1 LN2 ↵

Kills lines from current source file starting at LN1 to LN2 inclusive. If LN2 is absent, only one line is deleted. NOTE: Line numbers are decimal.

L (IST) LN1 ↵

Displays current file exactly as entered. Controlled by linefeed, RETURN, and CONTROL-C. If LN1 is present, listing starts on that line. NOTE: Line numbers are decimal.

M (OVE) Addr1 ↵

Moves current file to address Addr1, and displays new parameters of the file. Care must be exercised not to overlay the file during the move.

M (OVE) AddrS AddrE AddrR ↵

Moves block of memory from starting address (AddrS) to ending address (AddrE) to new starting address (AddrR). NOTE: Value for AddrS must fall on a hex 1ØØ boundary.

N (UMBER) INCR ↵

Renumbers current file beginning with the initial value in decimal, with increments of INCR in decimal.

O (UTPUT) Addr1 LNG ↵

Outputs a Tarbell cassette file, starting with address Addr1 of length LNG.

P (RINT) LN1 ↵

Displays entire current file in source listing format. If LN1 value is present, display will start with that line. NOTE: Line numbers are decimal.

Q (UIT) ↵

Exits to North Star D.O.S. at 2Ø28 hex.

R (EAD) Addr1 ↵

If Addr1 is not present, inputs XEK or ESP-1 format cassette source tape. If checksum is correct, outputs file parameters. If Addr1 is present, input object file resides at that starting address upwards; if checksum is correct, lists ending address.

S (ET) NUM ↵

Sets autoline editor to increment last line number by value of NUM in decimal.

T (EST) AddrS AddrE ↵

Tests memory beginning with starting address AddrS to ending address AddrE. Errors displayed as : Addr V1 V2 ERROR where Addr is the address of the error, V1 is the value of byte written, and V2 is the value of the byte read back.

U (NYANK) "NAME" ↵

Tests current file for correct format, locates diskette file "NAME," tests file for type eight, determines if file size is sufficient to hold the current file (XEK prints "--- spills" and lists the directory if not), writes out current file to disk file "NAME", and displays current file parameters on completion.

V (ERIFY) AddrS AddrE AddrR ↵

Verifies memory beginning with address AddrS (with a boundary at a 1ØØ hex value) to ending address AddrE with the data at address AddrR, and upward. Error format is the same as command T, with errors referenced to the block of memory described by AddrS to AddrE.

W (RITE) Addr1 Addr2 ↵

When Addr1 and Addr2 are absent, outputs entire current file to cassette in XEK format, and prints "WRITTEN" on terminal when done. If Addr1 and Addr2 are present, this command writes block of memory from Addr1 to Addr2 to cassette. Prints "WRITTEN" on terminal when through.

X LN1 ↵

Lists as in L command, but without line numbers. If LN1 is supplied, listing starts with that line number. If no file is current, you may expect unpredictable results.

Y (ANK) ↵

Lists diskette directory.

Y (ANK) "NAME" ↵

Locates diskette file "NAME" (prints "NO" and the diskette directory if unable to locate), tests for type eight file, reads entire file into memory beginning with current file's starting location, tests resulting file for correct format, and displays current parameters.

Y (ANK) "NAME" INCR ↵

As above, but appends to current file, and renumbers by INCR. Parameters not displayed.

Z (ERO) Addr1 Addr2 B1 ↵

Fills memory from Addr1 to Addr2 with value of B1 in hex. If B1 is absent, zero is used.

: "NAME" Addr1 ↵

If only : is used, XEK performs a scan on current file for correct format, indicates the address at which first error occurs (if any), and

displays file parameters. With "NAME" and Addr1, the : command creates a file on "NAME" at address Addr1, then scans for format errors. e.g.; Assume it is desired to rename "FILE1" at location Addr1 to "FILE2". The following command sequence does this:

F "FILE1" Ø ↵ (deletes file)

: "FILE2" Addr1 ↵ (renames the file)

= VAL1 VAL2 ↵

Sums (VAL1 + VAL2) and differences (VAL1 - VAL2) and (VAL2 - VAL1) are displayed for hex arithemetic.

? ↵

This command initializes XEK's RAM area and clears file table. The files themselves are not destroyed, and may be recovered with the : "NAME" Addr1 command.

AUTOLINE OPERATIONS

The autoline function takes the last line number supplied in the parameters list of the file table and increments it by the value of the last S (ET) command. The resulting number is placed in the input buffer as well as echoed to the terminal just as if it had been keyed in as an input. Thus, it may be deleted or replaced just as any other input is.

Autoline is invoked by pressing the space bar as the first character in a command line. The increment may be reset at anytime by the S (ET) command.

DIS COMMAND SUMMARY

COMMAND	ACTION
1	Creates RAM file one byte at a time
2	Creates RAM file one address (2 bytes) at a time
3	Creates RAM file by alternating 1 and 2
:	Sets or displays file parameters
?	Clears symbol table and file address, and then resets line counter
A	Displays address and 16 ASCII characters per line
B	Bias (offset) disassembly for program not at correct execution address
C	Cross-references symbol table
F	Finds and generates symbol table
G	Goes to and executes program at given address
H	Hex dump of address then 16 bytes of data
L	Creates RAM file in list format with symbolic instructions
P	Displays file in formatted list
Q	Quits and executes North Star D.O.S.
S	Sets line number increment and starting line number
T	Displays symbol table addresses
V	Views symbolic instructions with hex and ASCII data

NOTE: "RUB" deletes character and backspaces cursor. 1,2,3,A,H,L,P, T, and V are stopped by linefeed, continued with return and aborted with CONTROL-C.

All commands to DIS are executed only after the complete command line is terminated by a carriage return, indicated by a ↵ in this documentation.

Command lines contain embedded blanks, exactly as indicated, between the command and any parameters, optional or otherwise.

All values are in hexadecimal notation unless otherwise noted. Leading zeroes are not required for any command.

"Addr" is an abbreviation for address, usually followed by a symbol specifying what the address is for.

Commands 1,2,3,A,H,L,P,T, and V are stopped by the LINE-FEED (CONTROL-J) key; restarted by RETURN, and aborted by CONTROL-C.

Commands 1,2,3, and L will create a RAM file in memory only if the file starting address is not zero.

1 Addr1 Addr2 ↵

Displays hex data one byte per line in the format of the psuedo-op "DB" format for the area from address "Addr1" to address "Addr2" inclusive. If file address specification is non-zero, a RAM file is generated.

2 Addr1 Addr2 ↵

As for 1 above, but displays in "DW" pseudo-op format. File generation still dependent on the file address being non-zero.

3 Addr1 Addr2 ↵

Alternate commands "1" and "2" on a line-by-line basis, with file generation still dependent on the file address being non-zero.

: ↵

Displays starting and ending address of RAM "FILE." If the address is zero, no file will be generated.

ORDER SHEET

Additional copies of XEK can be ordered from the Byte Shop of Westminster for $48.00 each (including the diskette and the manual). First class postage, handling, and California resident's sales tax is included.

NAME _____

STREET NO. _____

CITY _____

STATE _____ ZIP _____

QUANTITY _____ at $48.00 each = _____
amount enclosed

Mastercharge, Bank of America, and VISA cards accepted.

CARD NO. _____

SIGNATURE _____

Please allow two weeks for check to clear and two weeks for delivery. Special assemblys available on request at no extra charge.

MAIL TO: Byte Shop of Westminster
14300 Beach Boulevard
Westminster, CA 92683
714/894-9131

: Addr1 ↵

Sets file starting address to value of Addr1. NOTE: Only commands 1,2,3, and L produce RAM files. Both forms of the : command reset the line counter.

? ↵

Clears symbol table, file address and bias value. Resets line number counter. Outputs "SYMBOL TABLE CLEARED."

A Addr1 Addr2 ↵

Displays address of up to 16 ASCII characters per line from address "Addr1" to address "Addr2" inclusive. This display is formatted so that addresses ending in "Ø" commence new lines.

B Addr1 Addr2 ↵

Bias (offset) disassembly for programs loaded at addresses other than their regular execution position. This bias is cleared by the ? command. Addr1 is where the program is loaded, and Addr2 is where it normally resides.

C Addr1 Addr2 Sym ↵

Cross-reference to the symbol table, checking for referents from address "Addr1" to address "Addr2" inclusive, starts with symbol "Sym;" if a value is absent, commences with the first symbol in the table.

F Addr1 Addr2 ↵

Finds and generates a symbol table of references within area covered from address "Addr1" to address "Addr2" inclusive.

G Addr1 ↵

Goes to and executes program at address "Addr1."

H Addr1 Addr2 ↵

Displays address, then up to 16 bytes for hex data per line from address "Addr1" to address "Addr2 inclusive. Format of lines is such that addresses ending in Ø commence new lines.

L Addr1 Addr2 ↵

Displays memory contents with decimal line numbers, labels with a leading "H," mnemonics and two byte data with leading "H" to reference labels or one byte data leading "Ø" and trailing "H" as a constant. Strips all unnecessary blanks. If the file address is non-zero, creates RAM file in memory.

P Addr1 Addr2 ↵

As for the L command, but with column justified output, with no RAM file generated.

Q ↵

Quits and executes North Star D.O.S. at 2Ø28H.

S Incr Strt ↵

Sets decimal line number counter to increment by the value of "Incr" (decimal) and begins with the value of "Strt" (decimal).

T ↵

Lists symbol table sequentially, as eight hex address per line.

V Addr1 Addr2 ↵

Views disassembly with hex addresses, hex code, ASCII equivalent, label, mnenomic, and hex arguments justified into columns. This is the primary "quick-look" command.

BYTE SHOP

The Affordable Computer Store

SAMPLE ASSEMBLY USING XEK

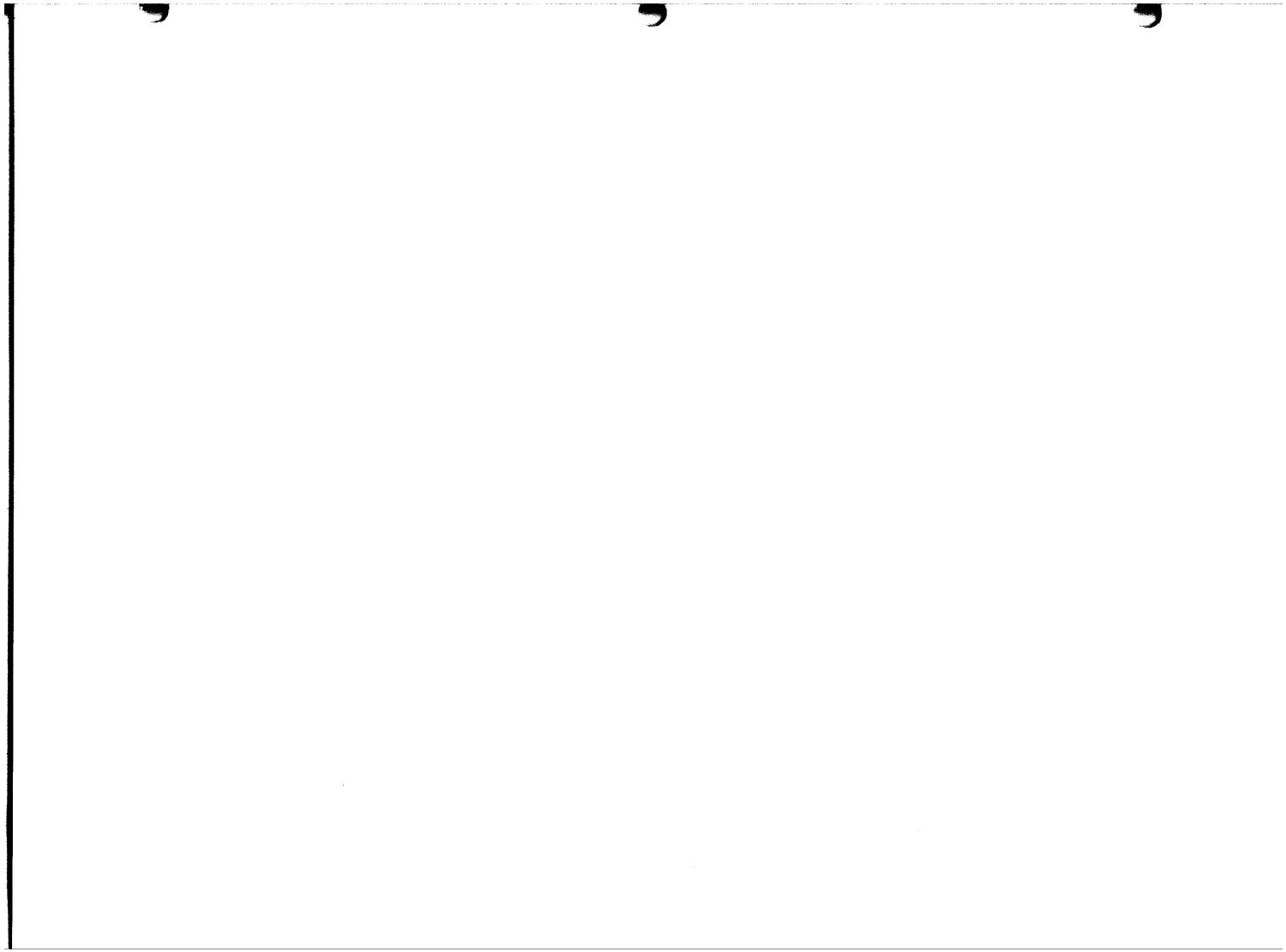
```
7000                    0010 *PAPER TAPE LOAD RTN FOR OLIVER
7000                    0020 * WRITTEN FOR THE BYTE SHOP OF WESTMINSTER
7000                    0030 *COPYRIGHT (C) 1977 BY R.D. PARKS
7000                    0050 *
7000 CD 3A 70           0100 READ   CALL  TTYIN
7003 CD 1F 70           0110        CALL  CHAR
7006 57                 0120        MOV   D,A
7007 C8                 0130        RZ
7008 CD 1F 70           0140        CALL  CHAR
700B 67                 0150        MOV   H,A
700C CD 1F 70           0160        CALL  CHAR
700F 6F                 0170        MOV   L,A
7010 CD 1F 70           0180        CALL  CHAR
7013                    0200 * HERE WE LOOP FOR DATA FROM READER
7013 CD 1F 70           0220 LOOP   CALL  CHAR
7016 77                 0230        MOV   M,A
7017 23                 0240        INX   H
7018 15                 0250        DCR   D
7019 C2 13 70           0260        JNZ   LOOP
701C C3 00 70           0270        JMP   READ
701F                    0290 * CHARACTER LOOP
7020 CD 3A 70           0310 CHAR   CALL  TTYIN
7022 CD 32 70           0320        CALL  HEX
7025 07                 0330        RLC
7026 17                 0340        RAL
7027 17                 0350        RAL
7028 17                 0360        RAL
7029 5F                 0370        MOV   E,A
702A CD 3A 70           0380        CALL  TTYIN
702D CD 32 70           0390        CALL  HEX
7030 83                 0400        ADD   E
7031 C9                 0410        RET
7032                    0430 * DO HEX SUBTRACTION
7032 D6 30              0450 HEX    SUI   48
7034 FE 0A              0460        CPI   10
7036 D8                 0470        RC
7037 D6 07              0480        SUI   7
7039 C9                 0490        RET
703A                    0510 * DO ACTUAL READ, AND DISPLAY ON CONSOLE LIG
HTS
703A DB 03              0530 TTYIN  IN    03
703C E6 40              0540        ANI   64
703E CA 3A 70           0550        JZ    TTYIN
7041 DB 01              0560        IN    01
7043 D3 01              0570        OUT   01
7045 D3 FF              0580        OUT   0FFH
7047 E6 7F              0590        ANI   127
7049 C9                 0600        RET

SYMBOL TABLE

CHAR  701F    HEX   7032    LOOP  7013    READ  7000    TTYIN 703A
```

BYTE SHOP

The Affordable Computer Store

SAMPLE DISASSEMBLY USING DIS

```
7000   CD 3A 70      :P        CALL   703A
7003   CD 1F 70        P       CALL   701F
7006   57            W         MOV    D,A
7007   C8                      RZ
7008   CD 1F 70        P       CALL   701F
700B   67            G         MOV    H,A
700C   CD 1F 70        P       CALL   701F
700F   6F            O         MOV    L,A
7010   CD 1F 70        P       CALL   701F
7013   CD 1F 70        P       CALL   701F
7016   77            w         MOV    M,A
7017   23            #         INX    H
7018   15                      DCR    D
7019   C2 13 70        P       JNZ    7013
701C   C3 00 70        P       JMP    7000
701F   CD 3A 70      :P        CALL   703A
7022   CD 32 70      2P        CALL   7032
7025   07                      RLC
7026   17                      RAL
7027   17                      RAL
7028   17                      RAL
7029   5F                      MOV    E,A
702A   CD 3A 70      :P        CALL   703A
702D   CD 32 70      2P        CALL   7032
7030   83                      ADD    E
7031   C9                      RET
7032   D6 30         0         SUI    30
7034   FE 0A                   CPI    0A
7036   D8                      RC
7037   D6 07                   SUI    07
7039   C9                      RET
703A   DB 03                   IN     03
703C   E6 40         @         ANI    40
703E   CA 3A 70      :P        JZ     703A
7041   DB 01                   IN     01
7043   D3 01                   OUT    01
7045   D3 FF                   OUT    FF
7047   E6 7F         %         ANI    7F
7049   C9                      RET
```