

KMR-100

INSTALLATION AND OPERATORS GUIDE

for the

HEATH/ZENITH 88, 89 AND 90 DIGITAL COMPUTERS

(c) 1983
KRES Engineering
P.O. Box 1268
LaCanada, CA 91011

Contents of ROM written by and
Copyright 1983 by Ultimeth Corporation

LIMITED 90 DAY WARRANTY

KRES Engineering warrants this product to be free from defects in materials and/or workmanship for a period of 90 days.

In the event of a malfunction or other indication of failure attributable directly to faulty workmanship and/or material, KRES will, at its option, repair or replace the defective part or parts to restore this product to proper operating condition. Return of this product is subject to the issuance of a RETURN MERCHANDISE AUTHORIZATION NUMBER by KRES engineering. This RMA number must be clearly marked on the outside of the package. Return the product postage paid to KRES Engineering at P.O. Box 17328, Irvine, California 92713 "Attention Warranty Claims Department". All repairs and/or replacements shall be rendered by KRES without charge for parts or labor when the product is returned within the specified period of the date of purchase. This warranty applies only to the original purchaser.

This warranty will not cover the failure of KRES products which at the discretion of KRES, shall have resulted from accident, abuse, negligence, alteration, or misapplication of the product. While every effort has been made to provide clear and accurate technical information on the application of KRES products, KRES assumes no liability in any events which may arise from the use of said technical information.

Our warranty does not cover and we are not responsible for damage caused by misuse or fire or unauthorized modifications to or uses of our products for purposes other than advertised. Our warranty does not include reimbursement for customer assembly, disassembly, set-up time, or unauthorized repairs.

This warranty is in lieu of all other warranties, expressed or implied, including warranties of mercantability and fitness for use. It is not extended to allied equipment or components used in conjunction with these products. IN NO EVENT WILL KRES ENGINEERING BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING FROM OR IN ANY WAY CONNECTED WITH THE USE OF ITS PRODUCTS. Some states do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

IMPORTANT: Proof of purchase necessary for products returned for repair under warranty. Before returning any product please call our customer service department for a return authorization number.

TABLE OF CONTENTS

Introduction	1
Monitor ROM installation.....	2
Setting of SW-501.....	4
Monitor commands.....	6
Boot.....	6
Check:	
DISK (rotational speed).....	8
RAM (memory test).....	8
Exit to HDOS.....	9
Go.....	10
HELP.....	11
Boot Help.....	11
In.....	12
KRES:	
Enable (EXP-700).....	13
CPU Speed (DSM-240).....	14
System Speed (DSM-240).....	14
Memory.....	15
Number Base.....	17
Out.....	17
Register.....	18
Step.....	19
View.....	20
Appendix A (Pictorial 1).....	21
Appendix B (Use with Non-Heath/Zenith)	
Booting.....	22
Replacement terminal ROMs.....	23
Appendix C (Technical Information).....	24
Appendix D (Optional hardware mod for HDOS).....	25
Appendix E (Heath Compatibility).....	27
Appendix F (HDOS Patches for Hex Display)	30
Appendix G (Ultimeth Copyright and Policy).....	34



INTRODUCTION

The KMR-100 is a replacement monitor ROM for the Heathkit H89 computer that provides many additional features for any H/Z89-90 owner. Owners of KRES Engineering hardware will find commands that support their hardware. The HDOS user will find increased HDOS processing speed and additional debugging capabilities. The following features are provided:

NOTE THAT SOME OF THESE FEATURES ARE AVAILABLE ONLY ON THE KMR-100 SOLD BY KRES ENGINEERING AND DO NOT APPEAR ON ANY OTHER ULTIMETH MONITOR ROM SOLD BY OTHER VENDORS.

You may toggle CPU and System speed directly from the Monitor ROM with the KRES DSM-240 hardware. This feature is particularly useful for some non Heath/Zenith operating systems where some drives MUST be accessed at the speed the system is booted.

You may phantom boards in or out of your system (Shadow Operation) with the KRES EXP-700 hardware. You may keep several conflicting disk controller boards in your system and choose the one(s) you wish to use before you boot.

The monitor is a 4K byte 4 MHz part that supports CPU speeds of 2 or 4 MHz, including booting from an H-17, H-37, H-47, H-67, Magnolia Microsystems' double density floppy controller board, Livingston Logic Labs 8" single density floppy controller board, CDR's double density floppy controller board, and Corvus hard disk; and from a pseudo disk in additional RAM memory, all at either 2 or 4 MHz.

The monitor automatically detects the terminal baud rate including 19,200 Baud.

A HELP feature lists all available monitor commands.

Full compatibility with HDOS and Heath and Magnolia CP/M has been maintained. Each ROM retains all of the common Heath H89 ROM entry point locations (except for cassette tape processing).

The clock value in memory (TICCNT) has been extended to a four byte counter.

The dynamic RAM test has been updated to provide faster, better testing; the disk drive test now correctly measures the rotation time of any mini-floppy unit.

The monitor clock interrupt processing routine has been rewritten, giving an effective 16% CPU speed increase to HDOS programs running on a 2 MHz system, and a 7% increase to HDOS programs running on a 4 MHz system.

The monitor supports debugging of programs under HDOS. Additional monitor debugging commands have been provided to display and alter all Z-80 registers, to read and write to I/O ports, to single step and breakpoint programs, and to restart HDOS. With an optional hardware modification, the monitor will allow the RESET key to interrupt a program.

INSTALLATION:

Be sure to follow **ALL** of the steps in this section when installing your new monitor ROM. Any deviations from or alterations to these procedures may result in damage to the monitor ROM or the computer system.

() **Disconnect the computer line cord from the AC outlet.**

() Remove the top shell of your computer. This procedure is detailed in both the DSM-240 manuals and EXP-700 manuals. Consult the appropriate sections in those manuals should you have problems.

BEFORE PROCEEDING MAKE SURE THAT YOU HAVE REMOVED THE LINE CORD FROM THE AC OUTLET TO PREVENT HAZARD OF ELECTRICAL SHOCK

Refer to PICTORIAL 1 for the following steps. It should be possible to install the KMR-100 without having to pull the CPU card from your system. It may make installation easier if you remove any accessory cards in the left hand bus. If you are also installing the EXP-700 I/O expansion, or the DSM-240 Dual speed module at the same time, it is easiest to install the KMR-100 while you have the CPU card out of the system as required to install the other hardware.

() Without removing any IC's yet, locate the old monitor ROM (Heath part number 444-40, 444-62, 444-84, or 444-142) at U518. If the part is marked 444-84 or 444-142 (MTR-90), you may skip directly to PART TWO. If the part is marked 444-40 or 444-84, or your ROM is from another vendor, you **MUST** make two other checks detailed in PART ONE to your CPU card before replacing the ROM.

PART ONE

Follow these steps if your system did NOT presently have the MTR-90 monitor ROM from Heath/Zenith (part number 444-84 or 444-142).

() Check the part number marked on the IC at U516 (see pictorial 1). If it is marked 444-83, you have the correct part and can proceed to the next step. If the part is marked 444-41, you must remove the old part and install a new 444-83. This IC is NOT furnished with the KMR-100 and is available directly from Heath/Zenith. (You will also need to order the part in the next step.) Note the alignment of the notch on the old part before you remove it, and check the new part for bent under or missed pins once it has been installed.

() Consult pictorial 1 to find the jumpers in a vertical row just to the left of the monitor ROM at U518. These will be labeled JJ505, JJ506, JJ507 and JJ508 on older units; or JJ504, JJ505, JJ506 and JJ507 on newer units. The center pin of the **TOP** jumper (JJ506 or JJ505 on older and newer units, respectively) should have a wire cable installed with the other end connecting to pin 14 of P508. This cable is NOT furnished with the KMR-100 and is available directly from Heath/Zenith as part number 134-1159. Install the new cable and verify the wire goes between the **CENTER** pin of the **TOP** jumper and pin 14 of P508.

PART TWO

CAUTION: NORMAL STATIC ELECTRICITY DISCHARGED FROM YOUR BODY THROUGH ANY PIN ON THE NEXT IC TO AN OBJECT CAN DAMAGE THE IC. WHEN YOU REMOVE THE OLD IC, PLACE IT BACK INTO THE PIECE OF FOAM SHIPPED WITH YOUR NEW MONITOR ROM.

() On the CPU board, locate and note the orientation of the old monitor ROM at U518; note that the notch in the ROM is toward the center of the CPU board. Remove the old ROM from the socket at U518. Carefully remove the new ROM from its conductive foam and immediately install it in the socket at U518; orient the notch in the ROM toward the center of the CPU board. Check to be certain no pins either missed the socket or accidentally bent under the IC. Save the old ROM in the in the conductive foam so that you will be able to go back to it in the future if you wish.

() Remove any IC that may be installed at U519, immediately below the ROM you just installed in the previous step.

() Check the settings on the row of jumpers just to the left of the Monitor ROM you just installed. They will probably require changing if you had a monitor ROM other than MTR-90 installed (i.e. other than 444-84 or 444-142). These should be as follows:

Older Units	Newer Units	Proper setting for Jumper
-----	-----	-----
JJ505	JJ504	"1" (down)
JJ506	JJ505	Jumper to pin 14 of P508
JJ507	JJ506	"1" (down)
JJ508	JJ507	"1" or "B" (down)

() If you removed any left-hand bus accessory boards, re-install them now.

This completes the first part of the installation of the monitor ROM. You will also need to alter the settings of switch SW501 on the CPU board; these settings are detailed in the next section.

SETTING OF SW501

See PICTORIAL 1 for the location of SW501 on your CPU card. The definitions of the settings of the switch sections on SW501 are different than for the Heath ROMs (terminal baud rate speed detection is automatic), and are as follows:

"1"	"0"	"1" TO THE LEFT	"0" TO THE RIGHT
0	-		
1	\		
2	/		
3	-		
4	-		Section 4 to map ROM into RAM
5	-		Section 5 for auto memory check
6	-		Section 6 for LLL controller
7	-		Section 7 for auto-boot

SWITCH SECTION 7:

0 Normal monitor boot operation.

1 Upon reset (at power-up or when "SHIFT - RESET" is depressed), a boot (called an "autoboot") is made from the default boot drive (see below), unit 0.

SWITCH SECTION 6:

0 Normal monitor operation.

1 This setting must be used if you have installed a Livingston Logic Labs 8" single density floppy controller.

SWITCH SECTION 5:

1 Normal monitor operation. This setting must be used when this switch is wired in parallel with an "external interrupt" switch as described below under "Optional Hardware Modification".

0 Upon reset (at power-up or when SHIFT - RESET is depressed), the "dynamic RAM test" is performed.

SWITCH SECTION 4:

0 Normal monitor operation. On boot, the lower 8K of ROMs are moved to RAM, and memory is remapped to all RAM. On an restart, memory is remapped to all RAM.

1 The lower 8K of ROMs are not moved to RAM. This setting must NOT be used if you boot from, or use under HDOS, a Livingston Logic Labs 8" single density floppy controller or a CDR double density floppy controller.

SWITCH SECTIONS 3-0:

Switch sections 3-0 control the default boot device type. The following table lists the boot devices available.

Switch Section	
3 2 1 0	
0 0 0 0	H-17 hard-sectored 5" floppy, units 0-5 (see note #1).
0 0 0 1	H-37 soft-sectored 5" floppy, units 0-3.
0 0 1 0	Corvus hard disk/Magnolia interface, partitions 0-8.
0 0 1 1	CDR 5"/8" double density floppy, units 0-3.
0 1 0 0	H-47 8" floppy at port 78H/170Q, units 0-3 (see note #2).
0 1 0 1	H-47 8" floppy at port 7CH/174Q, units 0-3 (see note #2).
0 1 1 0	Null device (reserved for future boot devices).
0 1 1 1	"HELP" device (null) - provides a list of boot devices.
1 0 0 0	Null device (reserved for future boot device).
1 0 0 1	SASI controller or H-67 disk at 78H/170Q, units 0-7 (see note #2).
1 0 1 0	SASI controller or H-67 disk at 7CH/174Q, units 0-7 (see note #2).
1 0 1 1	Livingston 8" singl. den. floppy, units 0-3.
1 1 0 0	Magnolia 5"/8" dbl. den. floppy, units 0-7. Units 0-3 are 8 inch Units 4-7 are 5 1/4 inch
1 1 0 1	Null device (reserved for future boot devices).
1 1 1 0	Null device (reserved for future boot devices).
1 1 1 1	Magnolia 128K aux. memory board pseudo disk, banks 0-1.

Notes:

1. This monitor ROM will support units 0-5 with the H-17, if the H-17 controller board is properly modified. Information on the modification of the H-17 controller board is available from:

Gregory J. Marsh 703/590-3360
5279 Miles Court
Woodbridge, VA 22193

2. If you boot Heath CP/M 2.2.03 from a floppy with this controller, you must set switch sections 0-3 as indicated (as required by the Heath documentation); you will thus lose the capability to have the switches specify a default boot device of your choice.

This completes the installation of the KMR-100 and the configuration of your CPU card switches and jumpers. While the cover is still off your machine, and, if you were installing other hardware from KRES and have the CPU card out, we suggest you read the section on OPTIONAL HARDWARE MODIFICATION in appendix B to see if this HDOS debugging tool would be of interest to you. If you decide to make the modification, it is best to reinstall enough of your machine to verify that the new Monitor ROM is functioning properly before making yet another modification.

Replace the top shell of the computer cabinet. Turn on power to the H89; you should immediately hear one "beep"; most machines will "beep" twice. The prompt "KE:" should appear on the CRT screen. If not, turn off the power and make sure that the new ROM is properly installed at socket U518, and that you did not jar any cables loose.

MONITOR COMMANDS:

There are several commands you may type to the monitor to boot and check disks, to check system RAM, examine and change memory locations and registers, debug programs and more. Also included are commands that support some of the KRES Engineering hardware: the EXP-700 and DSM-240. These commands are explained on the following pages.

Values shown below in the examples are hexadecimal; operation is the same for split octal. When typing input to the monitor, letters (except for register names in the "Register" command) may be typed in either upper or lower case. When typing a command word, only the first letter of the command word is typed; the computer displays the entire word. If the entire command may be determined from the letter typed, the computer displays the entire command; otherwise, in the same manner type the first letter of the next word of the command. At any point prior to ending the line by typing the RETURN key (and, for some commands, a space, minus sign, or equal sign), a line may be cancelled and the command terminated by typing the DELETE key. When typing in a split octal or hexadecimal value, errors may be corrected by retyping the entire correct value; the monitor uses only the last digits typed.

In the examples given below, the following conventions are used:

<cr> shown where the RETURN key would be typed

<delete> is shown where the DELETE key would be typed.

<space> is shown where the space bar would be typed

() letters enclosed in parentheses are automatically filled in by the computer.

Boot

This command reads the bootstrap code from the boot track into RAM memory beginning at address 42.200 (split octal) or 2280 (hexadecimal), and then transfers control to the bootstrap code at the above address. The default boot device is unit 0, controller 0 (for SASI devices), of the default boot device type designated by switch settings on the CPU board.

Any other device type may also be booted by typing the device type letter after the "B" (see table on next page). You may then type an optional controller number (for SASI devices) and an optional unit number; the last digit typed (if any) becomes the unit number, and the next to last digit typed (if any) becomes the controller number. For certain boot devices (such as the H-67), you may enter an optional "command string" by then typing a slash ("/") and the "command string". Then type the RETURN key.

If the read of the boot track is not complete after 16 seconds, or if an I/O error occurs, the boot is aborted with the message "Boot error". The read of the boot track may also be aborted by depressing the DELETE key (the DELETE key will also abort an autoboot).

The following table lists the boot devices available. The letter in "[]" is used to identify the device type in the boot command.

[A]	H-17 hard-sectored 5" floppy, units 0-2 (see note #1).
[B]	H-37 soft-sectored 5" floppy, units 0-3.
[C]	Corvus hard disk/Magnolia interface, partitions 0-8.
[D]	CDR 5"/8" double density floppy, units 0-3.
[E]	H-47 8" floppy at port 78H/170Q, units 0-3 (see note #2).
[F]	H-47 8" floppy at port 7CH/174Q, units 0-3 (see note #2).
[H]	"HELP" device (null) - provides a list of boot devices. See HELP section for details.
[J]	SASI controller or H-67 disk at port 78H/170Q, units 0-7 (see note #2).
[K]	SASI controller or H-67 disk at port 7CH/174Q, units 0-7 (see note #2).
[L]	Livingston 8" sngl. den. floppy, units 0-3.
[M]	Magnolia 5"/8" dbl. den. floppy, units 0-7.
[P]	Magnolia 128K aux. memory board pseudo disk, banks 0-1.

EXAMPLES:

In all of the following examples, the computer will prompt you with KE: before you begin to type your command.

1. You want to boot the H17 drives. You type:

```
B(oot )A(: ) <cr>
```

and your computer will begin to boot the 5 1/4 inch 10 hard sector drives. Since no unit number was specified, it assumes unit 0.

2. You want to boot unit 1 of the H17 drives. You type:

```
B(oot ) A(: )1 <cr>
```

3. You want to boot unit 0 of the default boot device. The default boot device is the unit you set using switch sections 3-0 in the "Setting of SW501" section of this manual on page 5:

```
B(oot ) <cr>
```

4. You want to boot unit 1 of the H67, and pass a command string or "Hi" to the bootstrap code. Letter "K" is assumed to correspond to your SASI controller. (Check table at top of this page):

```
B(oot )K(: )1/Hi <cr>
```

5. You want to boot unit 3 of SASI controller 2. For this example, device type letter "J" is assumed to correspond to your SASI controller. (Check table at top of this page):

```
B(oot )J(: )23 <cr>
```

CHECK

The CHECK command provides a method of checking parts of your system. There are two subcommands:

1. DISK (CHECK DISK) measures disk rotational speed
2. RAM (CHECK RAM) does a memory check

Each of the commands requires two letters to be typed. "C" will autocomplete as Check, and your computer will wait for either a "D" for disk check, or R for RAM check; each command autocompleting as Disk or RAM, respectively. The check will begin when a carriage return is typed.

Check Disk:

This command replaces the function of the JMP instruction at 7.372 (split octal) or 7FA (hexadecimal) in the original Heath ROMs. Insert any diskette in a mini-floppy disk drive and type a "C" followed by a "D", an optional unit number, and a RETURN key. The diskette is not altered; any initialized or uninitialized hard-sectored (10 sectors per track) diskette may be used. The display will show the word "Rotation" flashing, and, if the disk drive is within tolerance (1%), the display will show a number between 1373 and 2005 (octal) or 1FB and 205 (hexadecimal). The drive may be adjusted to show a number as close to 2000 (octal) or 200 (hexadecimal) as feasible. This command is terminated by depressing the DELETE key.

EXAMPLE:

You want to test the rotational speed of disk unit 1. At the KE: prompt, you type:

```
C(heck )D(isk )1 <cr>
```

Your computer responds with:

```
Rotation time: nnnn
```

where nnnn is a number corresponding to the rotation speed of your drive (see above). When you are done with the test, type:

```
<delete>
```

and your computer will respond with the KE: prompt again.

Check RAM:

This command replaces the function of the JMP instruction at 7.375 (split octal) or 7FD (hexadecimal) in the original Heath ROMs. Type a "C", a "R", and the RETURN key; the display will show the address limits of the dynamic RAM tested, and will show a varying "Mask:" value. The mask value is used along with the address bits to generate a test value of either AA or 55 (hexadecimal) for each byte in memory.

This test is very effective because it tests for one data bit interfering with another data bit (crosstalk) both within a byte (circuit board capacitance crosstalk) and between bytes at different memory addresses (memory chip matrix crosstalk). The test uses 16 mask values, beginning with AB (hexadecimal), and then repeats. An error is indicated by the upper limit RAM address being incorrect, or by an error message. The error message will show a memory location and a byte value; the error bit is a missing or extra bit in a value of AA or 55 (hexadecimal). This command is terminated by resetting the computer (keying SHIFT - RESET).

EXAMPLE:

You want to test your system's memory. At the KE: prompt, you type:

C(heck) R(AM) <cr>

Your computer will respond with:

2 MHz reset	(or 4 MHz)
Dynamic RAM test 2000 to nnnn	(nnnn indicates RAM size)
Mask: AB	(the AB will change)

When you are done with the memory test, you must reset the computer by holding down the right hand SHIFT key at the same time as the RESET key.

EXIT TO HDOS

This command is equivalent to a "Go 40100" (split octal) or "Go 2040" (hexadecimal). Type an "H" and the RETURN key; if HDOS is still in memory, HDOS will regain control and display the HDOS prompt.

EXAMPLE:

You have been running HDOS and you are suddenly back in the monitor, either through error, or intentional debugging. To return to HDOS, at the KE: prompt, you type:

E(xit to HDOS) <cr>

and your computer will reenter HDOS, assuming it is still in memory and undamaged. Note that the control is returned to the program at 2040 Hex, regardless of what is at that memory location, when this command is issued.

GO

This command transfers control to a program. If you type an address and the RETURN key after the "G", control passes to the program at that address. If you type just the RETURN key after the "G", control passes to the current address value in the user PC register.

EXAMPLES:

In all of the following examples, the computer will prompt you with the KE: before you begin to type your command.

1. You wish to begin execution of a program at 2280 Hex. You type:

G(o)2280 <cr>

and you will exit the currently running monitor segment and start execution at 2280 Hex.

2. You wish to transfer control to the program at the address specified by the current value of the program counter register. You type:

G(o) <cr>

and you will exit the currently running monitor segment and start execution at whatever address was in the program counter register (see the REGISTER command to examine the contents of this register).

HELP and BOOT HELP

This command displays all monitor commands and options. Help is available from two commands:

1. For a listing of all monitor commands, type an "H" and then the RETURN key. Commands are displayed; options are displayed indented under the relevant command.
2. Help for the "BOOT" command is obtained by typing "B" for BOOT followed by "H" for HELP. A help text will appear showing each boot device type letter, the number of devices for that boot type, and the name of the controller or interface board for that device type.

EXAMPLES:

1. You want a listing of all monitor commands. At the KE: prompt, you type:

```
H(elp ) <cr>
```

Your computer will respond with a list of all commands and options for all commands except the BOOT command:

Commands/options:

```
  Boot
  Check
          Disk
          RAM
```

and return with the KE: prompt.

2. You want a listing of all options for the BOOT command. At the KE: prompt, you type:

```
B(oot ) H(: ) <cr>
```

Your computer will respond with a list of all options for the BOOT command. Valid device letters are listed, together with the unit numbers supported and the corresponding hardware controller. Where the controller may be jumpered to different port assignments, an "@" appears together with the port:

Dev	Units	Cntlr
A	6	H17
B	4	H37
C	9	Corvus
D	4	CDR
E	4	H47 @78
F	4	H47 @7C
J	8	H67 @78
K	8	H67 @7C
L	4	LLL
M	8	MMS
P	2	RAM

and return with the KE: prompt.

IN

This command inputs a value from any port (references to an undecoded port address produce meaningless values). Type a port address and the RETURN key after the "I"; the computer will type the port address again and the input value. You then have the option of several subsequent commands:

YOU TYPE	ACTION TAKEN
<cr>	Repeats the input at the same port address
= (equal key)	Repeats the input at the same port address
SPACE bar	Increments the port address and repeats
- (minus key)	Decrements the port address and repeats
DELETE key	Terminates the input command

EXAMPLES:

You wish to input from port F2. At the KE: prompt, you type:

I(nput)F2 <cr>

Your computer does an input from the designated port F2, and displays the value:

00F2 20

You may now input from F1 by typing a "-" (minus sign):

-

Your computer does an input from port F1, and displays the value:

00F1 FF

To leave the IN command, press the DELETE key:

<delete>

and your computer will redisplay the KE: prompt.

KRES

The KRES command provides a method of controlling some of the KRES hardware. Support is included for the EXP-700 seven slot I/O expansion, and the DSM-240 Dual Speed Module. There are three subcommands:

1. ENABLE for Shadow Operation on the EXP-700
2. CPU for CPU speed change on the DSM-240
3. SYSTEM for CPU and Bus speed change on the DSM-240

Each of these commands requires two letters to be typed. "K" will autocomplete as KRES, and your computer will wait for either an "E" for enable, a "C" for CPU speed, or a "S" for System speed; each command autocompleting.

ENABLE

The ENABLE command provides a method of displaying and changing which of the four added slots of your expansion will be active. This command overrides the power-on/reset switch setting of SW9 sections 6-9. This ability is particularly useful if you have more than two types of floppy disk controllers in your system; since Heath/Zenith software supports only two, you can select which you want enabled before you boot. Any selection via this command can later be overridden by the software supplied with the EXP-700.

Type a "K" for KRES followed by an "E" for enable, then RETURN. Any boards currently enabled (A-D, A being the top slot) will be displayed. You then have two options:

1. Type RETURN or DELETE to leave those boards as they are.
2. Type a "/" followed by the boards you want enabled A-D, then RETURN. Note that by typing "/" followed by RETURN, all boards are disabled.

EXAMPLES:

You want to check and possibly change the boards enabled. At the KE: prompt you type:

```
K(RES )E(nable) <cr>
```

Your computer responds (we will assume boards A and D are enabled):

```
AD
```

To change the boards enabled to AB and C, you type:

```
/ ABC <cr>
```

And now the top three slots, AB and C are enabled. We can check this by typing:

```
K(RES )E(nable) <cr>
```

Your computer responds with:

```
ABC
```

To leave these boards enabled, you simply type:

```
<cr>
```

and the KE: prompt appears, with AB and C enabled.

CPU and SYSTEM SPEED CHANGES

The CPU and System speed commands provide a means of changing clock speed directly from the monitor ROM. This feature is very useful when running software that measures the speed of the system at the time you boot. It provides a simple means of setting the CPU or System to the desired speed before you type the BOOT monitor command.

The **CPU speed** command changes just the CPU speed and leaves the bus speed at 2 MHz, as required by the H88-1 controller for the 5 1/4 inch 10 hard sectored disks. After the "K" for KRES is typed, type "C" which autocompletes as CPU speed, then either "2" or "4" followed by RETURN to set the CPU speed to either 2 or 4 MHz.

The **System speed** command changes speed on the entire system - both CPU and Bus to whatever has been selected. After the "K" for KRES is typed, type "S" which autocompletes as System speed, then either "2" or "4" followed by RETURN to set the System speed to either 2 or 4 MHz.

EXAMPLE:

1. You have the H88-1 controller for 5 1/4 inch 10 hard sectored disks, and you wish to boot at 4 MHz. At the KE: prompt, you type:

```
K(RES )C(PU speed )4
```

The KE: prompt returns, and your CPU is running at 4 MHz.

2. You do wish to boot from a controller other than the H88-1, and you wish to boot at 4 MHz. At the KE: prompt, you type:

```
K(RES )S(ystem speed )4
```

The KE: prompt returns, and your system (CPU and bus) is running at 4 MHz.

MEMORY

This command displays and alters the values in memory in dynamic RAM (ROM and static RAM memory values may only be displayed). This command is a replacement for the Heath "Substitute" command, and is named "Memory" to distinguish it from the new "Register" command. Once you type a memory address and the RETURN key after the "M"; the computer will type the memory address again and the memory value at that address. You then have the option of several subsequent commands:

YOU TYPE	ACTION TAKEN
Value <cr>	Change memory to new value and redisplay
Value <space>	Change memory to new value and increment
Value -	Change memory to new value and decrement
<cr>	Repeats current memory address
SPACE bar	Increments memory address
- (minus key)	Decrements memory address
DELETE key	Terminates the memory command
= (equal key)	Inserts and displays breakpoint

An equal sign inserts and displays a breakpoint "Pause" at a "Pause" (an "RST 0") breakpoint is subsequently encountered during program execution, interrupts are enabled and the monitor regains control. Note that any transfer of control to memory address zero (except for a hardware reset) is considered to be a "Pause"; the value in the top of the stack is assumed to be the current PC register value. If you intend to resume program execution after a "Pause", you should replace the byte at the address shown for register "PC" (in the "Pause" message) minus one, with the original value at that address (shown when the "Pause" was inserted in memory), and then transfer control to that address (via the "Go" command).

EXAMPLES:

You wish to display and change the memory values around 2293 Hex. You type

M(emory)2293 <space>

and your computer responds with the current value in the memory location 2293. Here we have assumed the value 78 is in 2293.

2293 78

You will be in the memory mode of the monitor, and can type any of the commands in the table at the top of this page. For the purposes of this example, we will assume some values already in memory, and use arbitrary numbers when changing those locations.

To change the value 78 leave the value unchanged in this location, and go on to the next location, simply tap the space bar:

<space>

The next memory location is shown and 2293 is unchanged. Your computer displays:

2294 03

To change this to, say 77 Hex, type 77 followed by a carriage return:

77 <cr>

The value in 2294 is changed to 77, and the memory location is redisplayed. Your computer responds with:

2294 77

To go back and display memory location 2293, type a "-" (minus key) to redisplay 2293:

-

Your computer responds with:

2293 78

Finally, we will change location 2293 to 00, and look at location 2292. You type:

00-

Your computer responds with:

2292 34

To leave the memory mode, type the delete key:

<delete>

And your computer returns with the KE: prompt.

NUMBER BASE

This command sets the number base (radix) for numeric values for all monitor commands except for the "Check RAM" command (below); it is a replacement for the Heath "Radix" command in MTR-90. Type an "H" to specify hexadecimal; type an "O" to specify octal. Then type the RETURN key.

EXAMPLE:

You wish to change the monitor display to Octal. At the KE: prompt, you type:

N(umber base)O(ctal) <cr>

The computer will return with the KE: prompt, and all subsequent numeric display under the monitor will be in octal.

OUT

This command outputs a value to any port. Type a port address and the RETURN key after the "O"; the computer will type the port address again. You may then type the DELETE key (which terminates the "Out" command), or you may type any of several command combinations:

Value <cr>	Output value to port and redisplay
Value <space>	Output value to port and increment
Value -	Output value to port and decrement
<cr>	Repeats current port address
= (equal key)	Redisplays port address
SPACE bar	Increments port address
- (minus key)	Decrements port address
DELETE key	Terminates the out command

EXAMPLE:

You wish to output a value to port E0. At the KE: prompt, you type:

O(ut)E0 <cr>

Your computer displays the port address and is ready for a command:

00E0

To output the value 48 Hex to port E0, then get set to output to 49 if you wish, type:

48 <space>

The computer will display the next port address:

00E1

To leave the out command, press the DELETE key:

<delete>

And your computer returns with the KE: prompt.

REGISTER

This command displays and alters the register values of a user program. Note that this command includes the function of the Heath "Program Counter" command. Type the RETURN key after the command; all register pair values will be displayed, and you may then optionally type a register name and a value.

Type the first letter of an 8080 register pair name in upper case (i.e., S(P), A(F), B(C), D(E), H(L), P(C)); type an upper case "X" or "Y" for the Z-80 index registers "IX" and "IY"; type the first letter of a Z-80 alternate register pair name in lower case (i.e., (a(f), b(c), d(e), h(l))). The computer will type the second letter of the name. If you type a value and the RETURN key after the register name, that value becomes effective for that register pair when control is next passed to the user program (any new values entered for register "SP" are ignored).

The updated register values will be displayed again, and another register name and value may be entered; to terminate the "Register" command, type the RETURN or DELETE key.

EXAMPLES:

You wish to display all register values. At the KE: prompt, you type:

```
R(register) <cr>
```

And the computer displays (picking values for registers at random):

```
XX=0123 af=4567 ... hl=89AB YY=CDEF
SP=FFF0 AF=FEDC ... HL=3210 PC=FFF0 =>
```

If you wanted to change the Program Counter to, let us say 228C Hex, you would type:

```
P(C )228C <cr>
```

And the computer will display the register values again, showing your change

```
XX=0123 af=4567 ... hl=89AB YY=CDEF
SP=FFF0 AF=FEDC ... HL=3210 PC=228C =>
```

To change the value of the Z80 alternate register, you must use lower case. For example, if you wanted to change alternate register hl to 567A Hex, you would type:

```
h(l )567A <cr>
```

and the computer would respond with:

```
XX=0123 af=4567 ... hl=567A YY=CDEF
SP=FFF0 AF=FEDC ... HL=3210 PC=228C =>
```

To leave the register command, type DELETE:

```
<delete>
```

and the computer will respond with the KE: prompt

STEP

This command transfers control to a program, executes one instruction, displays the new register PC value, and returns control to the monitor. All of the register values and the next four sequential instruction bytes are displayed.

If you type an address and the RETURN key after the "S", control passes to the program at that address. If you type just the RETURN key after the "S", control passes to the current address value in the user PC register. Note that if the single instruction executed is a "DI" (disable interrupts), execution of the program will continue until an "EI" (enable interrupts) instruction is executed. You should not step through any time-dependent code in HDOS, such as device drivers (if you step through a disk device driver, you might destroy information on a diskette).

EXAMPLE:

You wish to step through the program at 2280 Hex. You type:

```
S(tep )2280 <cr>
```

The computer will execute a single instruction beginning at 2280, and return to the monitor showing the registers:

```
XX=0123 af=4567 ... YY=CDEF  
SP=FFF0 AF=FEDC ... PC=2281 => AF 57 C9 FE  
KE:
```

To execute one more single step, beginning where the program left off, you type:

```
S(tep ) <cr>
```

And the computer will execute another single instruction beginning where the program counter was set (in our example 2281 Hex):

```
XX=0123 af=4567 ... YY=CDEF  
SP=FFF0 AF=0044 ... PC=2282 => 57 C9 FE 0A  
KE:
```

The register values will be displayed again, and any changes can be easily noted. The monitor will display the KE: prompt immediately after execution of the STEP command; it is ready for your next command.

VIEW

This command displays a block of values in memory in RAM or ROM. Type a memory address and the RETURN key after the "V"; the computer will display sixteen lines of dumped memory, followed by a seventeenth line displaying the address of the next memory location to be dumped. You may type the DELETE key (which terminates the "View" command), or a RETURN (which continues with another sixteen lines of dump).

A dumped line consists of the address of the first byte in the dumped line, followed by sixteen hexadecimal (or eight octal) bytes of dumped memory, followed by the same bytes as ASCII values; memory bytes whose low-order seven bits do not represent a displayable ASCII character are displayed as a period (".").

EXAMPLES:

You wish to view memory beginning at 3000 Hex, and are in the Hex Number Base mode. At the KE: prompt, you type:

```
V(iew ) 3000<cr>
```

and the computer will display sixteen lines of dumped memory:

```
3000 41 42 43 44 45 46 47 58 59 50 51 52 00 01 02 03  ABCDEFGHIJKL....
3010 ...
...
3100
```

You may now continue to view memory continuing from 3100 Hex. You type:

```
<cr>
```

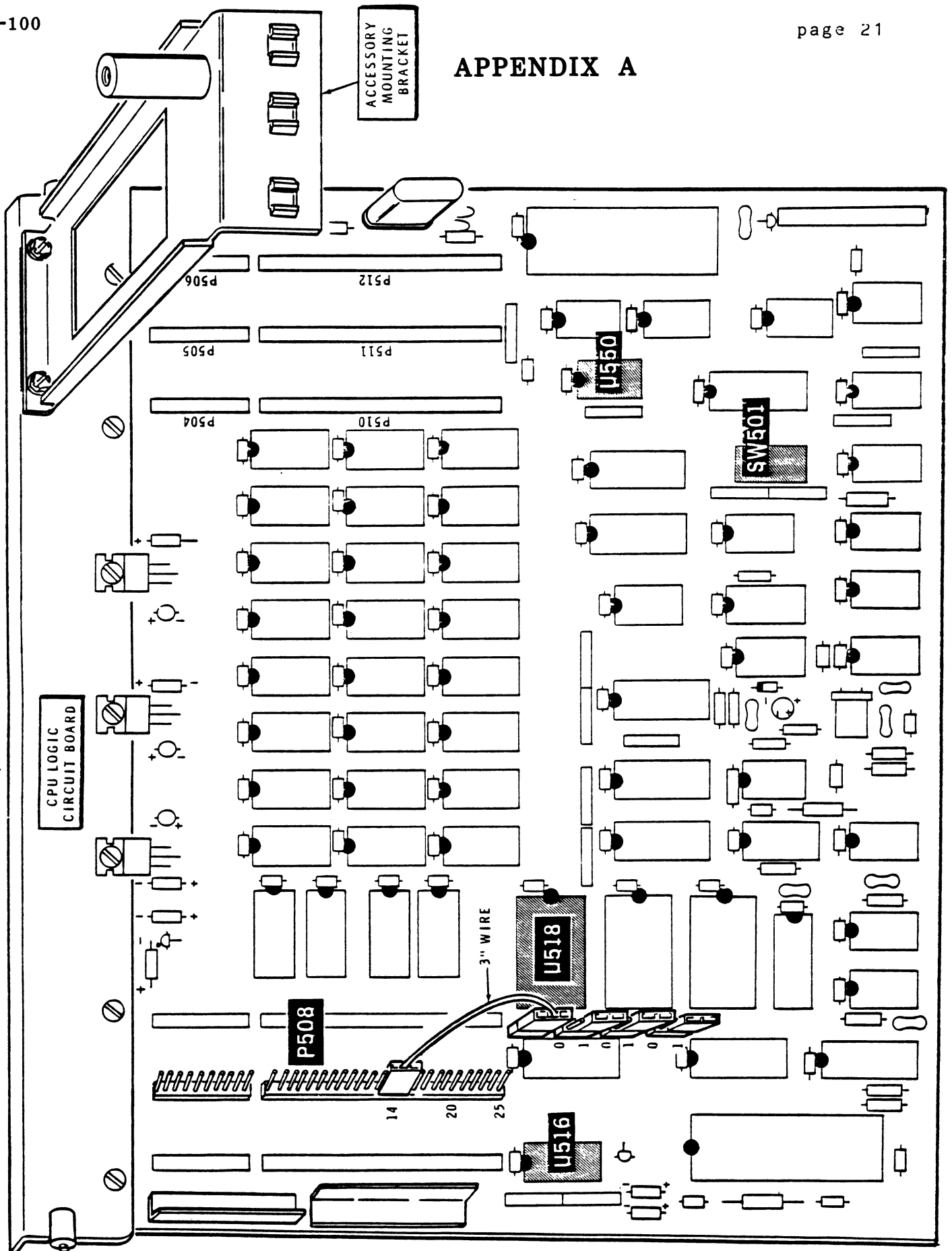
and the computer will respond with another sixteen lines of dumped memory:

```
3100 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76  abcdefghijklmnop
3110 ...
...
3200
```

You may continue to view memory by continuing to hit the RETURN key, or you may type DELETE which will return you to the KE: prompt.

APPENDIX A

PICTORIAL 1



**APPENDIX B
USE WITH NON - HEATH/ZENITH SOFTWARE AND ACCESSORIES**

The KMR-100 will work with all standard Heath/Zenith hardware, software, and firmware. The following paragraphs detail available information on some other manufacturers.

BOOTING

Several manufacturers are specifically mentioned as being supported in the earlier section on booting from the KMR-100 on page 6. Additional information on those and other manufacturers as available is provided in this section.

ACT

The standard ACT hard disk cannot be booted from the KMR-100.

CDR SYSTEMS INC

CDR CP/M is not compatible with the KMR-100, and consequently will not boot. The Livingston Logic Labs (LLL) CP/M Bios and HDOS support for the CDR controller is compatible and will boot from the KMR-100.

LIVINGSTON LOGIC LABS

All software from Livingston Logic Labs, which includes support for standard Heath/Zenith CP/M, the CDR controller, and hard disk support can be booted from the KMR-100.

MAGNOLIA MICROSYSTEMS

The KMR-100 can boot Magnolia Microsystems (MMS) CP/M from any floppy disk drive Magnolia supports (see page 7 for specific information). The KMR-100 cannot boot MMS CP/M from the 128K Ramdisk. UltiMeth Corporation's HDOS support for the Ramdisk, however, can be booted from the KMR-100.

ULTIMETH SOFTWARE

All UltiMeth software is compatible with the KMR-100.

TERMINAL LOGIC BOARD REPLACEMENT ROMS

The standard Heath/Zenith system and software is very tolerant about the form of the communications between the computer and terminal portion of your H89. Some other manufacturer's combinations are not, and your screen may suddenly react differently than it did before, notably acting erratically with software using direct cursor addressing or graphics.

The KMR-100 configures the communications from your CPU card for seven bit words, odd parity. Should you experience trouble with direct cursor addressing or graphics, and have a non-Heath/Zenith terminal logic board ROM (such as the ULTRAROM, WATZMAN ROM, or SUPER 19, consult the manufacturer's documentation on terminal logic board switch settings to configure your H89 terminal board for seven bit words, parity enabled, and odd parity.

**APPENDIX C
TECHNICAL INFORMATION:**

The KMR-100 is a 4K byte part, rated for 4 MHz operation. On power-up or reset, the monitor measures the processing speed of the CPU and also detects the baud rate of the terminal (the baud rate detection may cause garbage characters to be displayed briefly; and on power-up may take up to one second). The monitor can detect a terminal set to any of the following baud rates:

1200 2400 4800 9600 19200 38400

The CPU-terminal interface is initialized and the processing speed of the CPU is displayed. The monitor then determines the limit of RAM memory, and the user program counter (PC) and stack pointer (SP) are set to fifteen less than the address of the last byte of RAM. You may use the first 10 bytes of this area for a "JMP" instruction and/or other data. For example, if you have Org-0 CP/M, you could modify your BIOS cold start routine to place there a JMP to the warm start routine (make sure that the warm start routine remaps memory to all RAM, and reinitializes any serial ports used). Then, a reset followed by the monitor "G(o)" command (with no address) performs a CP/M warm start.

APPENDIX D
OPTIONAL HARDWARE MODIFICATION

This modification allows the monitor ROM to detect when an external switch is depressed. On each clock interrupt, section "5" of SW501 on the CPU board is sampled; if the switch is ON (closed, or set to "0"), the monitor regains control. Typing a "G(o)" command (with no address) returns control to the original program at the point of interrupt. Another switch can be wired in parallel with section "5" of SW501, giving an external interrupt capability.

NOTE THAT KRES Engineering and UltiMeth Corporation CANNOT ACCEPT ANY RESPONSIBILITY FOR THE PROPER FUNCTIONING OF YOUR UNIT IF YOU MAKE THE FOLLOWING MODIFICATION. This modification does require that you solder wires to some of the printed circuit boards inside your computer. You must use good engineering practices and perform the modification carefully. Before making any hardware modification to your unit, read the remainder of this section carefully to understand both the modification and its effect on your unit.

ALWAYS BEFORE MAKING ANY MODIFICATION TO YOUR COMPUTER, BE CERTAIN THE COMPUTER IS OFF AND THE LINE CORD IS UNPLUGGED

Any modification that provides a switch in parallel with section "5" of SW501 will work. Be sure and use the Heath x-ray charts provided with the unit to locate parts and to check your work; the circuit diagrams (schematics) are useful to double-check your work and to understand the modification. Note that some x-ray charts and circuit diagrams, section "5" of SW501 is numbered "6". The surface of SW501 should show numbers from "0" to "7"; at any rate, section "5" is the third section from the bottom of the circuit board. We recommend the following modification, which allows you to use the RESET key on the keyboard as the additional "external interrupt" switch:

() Remove the terminal board (the printed circuit board behind your CPU) and set it flat on a work area before you.

() Install a jumper wire on the component side of the circuit board from pin 12 of P404 to pin 17 of P403. On an H-19 terminal board, pin 17 is equivalent to the far end (near the bottom of the board) of R419 (above P403); alternatively, if you are using small diameter wire (which you should), you may use the feed-through hole just below C471. On an H-19A terminal board, pin 17 (the numbering on the x-ray view is wrong) is equivalent to RP3 pin 3; similarly, you may use the feed-through hole just above RP3 pin 3 (be careful to locate the correct feed-through hole).

() Modify the cable that runs from P404 on the terminal board to P513 on the CPU board to include a wire from pin 12 to pin 12. If you do not have any remaining "small spring connectors" from your assembly of the unit, you will have to find two. They are available from Heath as part number 432-866.

() Remove the CPU board and install a jumper wire on the foil side of the circuit board from pin 12 of P513 to the far end of section "5" of SW501.

This completes the OPTIONAL HARDWARE MODIFICATION to your machine. You may now reassemble your computer and test, or refer back to the manuals of other hardware installations that may be in progress.

Section "5" of SW501 (and any switch installed in parallel with it) is used for two purposes: When the computer is reset, if the switch is ON, the monitor performs a dynamic RAM test. At any other time, whenever the monitor updates the clock (TICCNT), which occurs every 2 milliseconds when the CPU is enabled for clock interrupts, the monitor examines the switch. If the switch is ON, the monitor suspends execution of whatever program is running. The following is displayed:

[Interrupt]

```
XX=nnnn af=nnnn bc=nnnn de=nnnn hl=nnnn YY=nnnn
SP=nnnn AF=nnnn BC=nnnn DE=nnnn HL=nnnn PC=xxxx => bb bb bb bb
```

Note that "xxxx" is the current PC register value (the location of the next instruction to be executed), and "bb bb bb bb" are the next four sequential instruction bytes.

This means that if you wire the RESET key on the keyboard in parallel with section "5" of SW501, and you depress the RESET and (right-) SHIFT keys to reset the computer, you may hear two "beeps" - - one when the monitor detected the RESET key being depressed, and another when the RESET key is released and the computer resets. Also, if you reset the computer and you release the SHIFT key before you release the RESET key, the monitor will perform the dynamic RAM test, destroying the contents of memory.

You should realize that some computer programs should not be interrupted. For example, some device drivers will lose data if they are interrupted in the middle of a critical task. A properly written device driver will protect itself by disabling interrupts during the execution of critical timing situations. Interrupts are generally only disabled for short periods of time. HDOS was written with this in mind; however, you should be aware of the following: The mini-floppy device driver uses the clock for timing, and thus is enabled during part of its execution to allow the clock to be updated. If you interrupt HDOS while one of the mini-floppy disk drives is selected (the LED is on), the LED will stay on until you exit from the monitor, and you may get a soft disk error. Therefore, we recommend not interrupting HDOS while disk operations are in progress. AGAIN, KRES Engineering and UltiMeth Corporation CANNOT ACCEPT ANY RESPONSIBILITY FOR LOST DATA AS A RESULT OF USING THIS FEATURE. You should always have adequate backups of both programs and data on separate diskettes.

Also, interrupting out of a program which is in the process of altering the terminal baud rate without protecting itself by disabling interrupts (such as the HDOS boot track code) may cause the terminal to "hang up", requiring the computer to be reset.

APPENDIX E HEATH COMPATIBILITY

The following routine entry points in the Heath MTR-88, -89, and -90 ROMs have been preserved at their original locations. Note that only the function of the routine has been preserved; the code may have changed. The routines have been preserved only to allow compatibility with existing programs; it is strongly recommended that you do not write code that references these routines, as Heath or UltiMeth may not provide them in future ROMs. In ROMs which support hexadecimal, routines that used to perform octal conversion functions now perform the corresponding hexadecimal conversion functions. The TYPMSG routine now stops on either a null character or a character with the high-order bit on.

Heath routine name	Location	
DLY	002B	000.053
INTXIT	007A	000.172
PIN	0137	001.067
ALARM	025E	002.136
HORN	0260	002.140
IOA	0332	003.062
IOB	0336	003.066
RCC	03B2	003.262
WCC	03C2	003.302
DAT	0613	006.023
COM	0617	006.027
TYPMSG	0640	006.100

The following RAM locations are used or reserved by the monitor ROM:

Heath location name	Location(s)	
(reserved)	2000-2007	040.000-040.007
.MFLAG	2008	040.010
CTLFLG	2009	040.011
(reserved)	200A-201A	040.012-040.032
TICCNT	201B-201E	040.033-040.036
UIVEC	201F-2033	040.037-040.063
(reserved)	2034-203F	040.064-040.077

.MFLAG

The default value for this byte is zero. If bit 0 is set, the monitor will CALL the instruction at UIVEC on each clock interrupt with all interrupts disabled; return should be made TO THE MONITOR via the "RET" instruction without enabling interrupts. After the CALL, the stack contains, in ascending order, the following items (note that this list is the same as Heath's list):

- (SP+0) = Return location in the monitor.
- (SP+2) = Stack pointer value at time of interrupt.
- (SP+4) = Program status word (register AF).
- (SP+6) = Register BC.
- (SP+8) = Register DE.
- (SP+10) = Register HL.
- (SP+12) = Register PC.

Note that HDOS supplies a JMP to its own clock interrupt processing routine (bit 0 is set during HDOS operation); this routine is critical to the proper operation of the mini-floppy disks.

If bit 7 is set, the monitor will not attempt to detect a halt condition (bit 7 is set during HDOS operation). If bit 7 is not set, the clock interrupt routine checks the byte prior to the next instruction byte to be executed (normally the last byte of the previous instruction executed) on each interrupt. If the operation code value for a "HLT" instruction is found, the monitor regains control. The following is displayed:

```
[Halt]
XX=nnnn af=nnnn bc=nnnn de=nnnn hl=nnnn YY=nnnn
SP=nnnn AF=nnnn BC=nnnn DE=nnnn HL=nnnn PC=xxxx => bb bb ...
```

Note that "xxxx" is the current PC register value (the location of the next instruction to be executed), which is the location of the "HLT" operation code value plus one; "bb bb ..." are the next four sequential instruction bytes. Note that many multiple-byte instructions could have the byte previous to the next instruction to be executed look like a "HLT" operation code, so the monitor may falsely detect a halt condition when none exists (which is why Heath provided bit 7 of .MFLAG to disable this feature under HDOS).

CTLFLG:

This byte should reflect the current value output to port F0 (hexadecimal) or 360 (octal). Since this port exists only on an H8, and not on an H89, the H89 hardware causes a Z-80 nonmaskable interrupt to be generated whenever a reference is made to an H8 port address. The monitor ROM intercepts the non-maskable interrupt and, if an 8080 "OUT" instruction reference to port F0 (hexadecimal) or 360 (octal) caused the non-maskable interrupt, the monitor changes the format of the output value to the corresponding H89 value, and outputs the value to port F2 (hexadecimal) or 362 (octal). (Note that the monitor "I(n)" and "O(ut)" commands use Z-80 input and output instructions, and thus produce meaningless results when used to reference H8 port addresses.)

User programs wishing to output to port 360 (octal) or F0 (hexadecimal) should disable interrupts, get the current value of CTLFLG, alter bits 4 and/or 6 as desired, store the new value in CTLFLG, then output to the port, and then enable interrupts.

If bit 4 is set, the single step interrupt feature is disabled (the default). If bit 4 is reset, the single step feature is enabled. When the single step interrupt occurs, the monitor will JMP to the instruction at UIVEC+3 with interrupts disabled; the return to the point of interrupt is the responsibility of the user program. After the JMP, the stack contains, in ascending order, the following items (note that this list is the same as Heath's list):

```
(SP+0) = Stack pointer value at time of interrupt.
(SP+2) = Program status word (register AF).
(SP+4) = Register BC.
(SP+6) = Register DE.
(SP+8) = Register HL.
(SP+10) = Interrupt return location (register PC).
```

Bit 6 is set to enable clock interrupts (the default). Note that HDOS requires that clock interrupts be enabled (see above under .MFLAG).

TICCNT:

This 4-byte area contains a 32-bit binary value which is incremented on every clock interrupt (every 2 milliseconds). TICCNT contains the lowest-order byte, and TICCNT+3 contains the highest-order byte. The two high-order bytes have been added to Heath's TICCNT to allow the value in TICCNT to represent more than 24 hours. Note that HDOS disk I/O operations disable interrupts for periods of time exceeding 2 milliseconds, causing clock interrupts to be delayed; thus, TICCNT cannot be used as a highly accurate 24-hour clock.

UIVEC:

This 21-byte area reserves a 3-byte instruction area for each possible user interrupt (1 to 7); user interrupts "1" and "2" are the clock and single step interrupts, respectively, and are discussed above under ".MFLAG" and "CTLFLG". User interrupts "3" through "7" cause the monitor to JMP directly to the following entry in UIVEC with interrupts disabled:

Interrupt number:	JMP location:
3	UIVEC+6
4	UIVEC+9
5	UIVEC+12
6	UIVEC+15
7	UIVEC+18

After the JMP, the stack contains only the interrupt return location. The user interrupt routine must save and restore any registers it uses, and must enable interrupts before returning to the point of interrupt. Note that HDOS uses interrupt "3" (for console I/O) and "7" (for SCALLs).

**APPENDIX F
HDOS PATCHES FOR HEX DISPLAY**

The following patches to HDOS 2.0 programs change the number radix (base) for output from split octal to hexadecimal (for SET, the patches allow hexadecimal input). THESE PATCHES ARE SUPPLIED AS IS, AS A SERVICE TO UltiMeth CUSTOMERS. NO WARRANTY IS MADE AS TO THE CORRECTNESS OF THESE PATCHES. CUSTOMERS SHOULD HAVE ADEQUATE BACKUPS OF THE PROGRAMS PATCHED, AS WELL AS CUSTOMER DATA, BEFORE MAKING THESE PATCHES. You should have some program for making the patches. An appropriately modified version of Heath's PATCH program will work; however, we recommend the use of the Heath User's Group program "DUMP". THESE PATCHES ONLY APPLY TO HDOS 2.0.

PATCHES TO ASM.ABS

The following patches make Heath's ASM print hexadecimal output:

FILE	PROG	OLD	NEW	EXPLANATION		
DISP	ADDR	DATA	DATA			
0E89	3101	3E68	3E67	MVI	A,67	GET LOW-ORDER BYTE OF PRINT LINE LOC OF LAST HEX DATA BYTE
0F67	31DF	215C3A	215B3A	LXI	HL,3A5B	GET PRINT LINE LOC OF FIRST HEX DATA BYTE
0FEF	3267	21533A	21553A	LXI	HL,3A55	GET PRINT LINE LOC OF FIRST ADDRESS BYTE
0FF6	326E	362E23	000000	NOP,NOP,NOP		DELETE PERIOD INSERTION
1237	34AF	C5	F5	PUSH	PSW	CONVERT TO HEX ROUTINE (SEE ISSUE 14 OF REMARK)
...	...	06	0F	RRC		
		03	0F	RRC		
		A7	0F	RRC		
		17	0F	RRC		
		1717F5	CDB834	CALL	*+4	
		E6	F1	POP	PSW	
		07C6	E60F	ANI	0FH	
		3077	C690	ADI	90H	
		23	27	DAA		
		F105	CE40	ACI	40H	
		C2	27	DAA		
		B3	77	MOV	M,A	
		34	23	INX	HL	
		C1	C9	RET		

PATCHES TO XREF

In addition, the following patches must be made to XREF to make it print hexadecimal output:

0514	278C	21B927	21BB27	LXI	HL,27BB	GET PRINT LINE LOC OF FIRST ADDRESS BYTE
0541	27B9	3030	2020	DB	' '	CLEAR UNUSED PRINT LINE
0A3E	2CB6	C5	F5	PUSH	PSW	CONVERT TO HEX ROUTINE (SEE ISSUE 14 OF REMARK)
...	...	06	0F	RRC		
		03	0F	RRC		
		A7	0F	RRC		
		17	0F	RRC		
		1717F5	CDBF2C	CALL	*+4	
		E6	F1	POP	PSW	
		07C6	E60F	ANI	OFH	
		3077	C690	ADI	90H	
		23	27	DAA		
		F105	CE40	ACI	40H	
		C2	27	DAA		
		BA	77	MOV	M,A	
		2C	23	INX	HL	
		C1	C9	RET		

PATCHES TO SYSCMD.SYS

The following patches to SYSCMD.SYS cause the STAT command to print its output in hexadecimal:

04CB	2743	218D27	218F27	LXI	HL,278F	GET PRINT LINE DATA LOC
04D5	274D	21A927	21AB27	LXI	HL,27AB	GET PRINT LINE DATA LOC
04DF	2757	21C927	21CB27	LXI	HL,27CB	GET PRINT LINE DATA LOC
0514	278C	093030	202020	DB	' '	CLEAR UNUSED PRINT LINE
0530	27A8	093030	202020	DB	' '	CLEAR UNUSED PRINT LINE
0550	27C8	093030	202020	DB	' '	CLEAR UNUSED PRINT LINE
0ACA	2D42	C5	F5	PUSH	PSW	CONVERT TO HEX ROUTINE (SEE ISSUE 14 OF REMARK)
...	...	06	0F	RRC		
		03	0F	RRC		
		A7	0F	RRC		
		17	0F	RRC		
		1717F5	CD4B2D	CALL	*+4	
		E6	F1	POP	PSW	
		07C6	E60F	ANI	OFH	
		3077	C690	ADI	90H	
		23	27	DAA		
		F105	CE40	ACI	40H	
		C2	27	DAA		
		46	77	MOV	M,A	
		2D	23	INX	HL	
		C1	C9	RET		

PATCHES TO SET.ABS

The following patches to SET.ABS allows hexadecimal values (e.g., port addresses) to be SET as well as binary, octal, and decimal values. The SET value should be followed by a radix indicator of "B", "O", "Q", "D", or "H" (e.g., "340Q" or "EOH"). The Heath device drivers have various default radices (bases) for each numeric parameter that can be entered, but they don't document what the default radix is for each parameter. Line widths, page depths, etc., are usually decimal; port addresses and character values are usually octal. When in doubt, specify the radix. Note that a hexadecimal value ending in "B" or "D" MUST have an explicit radix, or the "B" or "D" will be interpreted as a radix indicator!

09FA	2C72	D8FE0A	C3E92C	JMP	2CE9	CONVERT HEX INPUT TO BIN
0A2B	2CA3	32	47	MOV	B,A	SAVE DEFAULT RADIX
...	...	F02C44	CD6E2C	CALL	2C6E	FIRST CHARACTER VALID -
		4D	D8	RC		- NO, ERROR
		7E	5D	MOV	E,L	SAVE DATA LOC
		CD	54	MOV	D,H	
2CAA	6F	23	INX	HL		POINT TO NEXT CHARACTER
	2C23D2	CD6E2C	CALL	2C6E		VALID CHARACTER -
	A82C2B	D2AA2C	JNC	2CAA		- YES, CONTINUE SCAN
	79	7D	MOV	A,L		COMPUTE BYTES SCANNED
	BD	93	SUB	E		
	37	4F	MOV	C,A		SAVE STRING LENGTH
	C8	E5	PUSH	HL		SAVE END OF STRING LOC
	7E	2B	DCX	HL		GET POSS. BASE CHAR LOC
	E5	7E	MOV	A,M		GET POSS. BASE CHARACTER
	21F12C	21F62C	LXI	HL,2CF6		GET BASE TYPE TABLE LOC
	A7CACB	CD762C	CALL	2C76		GET BASE TYPE ENTRY LOC
	2CCD76	C2C62C	JNZ	2CC6		- NOT FOUND, SKIP
	2C	0D	DCR	C		DECREMENT STRING LENGTH
	7E	37	STC			
	C2CB2C	CAE72C	JZ	2CE7		- ZERO LENGTH, ERROR
	E1	46	MOV	B,M		GET BASE VALUE FROM TABLE
2CC6	23E532	210000	LXI	HL,0000		SET INITIAL VALUE
2CC9	F0	D5	PUSH	DE		SAVE INPUT CHARACTER LOC
	2C	EB	XCHG			(DE) = OLD VALUE
	11	78	MOV	A,B		GET BASE
	00003A	CD0719	CALL	1907		MULTPLY OLD VALUE BY BASE
	F0	D1	POP	DE		RESTORE INPUT CHAR LOC
	2C	37	STC			
	F5CD07	C2E72C	JNZ	2CE7		- MULTIPLY OVERFLOW, ERROR
	19	1A	LDAX	DE		GET INPUT CHAR TO CONVERT
	D1DAEE	CD6F2C	CALL	2C6F		CONVERT INPUT VALUE TO BIN
	2C	B8	CMP	B		VALID RADIX DIGIT -
	0A	3F	CMC			
	CD6F2C	DAE72C	JC	2CE7		- NO, ERROR
	03BA3F	CD4118	CALL	1841		ADD NEW DIGIT TO VALUE
	DA	13	INX	DE		GET NEXT INPUT CHAR LOC
	EE	0D	DCR	C		END OF STRING -
	2CCD41	C2C92C	JNZ	2CC9		- NO, CONTINUE CONVERSION
	18	EB	XCHG			(DE) = RESULT
	EB	AF	XRA			CLEAR CARRY

2CE7	0A	E1	POP	HL	RESTORE END OF STRING LOC	
	D6	C9	RET		RETURN	
2CE9	30D2	FE0A	CPI	10D	DECIMAL DIGIT -	
...	CD	3F	CMC			
	2C	D0	RNC		- YES, RETURN	
	A7E1	FE11	CPI	'A'-'0'	VALID HEX DIGIT -	
	C9	D8	RC		- NO, RETURN	
	0042	D607	SUI	'A'-'9'-1	CONVERT HEX DIGIT	
	024F	FE24	CPI	10D+26D	VALID ALPHABETIC CHAR -	
	08	3F	CMC			
	51	C9	RET		RETURN VALIDITY FLAG	
2CF6	0844	4202	DB	'B',10B		
...	0A00	4F08	DB	'O',100		
	????	5108	DB	'Q',10Q	The "??" indicate	
	????	440A	DB	'D',10D	uninitialized	
	????	4810	DB	'H',10H	values in the	
	??????	000000	DB	0,0,0	file.	

**APPENDIX G
ULTIMETH COPYRIGHT AND POLICY**

COPYRIGHT NOTICE:

This software is protected under the copyright laws of the United States of America. It is provided for the use of the original purchaser on one computer at a time. Copies may NOT be made for friends (whether or not for compensation), nor for use on multiple computers (we have a reduced rate schedule for multiple computers).

PRODUCT POLICY:

The purchaser is solely responsible for the installation, supervision, management and control of its use of the software, including, but not limited to:

1. Determining that the computer equipment on which the software will be operated, and the operating system and version thereof, are ones for which the software was designed.
2. Establishing, implementing, and using adequate procedures to protect users, data, programs, files, operating systems, and equipment from any undesired effects of the software, including, but not limited to, testing, backup, and restoration procedures.
3. Integrating the software with other software and hardware he may own or have a license to use, including testing all such combinations he intends to use.
4. Performing, or having performed for him at his own expense, any diagnostic procedures necessary from time to time to verify and isolate to a specific component or feature, any problems or difficulties he may encounter with any software or hardware component he may own or have a license to use.

We suggest that if you are not already somewhat familiar with your computer hardware and software that you obtain someone to assist you with any problem you may have with your computer. Computer hardware and software are very sophisticated, technical products; when everything works right, they can be easy to use, but problems can be difficult to solve without experience. We have found that customer problems fall into the following categories:

1. Approximately 50% have not obtained and thoroughly read and understood the documentation for the hardware and software that they already have before they purchase our product.
2. Approximately 30% have a problem with associated hardware: defective equipment, or improperly installed and configured equipment.
3. Approximately 15% have not thoroughly read and understood our documentation for our product.
4. Approximately 5% have a problem with associated software: a software "bug", or improperly installed and configured software.
5. Less than 1% have a problem with our distribution media (diskette or ROM chip) being defective.
6. Less than 1% have found a problem with our software.

We try to make our products compatible, insofar as feasible, with other known products. Unfortunately, we have found that most developers of software or hardware do not share this attitude. As a result, you may find another product that is not compatible with one of our products, particularly if the other product was developed after our product. Because of the above, we do not provide any of our products on a trial basis, subject to refund, nor do we always update our products when incompatibilities are found.

We are sorry, but we cannot provide, except on a prepaid fee consulting basis, subject to schedule availability, assistance in the following areas:

1. Installing or configuring anyone else's software or hardware.
2. Tracking down general system problems.
3. Modifying our software.
4. Designing and implementing software or hardware.

We do NOT provide source code for ANY of our products, because most such requests come from product developers who wish to use our ideas without compensation. We have generally coded our software so as to make comprehension of disassembled code difficult.