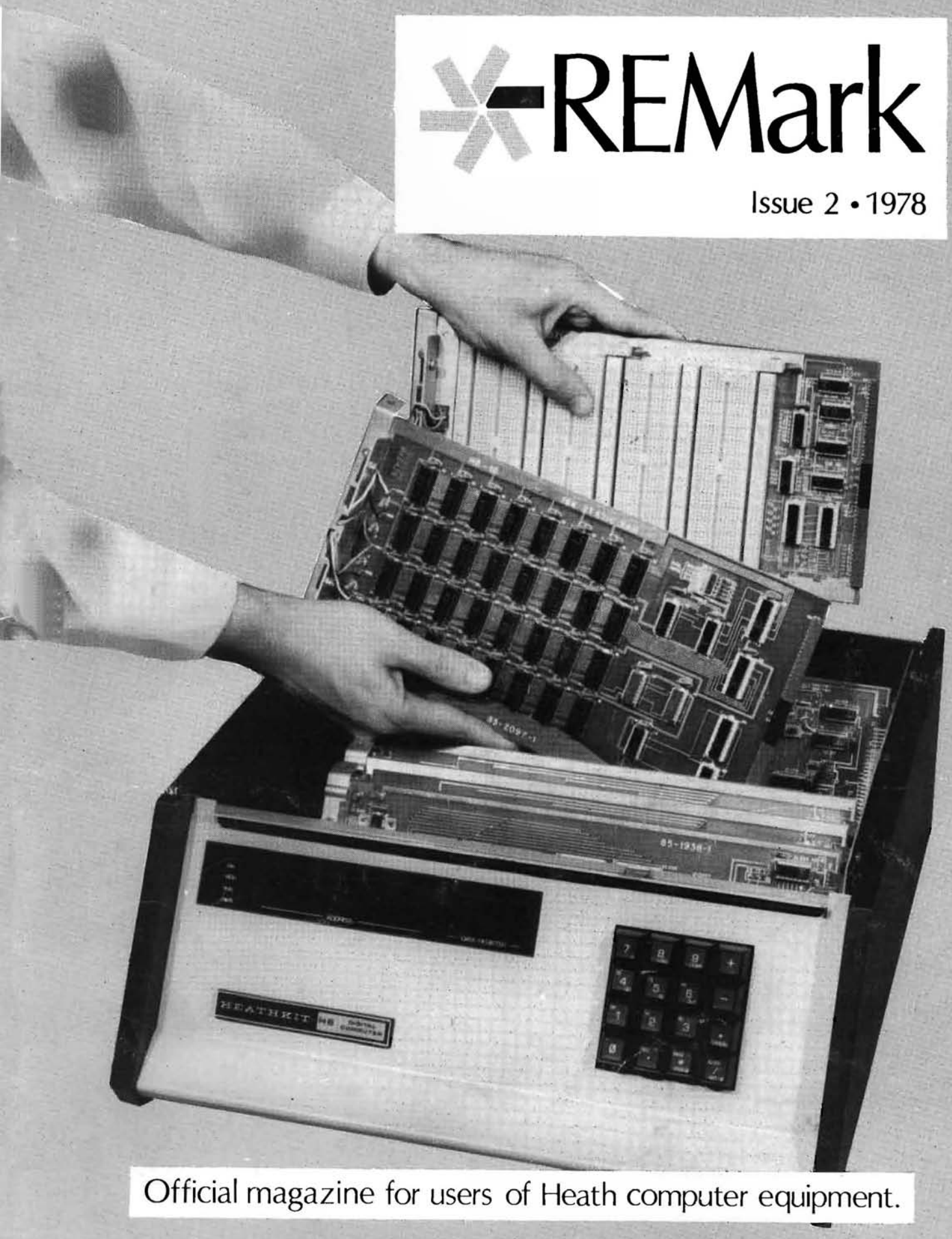
REMark

Issue 2 • 1978



Official magazine for users of Heath computer equipment.

REMark

CONTENTS

"REMark" is a HUG membership magazine published quarterly. A subscription can not be purchased separately without membership. The following membership rates apply.

	U.S. Domestic	Canada & Mexico	Internat'l
Initial	$14	$16	$24
Renewal	$11	$13	$18

Send payment to: Heath User's Group, Hilltop Road, St. Joseph MI, 49085. Back issues available cost $2.50 post paid in the U.S. Foreign requests should specify mailing method and add the appropriate cost.

All prices shown in REMark are U.S. mail order prices. Products sold through Heathkit Electronic Centers are slightly higher. For international orders, contact your nearest Heath Co. distributor or the Heath International Division, P.O. Box 440, St. Joseph, MI. 49085.

Although it is a policy to check material placed in REMark for accuracy, HUG offers no warranty, either expressed or implied, and is not responsible for any losses due to the use of any material in this magazine.

Computer Prod. Director .....Lou Frenzel
HUG Manager and Tech. Editor .....Robert Furtaw
Graphics .....Ron Hungerford

Two new boards soon to be available for the H8. 16K static RAM card for further memory expansion and a breadboard card for easy designer experimentation.

NEW PRODUCT INTRODUCTION

The Facts of Life

The Heath Company really broke with tradition when it introduced its new computer product line. By this I mean that we introduced the computer products months before they were actually available for shipment. Typically Heath Company has never announced a product unless it was available off the shelf by mail order or through one of our Heathkit Electronic Centers. We have always felt that the best time to introduce a product is when we actually have it in stock. However, the procedure of announcing a product well ahead of it's actual availability has been the standard not only in the personal computing business but also in all computer fields. While we do not necessarily agree with such an approach, it does have some benefits that tend to offset the disadvantages.

The disadvantages of introducing a product early is that customers, like you, become frustrated when they know of the existence of a product but are unable to get it. This generated hundreds of phone calls and letters. Most people simply do not want to wait. Another disadvantage is that product specifications or features may be changed at the last minute. When you announce a product early, you are often making the introduction during the latter part of the development phase. Products can undergo slight changes as the design is finished and all of the various quality assurance checks and evaluations have taken place. Then, when the product hits the market, it could be slightly different from what was originally announced. This can cause anxiety, confusion, and disappointment.

These are the main reasons Heath Company has generally elected not to announce products early. But with the computer products we felt obligated to do it. Why? Mainly because of the tremendous pressure put on us by you the customer. Most of you knew that we were developing products for personal computing. For one year before we announced we were literally inundated by requests to release information. We try to give customers what they want. In this case information was released.

Despite the disadvantages of introducing products earlier we are continuing with this policy. For example, we have already announced the availability of our floppy disk units for the H8 and the H11 computers. More product introductions are on the way. However, we are attempting to work back to our original policy of announcing when the product is available. In the meantime, you can take advantage of the early introductions. By knowing about which products are coming, you can better plan your computer system purchase. When making such a major investment, it is nice to know ahead of time the entire range of available products as well as those coming. By having this information you can select only those items that you want thus possibly avoiding duplication or the loss of money incurred when replacing one product with another.

What new products are coming? Well, you already know about the floppy disk systems. At the Second West Coast Computer Fair in San Jose in March, we also announced two new accessories for the H8 computer. These are the WH8-16, a low power 16K memory card. This new accessory will allow you to buy more memory for less cost. We also announced the H8-7 breadboard accessory. This is a card that allows you to breadboard special interfaces and circuitry for use on the H8. It comes with on-board voltage regulators, address decoders, and bus buffers. It also contains many breadboarding sockets where you can quickly and easily build prototype circuits.

Beyond those announced products there are many others. More memory and I/O accessories are coming for both computers. More and better software is on the way. And there are several new major peripherals in the works. Most of these products are enhancements or supplements to the existing products. I'm sure you will find them exciting and interesting. We will be announcing them and telling you about them here first.

Finally, let me say that quality new products are simply not developed over night. We are doing all we can to shorten our development times to get new products to you quickly. However, we simply will not compromise and reduce the quality of our products, the documentation or our ability to support them. You may have to wait a little while, but you can be assured that when the product comes that it will be a good one, and you will be able to get the desired service and support. Thanks for your continued support and patience.

Lou Frenzel

There were quite a few requests for information on how to use a low cost Baudot printer to obtain hard copy. This article, submitted by Howard L. Nurse answers this request. It was also submitted to Kilobaud Magazine Peterborough NH 03458 to appear in a future issue.

We have verified this article and made some changes to it to help insure your success. Also, we added a circuit board drawing which was not originally submitted by the author . . . hind sight always has the advantage. For Amateur Radio use, it can easily be used to key an FSK or AFSK circuit to provide "clatter" free TTY transmission.

Our thanks to Howard Nurse for the hours of preparation that we know went into developing this article.

Bob

An ASCII/Baudot Driver for the H8 System

By: Howard L. Nurse
655 Maybell Ave.
Palo Alto, Ca.

Baudot teleprinters, frequently available as surplus for next to nothing, can be used as a listing device for your H8 system with the driver described here. Baudot printers use 5 information bits to print 58 characters rather than the 7 bits used by ASCII printers. Even though ASCII teleprinters can print many more characters than Baudot machines, there are only a few characters used in programming which are lacking in the 5-level printer.

By assigning arbitrary symbols to cover the desirable ASCII characters not available in the Baudot set, the missing characters can be accommodated. The following assignments have been made in this driver:

Back Space	= X	Greater Than (>)	= (G)
Percent (%)	= (C)	AT (@)	= (A)
Asterisk (*)	= (X)	Left Bracket ([)	= (B)
Plus (+)	= (P)	Reverse Slant (\)	= (R)
Less Than (<)	= (L)	Right Bracket (])	= (B)
Equal (=)	= (E)	Up Arrow (↑)	= (U)
		Underline (_)	= (-)

You can make arbitrary assignments as you desire by modifying the look-up table used in the driver program to convert ASCII characters generated in the computer to Baudot characters.

Hardware

The ASCII/Baudot driver was written for the Heath H8 computer system. This documentation assumes that the interface to the teleprinter will be made via an H8-2 Parallel I/O port.

A Universal Asynchronous Receiver Transmitter (UART), a hex inverter integrated circuit, Baudot clock and 60 ma loop circuitry are required to complete the interface. The loop is not within the scope of this article. A schematic and parts list for this outboard circuitry are given in Figure 1 and Table 1, respectively.

The baud rate potentiometer, R2, should be adjusted to a frequency which is 16 times the teleprinter baud rate. If the teleprinter has 60 wpm gears, the clock rate should be 727 Hz, while if the machine has 100 wpm gears, the baud rate clock should be adjusted to 1187 Hz. This can be done easily by measuring the time output at pin 3 of U2 with a frequency counter or by adjusting the speed control until the printer operates normally.

INTERFACE PARTS LIST

INTEGRATED CIRCUITS

U1	AY-5-1013A	UART	-12 V Required on Pin 2
U1 (Alt.)	AY5-1014	UART	-12 V Not Required

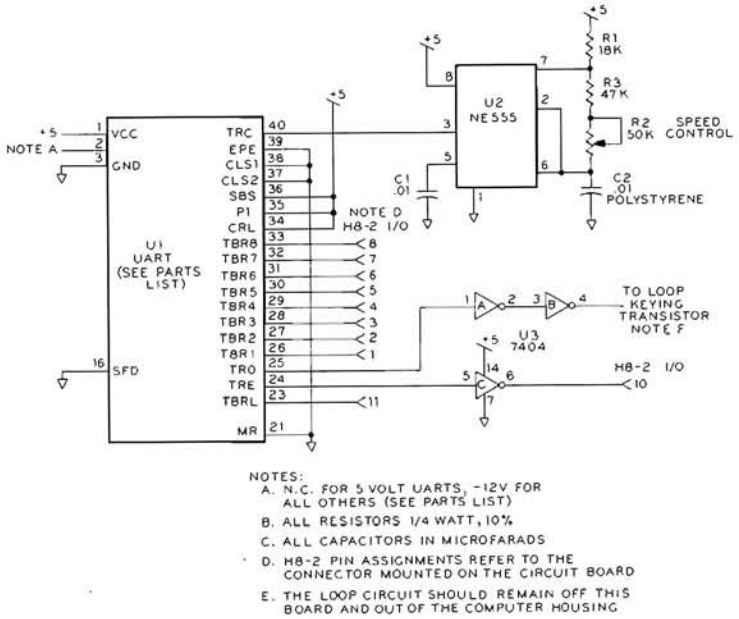
RESISTORS

R1	18K, 1/4 W	
R2	50K, 10 Turn	Potentiometer
R3	47K, 1/4 W	

CAPACITORS

C1	.01 μfd Disk Ceramic
C2	.01 μfd Polystyrene (High Stability)

TABLE 1



NOTES:
A. N.C. FOR 5 VOLT UARTS, -12V FOR ALL OTHERS (SEE PARTS LIST)
B. ALL RESISTORS 1/4 WATT, 10%
C. ALL CAPACITORS IN MICROFARADS
D. H8-2 PIN ASSIGNMENTS REFER TO THE CONNECTOR MOUNTED ON THE CIRCUIT BOARD
E. THE LOOP CIRCUIT SHOULD REMAIN OFF THIS BOARD AND OUT OF THE COMPUTER HOUSING

Figure 1

A flat ribbon cable was used to connect the UART board to the H8-2 parallel card. Since +5 volts is not available on the connector, direct connection to +5V is necessary. You can pick this voltage up at different points on the parallel card.

Select one of the three channels on the parallel board that you would like to use, then configure it as shown in Table 2.

Jumpers:	A1-A2 — OPEN
	B1-B2 — OPEN
	C1-C2 — INSTALLED
	E1-E2 — INSTALLED
	F, G & H — OPEN

TABLE 2

PHYSICAL PLACEMENT

The circuit board is mounted on edge, inside the left side of the computer. Two small L brackets will be sufficient for support. The ventilation holes on the bottom of the cabinet provide a natural mount for the board.

Software

The ASCII/Baudot Driver uses a look-up table to convert the ASCII character being sent to the system console to a Baudot character compatible with a 5-level teleprinter. The routine is placed at the upper end of user RAM, just below the stack, where provisions have been included in Heath software to allow user memory space. The following description assumes that you have knowledge of using the System Software.

1. Load Distribution Tape.
2. Change High Memory by pressing H and using "Reconfigured High Memory" table below. In other words, change the number first shown in column "2" to the corresponding number in column "5".
3. Change Pad from 4 to 1 (optional).
4. Record Heath program using "SAVE".
5. Proceed to instructions on Table 4.

1	2	3	4	5
TOTAL USER RAM	ACTUAL DECIMAL LIMIT	ACTUAL SPLIT OCTAL LIMIT	BEGIN EQU	RECONFIGURED HIGH MEMORY
4K	12287	057377A	056000A	11775
8K	16383	077377A	076000A	15871
12K	20479	117377A	116000A	19967
16K	24575	137377A	136000A	24063
20K	28671	157377A	156000A	28159
24K	32767	177377A	176000A	32255
28K	36863	217377A	216000A	365351
32K	40959	237377A	236000A	40447

TABLE 3: Reconfiguring Heath Programs

The Heath software must be reconfigured to allow the ASCII/Baudot Driver to be used. The changes include:

1. Redefining the high memory limit as detailed in Table 3.
2. Changing 5 memory locations in each Heath program, as shown in Table 4.

1. Load tape prepared using instructions from Table 3
2. Note Stop Address displayed on the front panel of the H8
3. Change contents of five memory locations using the Table.
4. Re-enter Stop Address from step 2.
5. Put new tape in record cassette machine.
6. Push "DUMP".
7. You now have a tape containing a program, the Editor for example, which has a new High Memory point (to protect the ASCII/Baudot Driver), a new Pad, and five changes in the I/O Driver portion to provide jumps into the ASCII/Baudot Driver routines. Note that this has taken two load and dump procedures.

MEMORY ADDRESS	WAS (Octal)	CHANGE TO (Octal)	COMMENTS
040365A	111	027	ENTRY LO
040366A	040	"BEGIN" HI*	ENTRY HI
041040A	303	315	CALL
041041A	117	000	INIT LO
041042A	040	"BEGIN" HI*	INIT HI

Use high byte of BEGIN EQU from Table 3. For example, if your H8 has 16K of RAM, you would use 136 at memory locations 040366A and 041042A.

TABLE 4: Modifying Heath I/O Driver

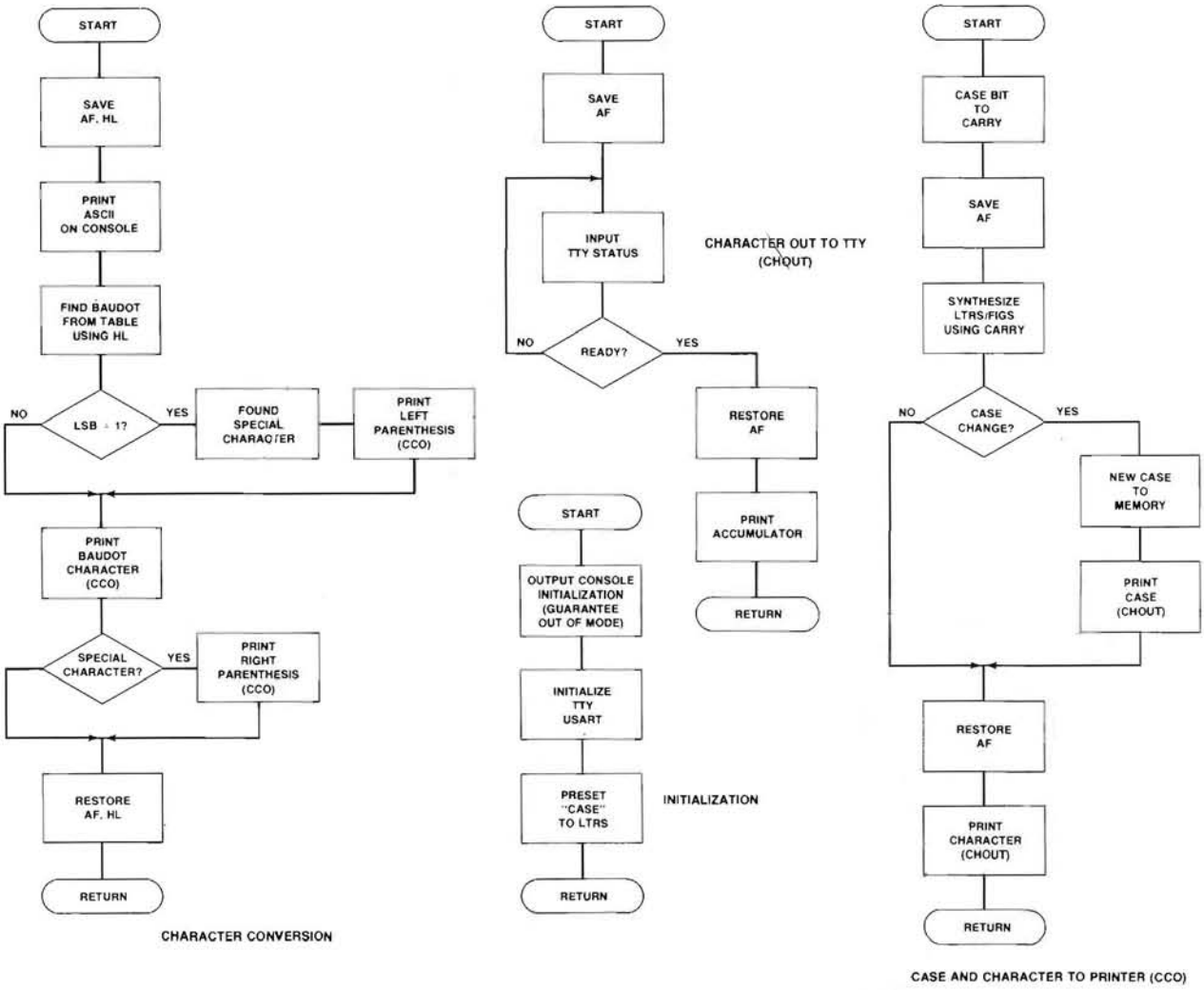


Figure 2

Once the Heath software has been reconfigured, the ASCII/Baudot routine is loaded with the Heath program. The ASCII/Baudot driver need only be loaded once when going from one Heath reconfigured program to another. For example, the following describes a typical use of the driver with the H8 Editor and Assembler.

1. Load ASCII/Baudot Driver from cassette tape.
2. Load reconfigured H8 Editor.
3. Push GO. The Baudot printer will now print each character sent to the CRT, with substitutions where required.
4. Write and/or edit source code.
5. Save source code on cassette tape.
6. Load reconfigured H8 assembler.
7. Push GO. The assembly will proceed normally, with all information printed on the Baudot teleprinter.

Flow charts for the driver are given in Figure 2, while the complete assembly listing (as printed on a Model 28 teleprinter) is given in Figure 3. The program requires approximately 270 RAM locations.

If you enter the source listing into TED8, the driver can be easily modified to be compatible with your H8 system. The program as shown in Figure 3 was assembled to run with a system having 16K of RAM. If your system differs from this, the BEGIN EQU statement must be modified to reflect your high memory limit in offset octal. For example, if your system has a total of 8K of RAM, the BEGIN EQU statement from Table 3 would be:

```
BEGIN  EQU  76000A
```

If you wish to use an output port other than 70 octal, the TTY EQU statement must be changed to the new port number.

Once you have defined the changes to be made to the ASCII/Baudot Driver, the source code listing can be changed and the program reassembled.

Figure 3

```
H8 ASCII/BAUDOT DRIVER - HLN - 12/10/77                    HASL #04.01.02
INITIALIZATION                                              PAGE  1


                        (X) THE FOLLOWING PROGRAM ALLOWS A STANDARD
                        (X) 5-LEVEL BAUDOT TELEPRINTER TO BE USED AS
                        (X) A LISTING DEVICE FOR SOFTWARE USING THE
                        (X) H8 CONSOLE DRIVER, V 1.1.

                        (X) THIS H8 ASCII/BAUDOT DRIVER
                        (X) WAS WRITTEN BY HOWARD L. NURSE.


                        (X) SYMBOLIC CONSTANTS:

136.000                 BEGIN   EQU     136000A 511 BYTES BELOW HI MEM
000.070                 TTY     EQU     070     TTY PORT
040.111                 $CDOUT  EQU     40111A  H8 CONSOLE DATA OUT
040.117                 $CISO   EQU     40117A  H8 CONSOLE IN STATUS OUT

136.000                 START   ORG     BEGIN


136.000  315 117 040    INIT    CALL    $CISO   GUARANTEE OUT OF MODE
136.003  323 071                OUT     TTY(P)1 SAME FOR TTY PORT
136.005  076 100                MVI     A,100   RESET USART
136.007  323 071                OUT     TTY(P)1
136.011  076 116                MVI     A,116   MODE SET
136.013  323 071                OUT     TTY(P)1
136.015  076 005                MVI     A,005   COMMAND SET
136.017  323 071                OUT     TTY(P)1
```

```
136.021  076 037          MVI     A,370   CASE TO LTRS
136.023  062 153 136       STA     CASE    STORE SAME
136.026  311              RET

                  (X) SAVE ACTIVE REGISTERS AND OUTPUT
                  (X) CHARACTER TO H8 CONSOLE.

136.027  365        ENTRY PUSH    PSW     SAVE AF
136.030  345              PUSH    H       SAVE HL
136.031  315 111 240       CALL    $CDOUT  CONSOLE CHARACTER

                  (X) CONVERT ASCII CHARACTER USING TABLE.

136.034  041 154 136 TABLE LXI     H,TBL   ESTABLISH POINTER
136.037  205              ADD     L       OFFSET POINTER BY ASCII
136.040  322 044 136       JNC     COK     NO CARRY AFTER ADD
136.043  244              INR     H       WAS CARRY AFTER ADD
136.044  157        COK   MOV     L,A     COMPLETE TABLE ADDRESS
136.045  176              MOV     A,M     GET BAUDOT CHRCTR
136.046  017              RRC             IF SPECIAL CHARACTER
136.047  334 122 136       CC      LFT       PRINT LEFT PARENTHESIS
136.052  315 072 136       CALL    CCO
136.055  346 040           ANI     40C     IF NOT A SPECIAL CHRCTR
136.057  312 065 136       JZ      DONE      EXIT
136.062  315 132 136       CALL    RGT     SPECIAL CHRCTR REQUIRES
136.065  341        DONE  POP     H         RIGHT PARENTHESIS
136.066  361              POP     PSW     RESTORE HL AND AF
136.067  311              RET

                  (X) CASE AND CHARACTER OUT TO PRINTER.

136.070  017        CCO   RRC             START CASE CHECK
136.071  365              PUSH    PSW     SAVE SHIFTED CHARACTER
136.072  076 303          MVI     A,303C  SYNTHESIZE CASE
136.074  077              CMC             INVERT CARRY
136.075  027              RAL             AND SHIFT LEFT
136.076  027              RAL             AGAIN
136.077  027              RAL             ONCE MORE
136.100  041 153 136       LXI     H,CASE  POINT TO CASE
136.103  276              CMP     M       IS IT SAME AS BEFORE?
136.104  312 115 136       JZ      CASOK   YES, SO EXIT
136.107  062 153 136       STA     CASE    NO, SO FIX MEMORY
136.112  315 140 136       CALL    CHOUT   AND SEND IT TO PRINTER
136.115  361        CASOK POP     PSW     RESTORE AF
136.116  315 140 136       CALL    CHOUT   AND PRINT CHARACTER
136.121  311              RET

                  (X) SEND LEFT PARENTHESIS.

136.122  365        LFT   PUSH    PSW     SAVE AF
136.123  076 037          MVI     A,37C   LEFT PARENTHESIS
136.125  315 070 136       CALL    CCO     PRINT SAME
136.130  361              POP     PSW     RESTORE AF
136.131  311              RET

                  (X) SEND RIGHT PARENTHESIS.

136.132  076 045    RGT   MVI     A,45C   RIGHT PARENTHESIS
136.134  315 070 136       CALL    CCO     PRINT SAME
136.137  311              RET

                  (X) SEND CHARACTER TO PRINTER AFTER
                  (X) CHECKING STATUS.

136.140  365        CHOUT PUSH    PSW     SAVE AF
136.141  333 071    STAT  IN      TTY(P)1 GET STATUS
136.143  017              RRC             IS USART READY?
136.144  322 141 136       JNC     STAT    IF NOT, WAIT
136.147  361              POP     PSW     RESTORE AF
136.150  323 070          OUT     TTY     PRINT CHARACTER
136.152  311              RET
```

```
136.153                 CASE    DS      1       LTRS/FIGS STATUS MEMORY

                        (X) BIT 0, IF 1, YIELDS LEFT PARENTHESIS
                        (X) BIT 1, IF 1, YIELDS UPPER CASE (FIGS)
                        (X) BITS 2 - 6 ARE BAUDOT CHARACTER
                        (X) BIT 7, IF 1, YIELDS RIGHT PARENTHESIS

136.154  200           TBL     DB      00      BLANK
136.155  220                   DB      200
136.156  020                   DB      200
136.157  220                   DB      200
136.160  020                   DB      200
136.161  020                   DB      200
136.162  020                   DB      200
136.163  026                   DB      260     BELL
136.164  164                   DB      1640    BS (PRINTS 'X')
136.165  220                   DB      220
136.166  012                   DB      120     LINE FEED
136.167  220                   DB      220
136.170  022                   DB      220
136.171  040                   DB      400     CARRIAGE RETURN
136.172  020                   DB      200
136.173  020                   DB      200
136.174  220                   DB      220
136.175  220                   DB      220
136.176  020                   DB      200
136.177  020                   DB      200
136.200  020                   DB      200
136.201  020                   DB      220
136.202  020                   DB      220
136.203  220                   DB      200
136.204  020                   DB      200
136.205  020                   DB      200
136.206  020                   DB      200
136.207  020                   DB      200
136.210  020                   DB      220
136.211  020                   DB      200
136.212  020                   DB      200
136.213  020                   DB      200
136.214  020                   DB      200     SPACE
136.215  266                   DB      660     !
136.216  106                   DB      1060    "
136.217  122                   DB      1220    #, OR USE 3210 (H)
136.220  046                   DB      460     $, OR USE 2450 (D)
136.221  271                   DB      2710    PERCENT (C)
136.222  152                   DB      1520    &, OR USE 3610 (M)
136.223  056                   DB      560     '
136.224  076                   DB      760     (
136.225  112                   DB      1120    )
136.226  365                   DB      3650    ASTERISK (X)
136.227  331                   DB      3310    PLUS (P)
136.230  062                   DB      620     ,
136.231  016                   DB      160     -
136.232  162                   DB      1620    .
136.233  166                   DB      1660    /
136.234  132                   DB      1320    0
136.235  136                   DB      1360    1
136.236  116                   DB      1160    2
136.237  006                   DB      60      3
136.240  052                   DB      520     4
136.241  102                   DB      1020    5
136.242  126                   DB      1260    6
136.243  036                   DB      360     7
136.244  032                   DB      320     8
136.245  142                   DB      1420    9
136.246  072                   DB      720     :
136.247  172                   DB      1720    ;
136.250  311                   DB      3110    LESS THAN (L)
136.251  205                   DB      2050    EQUAL (E)
136.252  351                   DB      3510    GREATER THAN (G)
136.253  146                   DB      1460    ?
136.254  215                   DB      2150    AT (A)
```

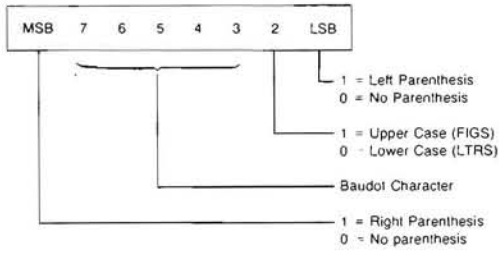
```
136.255   214          DB      140      A
136.256   144          DB      1440     B
136.257   270          DB      720      C
136.260   044          DB      440      D
136.261   004          DB      40       E
136.262   264          DB      640      F
136.263   150          DB      1500     G
136.264   120          DB      1220     H
136.265   032          DB      300      I
136.266   054          DB      540      J
136.267   074          DB      740      K
136.270   110          DB      1120     L
136.271   160          DB      1620     M
136.272   060          DB      620      N
136.273   140          DB      1420     O
136.274   130          DB      1320     P
136.275   134          DB      1340     Q
136.276   052          DB      520      R
136.277   024          DB      240      S
136.300   100          DB      1000     T
136.301   034          DB      340      U
136.302   170          DB      1720     V
136.303   114          DB      1140     W
136.304   164          DB      1640     X
136.305   124          DB      1240     Y
136.306   104          DB      1040     Z
136.307   345          DB      3450     LEFT BRACKET (B)
136.310   251          DB      2510     REVERSE SLASH (R)
136.311   345          DB      3450     RIGHT BRACKET (B)
136.312   235          DB      2350     UP ARROW (U)
136.313   217          DB      2170     UNDERLINE (-)

136.314                END     START

STATEMENTS (E) 00217
FREE BYTES - 09571
NO ERRORS DETECTED.
```

LOOK-UP TABLE

The look-up table contains one data byte per ASCII character. Each byte contains the 5-bit Baudot character to be printed, plus 3 control bits. As shown in Figure 4, the least significant bit in the byte is 1 if the character is one of the special Baudot characters that will be enclosed in parenthesis. If the second bit in the byte is 1, the Baudot character will be shifted to upper case (FIGS). If the most significant bit in the byte is 1, a right parenthesis will be printed after the Baudot character. Several examples of typical characters are given in the figure.

Minor keyboard variations can exist from one printer to another which may require you to alter your look-up table.

Machines in common use include those having Telex, TWX, Weather, Military Standard, and International Alphabet #2 character sets. Stay away from Weather machines since their character set contains many useless upper case weather symbols. The differences in the remaining four character sets are summarized in Table 5.



Examples

1. Assume "A" is the ASCII character.
   Binary byte for "A" is 00001100.
   Octal equivalent of binary byte is 140O.
   Baudot machine points A

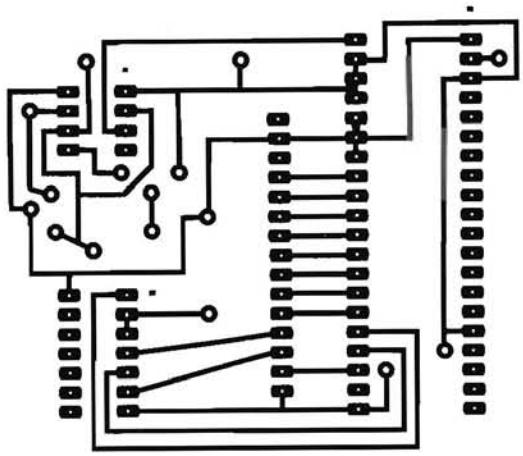
2. Assume "-" is the ASCII character.
   Binary byte for "-" is 10000101.
   Octal equivalent of binary byte is 205O.
   Baudot machine prints (E).

3. Assume "5" is the ASCII character.
   Binary byte for "5" is 01000010.
   Octal equivalent of binary byte is 102O
   Baudot machine prints 5.

FIGURE 4: Look-Up Table Byte Structure

UPPER CASE	TELEX	TWX	MILITARY	INTERNATIONAL #2
B	?	5/8	?	?
C	:	WRU	:	:
D	WRU	$	$	WRU
F	$	1/4	!	NA
H	#	NA	STOP	NA
J	BELL	'	'	BELL
K	(	1/2	(	(
L	)	3/4	)	)
S	'	BELL	BELL	'
Z	"	"	"	"

TABLE 5: Differences in Common Baudot Character Sets



FOIL LAYOUT (FULL SIZE)

The look-up table in this driver assumes that a "Military Standard" printer is used. The printer, a model 28KSR, has been modified to provide a "#" character for upper case H. It is particularly easy to substitute characters in the type basket of the model 28. Other characters that may be substituted include a "*" for upper case H, and a "+" for upper case G. Remember that the look-up table must be modified to reflect any changes that you wish to make.

CRT displays allow use of the backspace function (ASCII Control H), while most printers do not. Since it is desirable to tell when a backspace has occurred, the character "X" is printed each time you backspace. If during an edit you made a mistake and backspaced to correct it, the following would be printed:

MISTOKEXXXAKE

When listed a second time the spelling would be correct.

All ASCII characters not covered by normal and special equivalents in the table will cause a "space" character to be printed. Note that lower case alphabetic characters are not included in the table, although they can be easily added at the end of the table if necessary.

ACKNOWLEDGMENTS

Thanks are due Dr. Glyn Harding for his helpful suggestions on program structure, and Irv Hoff for his suggestions pertaining to the look-up table.

H11 ASCII Printout Test

You can verify simple system performance with a minimum of hardware and software by printing ASCII characters on a terminal. This test can be performed with only the processor and serial card installed. No additional memory other than what is provided on the processor board is required. A terminal such as a DEC Writer must be connected to the serial card.

Enter the program as shown in Figure 1 into your H11 via the console terminal using μODT. The program starts at address 1000_8 but if you prefer, you may start at any point since it is completely relocatable.

```
                ;GENERATES ASCII OUTPUT TO A TERMINAL
        000000          R0=%0
        000000          .ASECT
        001000          .=1000
001000  105737 LOOP:    TSTB    @#177564
        177564
001004  100375          BPL     LOOP
001006  110037          MOVB    R0,@#177566
        177566
001012  005200          INC     R0
001014  000771          BR      LOOP
        001000          .END    LOOP
```

Figure 1

Once loaded, it is executed by entering the starting address then typing G. If you have loaded it exactly like it appears type:

1000G

The console will then start typing the complete printable ASCII character set as shown in Figure 2. It will continue to do so until you hit the break key on the console or momentarily place the RUN/HALT switch in the HALT position.

With the system producing a successful output you can be reasonably sure that the following items work properly:

Processor

Serial Card

Power Supply

DEC Writer or other terminal*

* The H9 terminal will not convert lower to upper case characters.
Only the upper case characters will be printed properly.

This is not a complete system checkout. It only provides a reasonable reassurance that the system is in fact functional.

The BASIC binary bit pattern which makes up an ASCII character can be directed to an H10 reader/punch. In this case the bit pattern representing ASCII characters will be punched on paper tape. You can do this by changing the data at addresses 1002 and 1010 to 177554 and 177556 respectively. The program can be executed and stopped exactly as before.

Execute the program and allow it to punch about 4 feet of tape then stop the program. When you examine the tape, you will find that a series from 1 to 377_8 has been punched. The most significant bit (MSB) is located on the right hand side of the tape as it comes out of the punch.

Both the DEC Writer and the H10 reader/punch output characters at exactly 10 characters per inch. Lay the paper tape beneath a line of the ASCII character produced from the DEC Writer as shown in Figure 3. Locate 101_8 on the paper tape and line it up with the upper case A. You should then find that all the other characters produced by the terminal line up with their corresponding bit pattern produced by the punch.

When the tape is matched against the printout you may notice that one edge of the punched tape from 1 to 377_8 is much longer than the sequence provided on the printout. This is accounted for in two ways.

First, ASCII characters are represented by 0 to 177_8. Therefore, the MSB from an eight bit pattern is ignored since the maximum number for eight bits is 377_8.

Second, many of the lower number of characters are not printable characters. They are control characters. They appear on the tape but not on the printout. Presence of some of these are proved by the execution of the carriage return and line feed and also by the occurrence of the bell that you hear.

You may want to permanently glue the paper tape to the printout. This will provide a handy ASCII reference chart. The program can be used to quickly check the output section of the system if you ever suspect any problems in the future.

```
@1000G
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ
001004
```

Figure 2

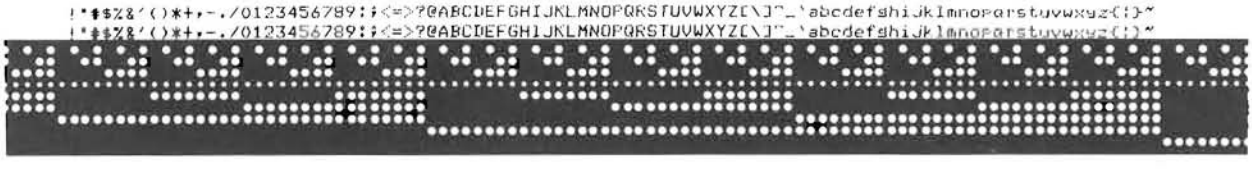


Figure 3

THE COMPUTER HOBBYIST WHAT'S HE LIKE?

By: Murray Nichols
Heath Company
Manager, Market Research

In developing a new product line, such as computers, it is very important to know as much about the background, interests, and desires of the potential customer as possible. Market Research at Heath Company has a number of ways to accomplish this, some of which are as follows:

Survey forms mailed to Heath customers and to people known to be interested in the computer hobby.

Survey forms packed with computer and peripheral kits that are returned by the builder.

Personal interviews with computer hobbyists.

The new product request section that is filled out on the Heathkit mail order form.

Analysis of actual orders from our customers.

We have made a great deal of use of survey forms in our quest to understand the computer hobbyist. Some of you may have received one of our forms in the mail last year. All of you have seen the forms packed with our computer kits. One thing we learned early was that computer people like to be asked their opinions. We know this because the rate of return of the forms has been very high.

We also learned there was great curiosity among the survey respondants to know the results of the surveys. Thats the reason for this article — to let you know what you, as a group of computer hobbyists, are like.

Personal Background

One of the first things of interest about the computer hobbyist is his general background. This includes such things as age, education, occupation, income, knowledge of computers, etc. In the market research business, these data in statistical form are known as demographics. Demographics of groups of different interests are usually different from each other and from the U.S. population as a whole. As we will see, our surveys show the computer hobbyist to be adult but relatively young, well educated, and rather affluent.

Chart one shows the percent of persons in various age ranges who are interested in Heathkit computers and compares them to U.S. population. Almost half of these computer hobbyists fall into the 25 to 34 year old age range and another quarter are 35 to 44. Notice that only twelve percent of hobbyists are under the age of 25 while this same group accounts for almost half of the U.S. population. Some reasons could be that this group is still in school or becoming established in society and does not have the financial resources or time to actively persue the hobby. Many of those in school also have access to an institutional system. Notice also the sharp decline in computer interest from the age of 45 onward. This group sometimes tells us they have reached a point in life where they no longer want to learn a whole new field.

AGE OF COMPUTER HOBBYISTS		
Age Range	Percent of Replies	1974 U.S. Population
Under 25	12%	46%
25-34	46%	14%
35-44	25%	11%
45-54	11%	11%
Over 55	6%	18%
TOTAL	100%	100%

CHART 1

Chart two shows the level of education reached by the computer hobbyist. Almost two thirds have obtained an undergraduate college degree, continued on to graduate work, or have a graduate degree. All but six percent have some additional training beyond the high school level. I do not have statistics of educational attainment for the general population arranged in the same way as the chart. It is fair to say, however, that those interested in hobby computers are much better educated than the general public.

EDUCATION OF COMPUTER HOBBYISTS	
Level of Education	Percent of Replies
Completed High School or less	6%
Some College or Junior College	19%
Completed Junior College	10%
College Degree	20%
Some Graduate Work	17%
Graduate Degree	28%
TOTAL	100%

CHART 2

Chart three shows the occupations of people interested in Heathkit computers. As we might expect from the group's high level of education, quite a large percentage are employed at the professional level. Notice that 26% of hobbyists have a computer related job, 31% are Engineers or Scientists, and 7% are Technicians. That means that a total of 64%, or almost two thirds of those interested in computers as a hobby either work with computers during the day or have a strong technical background. The last category on the chart, "other occupations", contains all those jobs which by themselves would not be a significant percentage. Many of them are also at the professional level. It is safe to say that in almost any occupation you can name, there is at least one computer hobbyist.

OCCUPATIONS OF COMPUTER HOBBYISTS	
Occupational Field	Percent of Replies
Computer Related	26%
Engineer or Scientist	31%
Technician	7%
Student	4%
Teacher	5%
Business	4%
Other Occupations	23%
TOTAL	100%

CHART 3

As we might expect from the occupational level of people interested in computers, their family income levels are well above those of the general population. Chart four shows the family incomes of computer hobbyists and compares them to U.S. families as a whole. The majority of those interested in computers, 83%, have incomes of $15,000 a year or more compared to 45% of all U.S. families. By contrast, the remaining 17% of computer hobbyists have incomes below $15,000 compared to 55% of all U.S. families.

FAMILY INCOME OF COMPUTER HOBBYISTS		
Income Range	Percent of Replies	U.S. Families 1975
Under $5,000	1%	12%
$5,000 - 9,999	2%	21%
$10,000 - 14,999	14%	22%
$15,000 - 24,999	47%	31%
$25,000 and above	36%	14%
TOTAL	100%	100%

CHART 4

People interested in Heathkit computers were asked to rate their knowledge of software and programming. Chart five shows that only 11% consider themselves expert in this area. However, 45%, almost half, feel they are average to very good in software knowledge. Then there is almost the other half, 44%, who feel their software skills are mediocre or less. This means that if you think you have a lot to learn about software, welcome to the club, you are not alone.

Computer hobbyists were also asked to rate their knowledge of hardware and electronics. About the same number as in software, 12%, consider themselves hardware experts. However, 68% feel their

SOFTWARE KNOWLEDGE OF COMPUTER HOBBYISTS	
Knowledge Level	**Percent of Replies**
Expert	11%
Very Good	21%
Average	24%
Mediocre	37%
What is Software?	7%
TOTAL	**100%**

CHART 5

HARDWARE KNOWLEDGE OF COMPUTER HOBBYISTS	
Knowledge Level	**Percent of Replies**
Expert	12%
Very Good	34%
Average	34%
Mediocre	18%
What's an IC?	2%
TOTAL	**100%**

CHART 6

hardware skills are average to very good. It would seem that computer hobbyists are a little more comfortable in hardware than software.

Why Computers?

A key question in the study of the computer hobbyist is why is he interested in having his own computer? We didn't ask this question directly but we can gain a lot of insight by examining the applications he plans for his computer. When we asked about applications we asked both what they would be initially and what they would be within two years.

There is no doubt that the biggest reason hobbyists want computers initially is because they are fascinated by them and want to learn about them. It is the learning experience that is initially the largest single factor in the desire to have a computer. Playing games is also a popular application initially but there is little interest in them as a long term application. It seems to me that games are a natural initial application because they give the hobbyist something to do as a part of his learning process. We also know learning about computers is important because of the strong interest expressed in software and hardware courses we asked about in our surveys. Other popular initial applications are mathematics, using the computer as an aid to one's job, amateur radio, and word processing.

As I mentioned, we asked hobbyists about applications beyond those they planned initially. Control of home devices is a very popular objective. Others are graphics, data bases, and synthesized music. While these are the most popular there are many others too numerous to mention. They range from satellite tracking to control of a loom.

At this point you may have thought I forgot to mention business applications. I really haven't but it is an area that needs special treatment because there is a fine line between the hobbyist businessman and the businessman hobbyist. They are hard to distinguish in surveys. We know that some H-8 purchasers plan to use their computers purely for business or institutional applications. This is even more noticeable among H-11 purchasers. But what about the hobbyist?

Many say they plan to use their computers for both business and personal applications. Business, however, ranges from balancing a checkbook at home to inventory control, accounts receivable, payroll, etc. It is not clear, however, whether the commercial applications mentioned will be done at home or on the business premises. It is evident that there is also a desire among some hobbyists to engage in a future computer business activity as a sideline, such as providing data processing services to clients. Some of this desire may also help the hobbyist rationalize his computer investment.

System Patterns

Speaking of investment, what is the computer hobbyist doing with the money he spends on his hobby? One of the things hobby publications have been saying is that there is an increasing tendency for him to want assembled hardware rather than kits. As the hobby continues to grow this might happen but our surveys revealed that more than 80% of current hobbyists, given the choice between wired and kit computers prefer the kit. The two reasons most mentioned were to save money and to become better acquainted with the hardware through the assembly process.

How much do hobbyists invest in a computer system? Before our own computers were announced our surveys showed that about sixty percent of hobby computer owners had invested between one thousand and three thousand dollars in their system. Another twenty percent exceeded three thousand dollars. This investment includes random access memory and peripherals. You might be interested in knowing that the average H-8 system has 13.3K of memory initially. Almost seventy percent of H-8 customers tell us they plan to increase their memory to 32K to 64K in the future.

What peripherals are popular? Other than tape I/O, we knew from surveys that a video terminal would be a most popular peripheral. Not only is it the most desired peripheral, it is one that the hobbyist wants for his initial system rather than something he wants later. Two other peripherals are also in great demand. One is a floppy disk storage device and the other is a printer. The interesting thing about these two peripherals is that while the hobbyist wants them, they are not something he usually wants initially but something he plans to add later. That is one reason Heath put its efforts into the video terminal first and then began developing a printer and floppy disk which will be available shortly.

I hope you have enjoyed learning some of the things computer hobbyists as a group have told us about themselves. I say as a group because in Market Research we never discuss information given us by a single individual. That is always confidential. The hobby computer market is an emerging market in which attitudes will constantly change in the process of maturing. We at Heath will continually probe the market to be sure we are serving your needs the best we can. That's the reason for Market Research surveys. Next time you get one, please answer it carefully. It will help all of us.

Using the H8 Console Driver

INTRODUCTION

A console driver is a group of software routines which provides communication between a terminal and a computer. Such a driver is placed at the beginning of each one of the software tapes supplied with the H8 computer. It appears in TED-8, HASL-8, BUG-8, and B.H. BASIC. It is also used in Ex. B.H. BASIC and other machine language programs supplied by Heath. If you intend to do any machine language programming considerable time will be saved by using this driver.

The purpose of this article is to tell you how to use the routines. It is beyond the scope to tell you how the routines work.

Some of the features of this driver are:

- Input from a keyboard is interrupt operated.
- A 28 character type ahead buffer is provided. Characters may be typed in before they are used and saved.
- Initialization routines are provided to preset UARTS.
- Addresses reserved for starting programs.

Obtaining the Driver

The listing for this software driver is found in the software documentation supplied with the H8. It is not necessary to load the program by hand using the listing. The driver can be loaded into the H8 by loading the beginning of any one of the software tapes, such as TED-8. The driver part of the program starts at 040.100 and ends at 041.144. As the program is loaded, watch the display increment through memory. You can stop the loading anytime past the 041.144 address, but it would be wise to allow all of the tape to load. This will assure a valid load, as indicated by a short beep at the end.

Once loaded, you will want to make a copy of the driver alone. Use the memory dump routine in PAM-8 to make a memory image from 040.100 to 041.144.

Presetting UARTS

The console terminal UART is initialized at 372 by using this routine called $PRSCL.. It must be called following a hardware reset or right after the H8 is turned on. It must be the first routine called in order to provide any communication at all between the computer and the console terminal. The beginning address of this routine is 040.367. However, a jump is provided earlier in the listing which is preferred. This address is 040.152. This should be the first routine called in any program and it should only be called once. The following shows the convention and how this routine is called.

```
315 152 040 CALL $PRSCL
```

Character Input

Somewhere during your program you will probably want to input a character from the keyboard. This is accomplished by using the routine called $RCHAR.. A type ahead buffer is provided where characters are stored until you reached the point in your program when $RCHAR. is called. When you do reach this point, an ASCII character in the buffer is loaded into the accumulator. The buffer can contain 28 charac-

ters. When the buffer is full, a beep is heard from the terminal. This beep is heard because a control character, 07_8 (BELL) is sent to the terminal. It is important to note that the last character filling the buffer, causing the beep, is lost. Therefore, if you want that character, you will have to retype it.

Like the previous routine, there is a jump to the actual routine earlier in the listing. To use this routine in your program, CALL $RCHAR at 040.147. The convention and assembly looks like:

```
315 147 040 CALL $RCHAR
```

When your program enters this routine, it will wait for a character to enter the buffer if it is empty. It will not leave this routine until it gets one. Also note that this routine, by itself, will not echo a character back on the screen or page.

Outputting Characters

Once you have processed some information you will eventually want to print the results. This is accomplished by calling $WCHAR. at the point in your program where you have an ASCII character in the accumulator to output. The routine itself is found starting at 040.332. However, like before the preferred calling address is 040.147. The convention and assembly looks like this:

```
315 147 040 CALL $WCHAR
```

Echoing Characters

Most of the time it is desirable to see the character on the screen as you type it. This is accomplished, for the most part, by calling $WCHAR immediately following your calling $RCHAR. This would look like this:

```
315 144 040 CALL $RCHAR

315 147 040 CALL $WCHAR
```

The echo feature is effective only while the buffer is empty, before a character is typed in. When the buffer is partially full, you will not see the characters being echoed. You may have noticed this in other Heath programs supplied with the H8.

There are times when you may not want the character to be echoed. This may be, for example, when security is involved. If you have a password that you would not like to see printed you can simply not use the $WCHAR after the $RCHAR.

If you suspect that you will echo characters each time, you can save some program space another way. In the listing $WCHAR immediately follows $RCHAR. They are separated by a RET instruction. Change this instruction at address 040.331 to 000, a NOP. When this is done, you only need to call $RCHAR and the program will automatically fall through to $WCHAR, then RETURN.

Starting Programs

It's easy to remember one starting address for all your programs. Provisions are made to start all programs at address 040.100. The first six addresses are set aside to first CALL $PRSCL and then JUMP to the start of your program which should be at 041.145. Then all you have to remember is to load the Program Counter with 040.100 then hit GO.

Summary Program

The best summary is to provide you with an example of what has been already explained.

```
040 100   315 152 040          CALL $PRSCL

040 103   303 145 041          JMP START

              .                    .

              .                    .

              .                    .

041 145   315 144 040  START   CALL $RCHAR

041 150   315 147 040          CALL $WCHAR

041 153   303 145 041          JMP START
```

The preceding program is used as follows:

1. Load the console driver from one of the distribution tapes such as TED-8.
2. Change and add the data at addresses shown in the previous listing.
3. Change the program counter to 040.100.
4. Press GO on the H8 pad.

You will be able to press any key on a console terminal and have it echo.

Good luck in your programming.

H9 Lower to Upper Case Modification

INTRODUCTION

The H9 video terminal was designed for upper case only. It is required that lower to upper case conversion be done in software before the character is sent to the terminal. All Heath software can handle this. If lower case characters are applied, unrelated symbols and characters are printed.

This modification will provide a hardware lower to upper case conversion. It is installed in your H9. This modification will not be supplied in new kits.

Parts Required

The change requires one 7402 and about 3 inches of hookup wire.

Installation Instructions

( ) Remove the AC plug from the power socket.

( ) Remove the bottom cover from your H9.

( ) Locate the Character Generator board and remove it.

( ) Refer to Figure 1. Examine the drawing carefully then cut the foil where indicated.

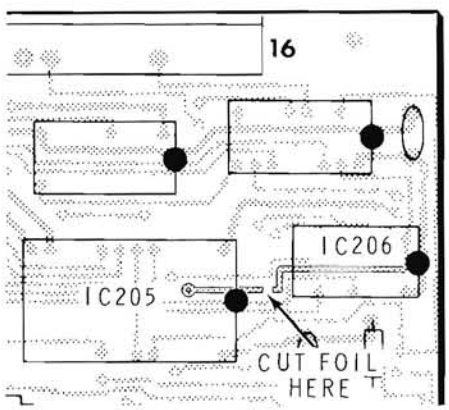


Figure 1

( ) Refer to Figure 2. Except for pins 7 and 14, bend all the leads of the 7402 IC out 30° as shown. Pins 7 and 14 should still be perpendicular to the bottom of the IC.

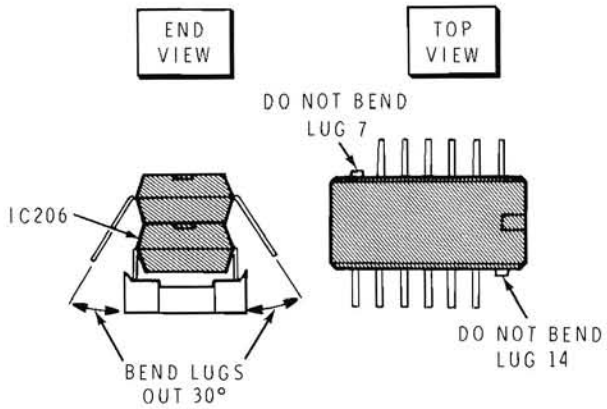


Figure 2

( ) Orientate the IC as shown in Figure 3. Place the IC directly on top of IC206 as shown in Figure 2. Solder pins 7 and 14 of the top IC to pins 7 and 14 of IC206.

( ) Connect a wire between pin 2 of the top IC to point X on the board.

( ) Connect a wire between pin 3 of the top IC to point Y on the board.

( ) Connect a wire between pin 1 of the top IC to point Z on the board.

NOTE: Point Z is located on the side of the cut that you made nearest IC205.

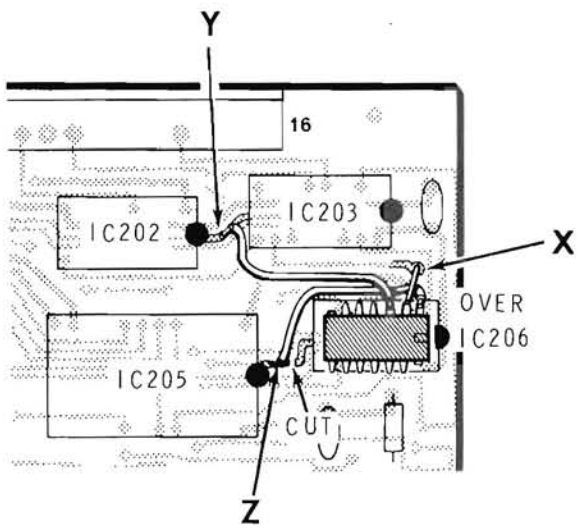


Figure 3

( ) Plug the board onto the connector pins in the H9.

( ) Put the bottom cover on.

( ) Plug the AC line plug into an AC outlet.

The modification is now complete.

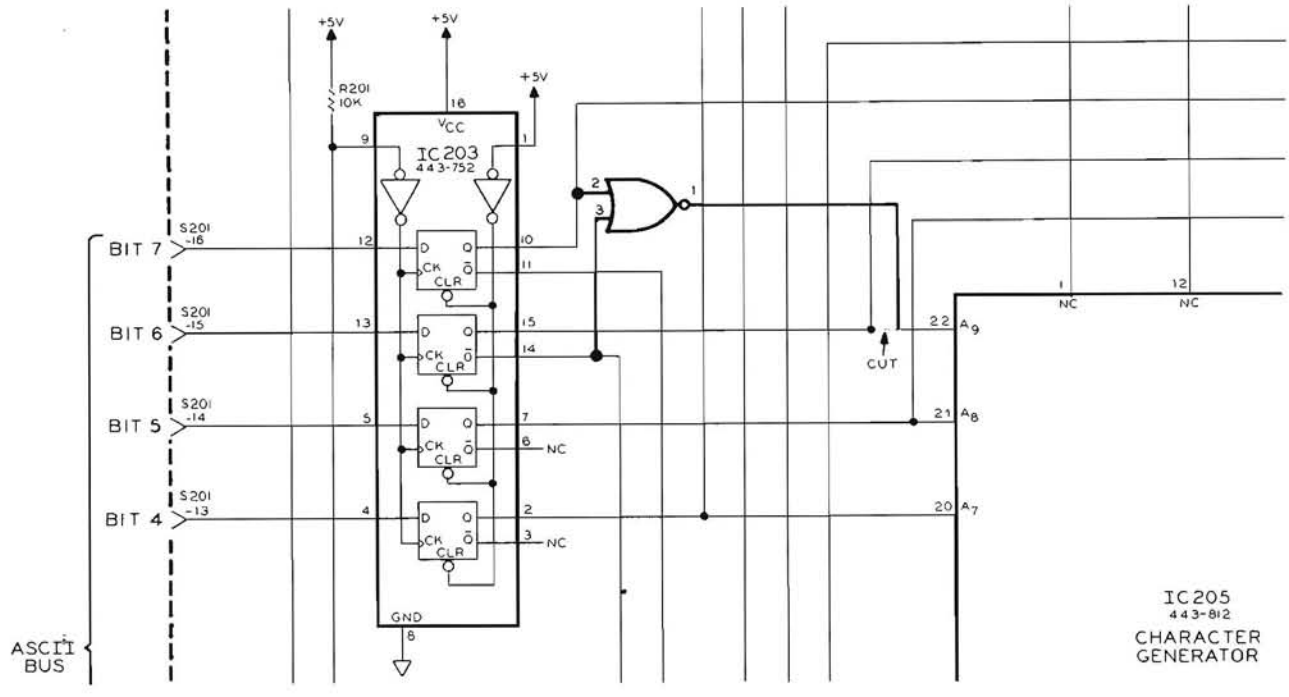
Check Out

The following method can be used to verify that the modification works:

1. Load BASIC into your computer.
2. Enter the following program:

```
10 FOR X = 97 TO 122

20 PRINT CHR$(X)

30 NEXT X
```

3. Run the program. The terminal should print A to Z in upper case format, then stop.



SCHEMATIC OF H9 MODIFICATION

MORE ABOUT THE H17 MINIFLOPPY DISK SYSTEM

The H17 is a single or dual-drive floppy disk system for the H-8 digital computer, which utilizes a standard 5.25" floppy disk. Each diskette contains 40 tracks; each track consists of ten sectors of 256 bytes each, giving a raw capacity of 102.4 kilobytes of storage on each diskette.

The basic system consists of a single WANGCO model 82 drive-unit and a power supply, mounted in an attractive cabinet which is color-coordinated with the rest of the Heath computer line. The H-17 can be expanded to a dual-drive system by simply purchasing and installing the second drive-unit in the cabinet. Each drive mounts a single diskette spinning at 300 RPM, yielding a data transfer rate of 12800 bytes per second. The head track-to-track step time is guaranteed at 30 milliseconds per track max, but most units perform at 12 milliseconds or better. The operating system provides facilities to determine and utilize the fastest reliable rate for each system. This make the H17 the fastest 5.25" floppy available today. See "SEEKS and the SINGLE DISKETTE" below for details.

The heart of the H-17 is HDOS, the Heath Disk Operating System. This is the most sophisticated hobbyist system currently available. HDOS provides an advanced file structure, full device independence, and dynamic space allocation.

The H17 disk system includes case and power supply, and yet sells for less than todays leading 5.25" disk system! Its cost-effectiveness is incomparable, because it offers, at a price the hobbyist can afford, software features which have heretofore been available only in large main-frame systems.

HDOS FEATURES

HDOS supports dynamic space allocation, complete device independence, utilities, and diagnostics. HDOS also provides full support of one-drive systems.

DISK SPACE MANAGEMENT

H17 HDOS provides dynamic space allocation for your files. You never have to guess at the size of a file, never get blown off for guessing low, never waste space for guessing high! HDOS's expanded 40 tracks of storage area, coupled with the no-wasted-space file management techniques, gives you the most usable storage space for your dollars. And, of course, the disks never need squeezing or compressing. Naturally, HDOS contains special provisions to prevent the disk fragmentation possible with many other systems. This, plus the H17's accelerated seek times, provides the fastest 5.25" floppy system you can buy.

SYSTEM FLEXIBILITY

It's impossible for any manufacturer to anticipate everything you will be doing with your system. In fact, it's impossible for you to anticipate everything you will be doing with your system! That's why we've made HDOS so extremely flexible. When you get a great new idea, you'll find HDOS a help, not a hinderance.

1. **Full Device Independence**
   Full device independence means that any valid device may be used by any program, be it HEATH supplied or written by you. And when you add more devices to your system, your current and new programs can automatically make use of them, without any change. Not enough room on your disk for an assembly listing file? Simple, just write the listing file directly to your hard-copy device. Want a printout of a BASIC program you're working on? Simple, just tell BASIC to write your program text to a file, and give it the file name of your printer!

2. **Memory Expandability**
   Adding more memory in the future? HDOS automatically takes advantage of it. The system itself, BASIC, the editor and all other programs make immediate use of the new space. No being hemmed in by a one-size-fits-all system!

3. **Console Configuration**
   Got a CRT console terminal and want to backspace? Got a TTY terminal and can't backspace? Got a lower-case DECWRITER? Your terminal need fill characters? No problem! HDOS allows you to configure your system to make optimal use of what you have! And when you add new peripherals in the future, HDOS will instantly adapt to them, too.

4. **Support for Home-Brew Devices**
   Future releases of HDOS will provide support and instructions for writing your own device drivers. This means that anything that you can possibly interface to the H8, via serial, parallel, or even kludge cards, you can use with the HDOS system without patching or modifying HDOS. Naturally, all programs, whether written by you or Heath, will run with your new device. Of course, HEATH will provide device drivers for all new HEATH products, as well.

SYSTEM CAPABILITY

Flexibility means little unless the system has the capability to do your job, and thats where HDOS really comes through. First, HDOS includes the powerful Extended Benton Harbor BASIC, with added file handling capability. Create, read and write with full device independence right from BASIC! Of course, HDOS includes a powerful ASSEMBLER, TEXT EDITOR, and DEBUGGER. The text editor is useful for editing BASIC programs and manuscripts, too. HDOS includes TEST17, a diagnostic exerciser for the H17 drives, a first for floppy operating systems! TEST17 performs drive validity and speed checkouts as well as certification of diskettes. HDOS provides full support of system generation, file copying, and file backups for one-drive systems via SYSGEN and ONECOPY. One-drive operation allows you to get running now and expand later. For two-drive owners, if one drive should ever fail, you can continue operation! Finally, FLAGS allows you to write-protect your files for safety, as well as to free up disk space by deleteing unwanted parts of the operating system.

THE FUTURE

For most people, a computer system is a major investment. HEATH is a stable company, with an excellent reputation and tens of thousands of satisfied customers. Your system will still be running years from now, and HEATH will still be there to support it. As an H17 owner, you can look forward to exciting new languages, applications, and utilities at affordable prices. New software is expensive to produce, and only HEATH has the resources and commitment for full scale software development, the kind you want for your system.

SEEKS and the SINGLE DISKETTE
(or)
What Makes the H17 Run so Fast?

The secret of the high-speed performance of the H17 diskette drive is its very fast seek time. As discussed earlier, the ratio of seek time to rotational latency is important. Since 5.25" floppies rotate at close to the same speed as full sized floppies, but usually seek much slower, the 5.25" floppies have a lot to gain from accelerated seek performance.

Most competitive units seek at 40 milliseconds, track-to-track. Some of the newer designs seek at 30 milliseconds. The actual speed that the HEATH H17 achieves depends upon the individual drives, but most will step faster than 12 milliseconds track-to-track. Assuming a 40 track diskette, a mathematically random seek requires:

40 ms: 520 (seek) + 100 (latency) = 620 ms access time
30 ms: 390 (seek) + 100 (latency) = 490 ms access time
12 ms: 156 (seek) + 100 (latency) = 256 ms access time

This means that an H17 drive performs a guaranteed minimum of 1.25 times faster than the biggest seller, and most will perform at least 2.4 times faster! Of course, most seeks are for shorter periods than the 13 track mathematical average. But then, most H17 drives will seek faster than 12 ms.

The HEATH company has always offered more for your money, and the H17 upholds this fine tradition, with

- Guaranteed High Performance
- HEATH Engineering Know-How
- HEATH Software Expertise, and
- The BEST PRICES AVAILABLE

STRAIGHT TALK ABOUT "SEEK TIME" VERSUS "ACCESS TIME"

The speed with which data can be accessed from a disk can be computed from two figures: track access time, and rotational latency. "Track access time" is the amount of time it takes the disk's read/write head to reach the proper track. "Rotational Latency" is the amount of time you have to wait for the data you want to rotate under the disk head. This means that the time it takes to get a piece of data from the disk is the seek time added to the rotational latency time. This is the critical factor that is conveniently ignored by some manufacturers of fast "track access" disk drives: by itself, *a drive's seek time is meaningless, it is the access time that is important.* All full sized floppy disks rotate at 360 RPM, giving an average rotational latency of about 85 milliseconds (1/2 a turn). Thus, if it takes a "slow" drive 30 milliseconds to reach a track, and a fast one only 2 milliseconds, the slow drive will reach the data (on the average) in 30+85 = 115 milliseconds, whereas the fast drive will reach the data (on the average) in 2+85 = 87 milliseconds. Thus, the fast drive is not 15 times faster, but only 1.3 times faster.

This means that the widely shown head motion demonstrations, where a disk unit is seen rapidly seeking back and forth, are misleading. The drives do seek that rapidly, but they do not access the data that rapidly. *The demonstration is showing head motion, not access time.*

Since all drives suffer from the same rotational latency, speeding up the seek time soon reaches a point of diminishing returns, where cutting the seek time in half only reduces the access time 10%. As an example, a well-known "voice-coil" drive advertises and end-to-end seek time of 100 ms. A typical worm-gear/stepper drive, at 6 ms per step, requires 462 milliseconds. The advertisements thus claim that the drive performs 4.6 times faster! An actual calculation of a random access (seek of 25 tracks) shows:

Voice Coil:
45 ms (seek) + 88 (latency) = 133 ms random access

Stepper:
150 ms (seek) + 88 (latency) = 238 ms random access

Thus, the voice coil systems do not perform 4.6 times faster, as it might at first appear, but only 1.8 times faster. Also note that this timing is for a mathematically "random" access. Most disk accesses are to read more data from some open file and do not involve a head motion. Operating systems group files on the disk together, so that when a head motion is necessary, it averages much less than the mathematical prediction. Thus, a typical read does not move the head much, but still does suffer from rotational latency. Therefore, the "diminishing returns" effect of the access time formula reduces the advantage of the fast seek time still further.

For example, assume that you have a payroll file which occupies 30 tracks on a floppy disk. A random lookup of some item in this inventory will require an average seek of 10 tracks. In this case, the figures are

Voice Coil:
30 ms (seek) + 88 (latency) = 118 ms access

Stepper:
60 ms (seek) + 88 (latency) = 148 ms access

giving the voice coil drive just a 25% speed advantage.

H17 PRICES

Description	Model No.	Price	Delivery
Wired Single Drive System	WH17	$675.00	End of June
Optional Second Drive	H17-1	$295.00	End of June
5 Blank Diskettes	H17-2	$ 25.00	End of June
Kit Single Drive System	H17	$530.00	End of August
Operating System Software	H8-17	$100.00	End of June

NOTE: A wired dual drive system consists of (1) WH17 and (1) H17-1. A kit dual drive system consists of (1) H17 and (1) H17-1. A WH17 or H17 is provided with a single disk drive which is mounted in the left hand side and a removable panel on the right. To install the second drive, you simply; remove the panel, bolt the drive in place, push the connectors (supplied) onto the back of the new drive, cut some jumpers on the drives and your done. Of course, all instructions are provided.

THE MAIL BOX

Dear HUG:

The following example doesn't seem to execute properly

```
LOOP  CALL  SUB
      JMP   LOOP
      HLT
 SUB  RET
```

In this example, if a clock interrupt occurs and the saved value of the PC points to the SUB label, the following action takes place; the "check for HLT" portion of the clock interrupt processing (beginning at address 000.274A) will fetch the saved PC, decrement it, fetch the instruction at PC-1, find it is a HLT and jump to the ERROR routine.

The problem is not severe once it is understood, however, I thought you might appreciate having it brought to your attention if you do not already know about it.

Sincerely,
J. R. Holden
Chicago, Il 60634

The situation described is correct. It is taken care of by the UO.HLT bit which is to be placed in .MFLAG (040.010). This routine should appear early in your program to disable HLT processing.

```
LDA   .MFLAG
ORI   200Q
STA   .MFLAG
```

Consider also, that this routine will ignore all HLT instructions in your program.

Bob

Dear Bob,

I'd like to pass on a suggestion to H9 owners concerning a problem that has disabled my H9 unit.

The problem lies in the tension springs which hold the window in place. These springs have a tendency to dislodge and drop onto the circuit board, shorting it out.

A technician at the Cleveland Heathkit Electronic Center has encountered this problem and suggests that the springs be held in place by electrical tape. I think that H9 owners should be aware of this possibility and tape them as a safeguard.

John S. Wilczewski
Cleveland, OH

Good suggestion. I have noticed that if the springs are pushed in as far as they will go, they stay in place pretty well. If not, then by jarring the unit, they may pop out. I guess it pays to be safe.

I might add a suggestion of my own. You may notice that the terminal is deep and some narrow tables may not handle the terminal properly. The front legs can easily be removed and placed in the center where the two bottom place center screws are. Since most of the weight is in the rear, the H9 balances beautifully and the key board can overhang the table.

Bob

Dear Bob,

I am having a problem regarding Heath Extended BASIC version #10.01.02. On page 5-45 of the software reference manual supplied with my H8 computer, it states "if the result of the expression is false, control passes to the next line or to a statement separated from the IF-THEN statement by a colon". I observe that when the expression is false, the control will **not** pass to a statement following a colon; rather, control passes to the statement on the next line. If the result of the expression is true, the entire line will be executed, even if there are multiple statements.

Ronald N. Santosuosso
Vista, CA 92083

Your observation is true and future manuals will be changed.

Multiple commands per line was not part of the original Dartmouth BASIC and therefore each extension of "BASIC" handles the situation differently.

Our methods provide for easy program reading and execution. It works particularly well with statements like:

```
IF A=0 THEN B=0:C=0:D=0
```

Where, if A is not 0 then neither will be B, C, or D (unless they are already) since the rest of the line will not be executed. This will not work:

```
IF A=5 THEN GOTO 100: GOTO 200
```

Because, if A is equal to 5 then line 100 is executed next. If A is not equal to 5, the rest of the line is aborted. To handle this, you will have to put GOTO 200 on the next line, such as:

```
10 IF A=5 THEN GOTO 100
20 GOTO 200
```

You will have to keep this in mind when writing programs using this feature.

Bob

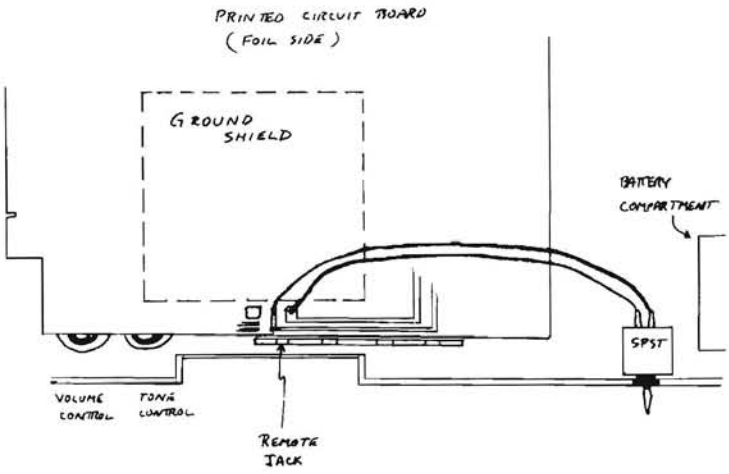
Dear Sirs:

I am enclosing two items that may be of interest to the members of HUG. One is a hardware modification to the ECP-3801 cassette tape recorder to add manual control while it is under computer control. The other is a display demonstration program for the H8 microcomputer. This program is quite short and may be entered via the keypad of the H8.

INSTRUCTIONS TO MODIFY HEATH'S ECP-3801 CASSETTE TAPE RECORDER

Parts Needed:
(1) SPST toggle switch (Radio Shack #275-324)
(2) 6" hook-up wires

1. Remove the rear cover of the recorder.
2. Drill a hole on the side wall near the battery compartment.
3. Install the SPST switch.
4. Solder the two 6" hook-up wires to the two lugs of the switch.
5. Remove the ground shield.
6. Solder the two hook-up wires from the switch to the foil side of the printed circuit board as shown.
7. Install the ground shield.
8. Replace the rear cover.
9. Leave the switch in the off position. The computer will control the recorder same as before.
10. When you want to use the Rewind or Fast Forward features on your recorder, turn the switch on and use the appropriate recorder controls. Then turn the switch off.



DISPLAY DEMONSTRATION PROGRAM

This program will make the front display of the H8 light up in a seemingly random manner (all identical) according to a binary value in register B which is being decremented by one. It may be entered via the keypad on the H8.

1st display	8.8.8. 8.8.8. 8.8.8.
2nd display	all blank
3rd display	
4th display	
5th display	
6th display	
7th display	
8th display	
9th display	
.	
.	
.	
128th display	

and then repeats itself with all decimal points turned on.

DISPLAY DEMONSTRATION PROGRAM

040 100	076	040 123	016
040 101	002	040 124	001
040 102	062	040 125	076
040 103	010	040 126	377
040 104	040	040 127	315
040 105	006	040 130	053
040 106	000	040 131	000
040 107	041	040 132	015
040 110	013	040 133	302
040 111	040	040 134	125
040 112	016	040 135	040
040 113	011	040 136	005
040 114	170	040 137	302
040 115	167	040 140	107
040 116	043	040 141	040
040 117	015	040 142	303
040 120	302	040 143	105
040 121	115	040 144	040
040 122	040		

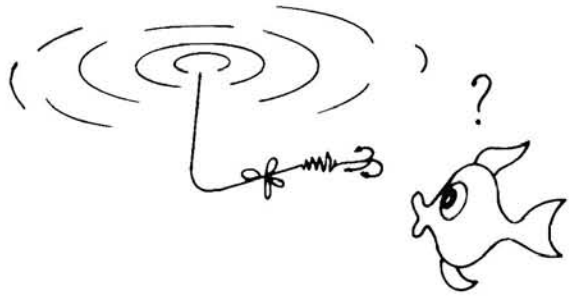
W. Wayne Frick
Staunton, Virginia

Larry Steckler, Editor for Radio Electronics Magazine called to point out that his magazine was not included in the list of suggested reading as printed in our first REMark issue. We want to acknowledge that RE is certainly a recommended magazine and in fact one of the first which included articles about computers.

Before any computer magazines were around, classic articles on the TV typewriter and the famous Mark 8 (an 8008 machine) were among the other many computer related articles published. It is still going strong in this area.

Radio Electronics simply got left out during the paste up process of REMark. Our apologies.

Subscription address:
Radio Electronics
Subscription Service
Box 2520
Boulder, CO 80322



BYTE SIZE

H8 Digital Computer
Wires Pinched Between Boards

Each of the boards that plug into the bus contain their own regulators. The wires to the regulators pass over the heat sink bracket which support them. It is important to lay these wires neatly, one along side the other as they pass over the bracket exactly as shown in the construction manual. There is plenty of room.

We have noticed in some units coming in for service that this wiring is not done as described. Twisted, braded or cable tied wiring is unacceptable and will not pass between the boards properly. The wires may be pinched and the insulation will be pierced.

H8 Digital Computer
Intermittant DUMP

There is about a 3 second delay between the time the DUMP key is pressed and the displays begin to increment.

If the count begins immediately after the key is pressed and runs faster than normal, you can make the following changes. Make the changes in sequence and stop when the problem is cured.

See Figure 1 (seriel I/O, H8-5).

1. Interchange IC131 with IC132. The 74LS240 should finally be in the IC132 position.

2. Replace IC127 with a 7414.

3. Cut the foil where shown. Install a 100 μh toroid coil across the foil cut. Remove C126 (.1 μfd) and discard it.

H9 Video Terminal
"OFF LINE" Ineffective (RS-232)

Check the H9 by loading a program into your computer which continuously outputs characters to the terminal. One way to do this is by loading BASIC and enter the following program:

```
10 PRINT "*";
20 GOTO 10
```

IF characters are still being received and displayed on the screen when the "OFF LINE" switch is depressed, make the following changes on the I/O card in your H9:

Change	From	To
R618	4.7K Ω	10K Ω
R617	10K Ω	39K Ω

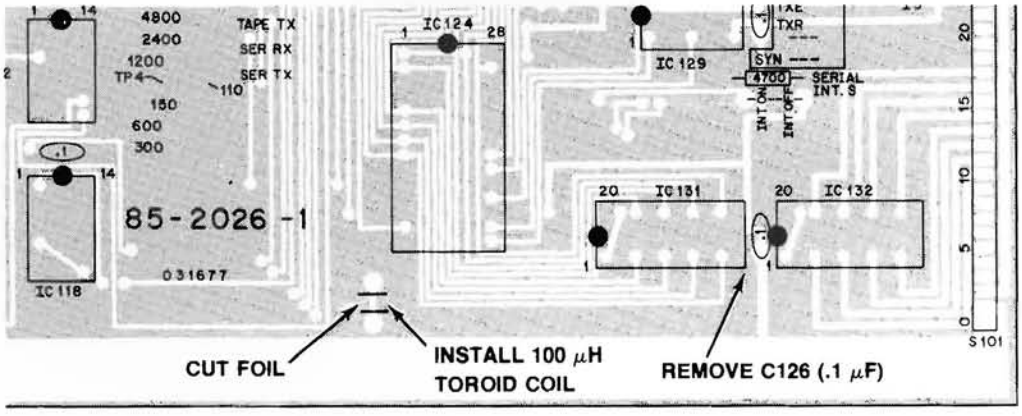


Figure 1

H9 Cursor Skips Intermittently

In some units you may notice that the cursor skips intermittently. Solder a 36 pF capacitor from S705-7 to ground on the TPU board to cure this problem.

CLASSIFIED ADS

A reminder: Ad placement is free to all members. Send your ads typewritten to us. Allow approximately 10 weeks for the ad to appear. We reserve the right to reject any ads that do not serve the best interest of Heath Co., HUG or its members.

FOR SALE

Heath H9 Video Terminal
Up and running, $550. I ship.
Wm. Thornburg, 400 E Jackson,
Desoto, IL 62924

H11 OWNERS

A recent mailing was made to all H11 owners to inform them to install some resistors to improve the system performance. If you have not received this mailing you should be aware of the following additions:

On the H11-5 serial card

Add a 470 Ω 1/4W resistor from pin 3 to 4 of IC9A.

Add a 1 KΩ 1/2W resistor from pin 13 to 24 of IC5.

On the H11-2 parallel card

Add a 470 Ω 1/4W resistor from pin 11 to pin 14 of IC7D.

Add a 1 KΩ 1/4W resistor from pin 13 to 24 of IC8.

LOCAL USERS GROUPS

CHICAGO AREA H11 OWNERS, TAKE NOTE.

Dear Bob,

I wish to start an H11 User Group in Chicago. Please print the following in the next REMark:

Chicago Area H11 User Group Forming. Interested persons are asked to send their name, address, phone number, suggestions and recommended location for a meeting site to:

Allen Cohn
1472 E. 56th Street
Chicago, IL 60637

Thank you,
Allen Cohn

We are happy to print such a request for you. HUG encourages such an undertaking and will do all it can to lend a hand in your support.

Bob

TEXAS COMPUTER OWNERS

If you are interested in becoming part of a local group in Texas, contact:

Stanley Weiss
701 Oak Hill Dr.
Killeen, TX 76541

LOS ANGELES HEATH USERS

Can contact:

Robert Fisher
1011 N. Palm Ave #310
W. Hollywood Ca. 90069

If you are interested in starting a local users group. Bob is willing to devote some time to the mutual benefit of local Heath computer products owners.

H8 Disassembler Program

Contributed by: Michael Jones
110 Twinbrook
Perrysburg, Ohio 43551

Runs in H8 under Ex B.H. Basic Ver. 10.01.xx at least 16K of memory.

Introduction

A disassembler, as you might know, is a utility program which converts raw machine code into a readable assembly language program listing. Such a program is handy in trying to decipher a large unknown program. It is used by programmers who are curious about how a particular program works, or in cases where one would like to locate an instruction to change in order to modify the program's performance.

Disassemblers hold a particular fascination for the newcomer because they appear to hold the key to "crack" a program written by someone else. This appeals to the little "criminal" found in most everyone. On the more positive side, this utility is useful in deciphering small sections of machine code for the purpose of education or satisfying curiousity.

Disassembler Limitations

In order to use a disassembler properly, you must already understand assembly language programming. You will soon find, if you have not used a disassembler before, that contrary to popular belief, disassemblers do not disassemble programs completely. There are some limitations. If you understand the limitations, the disassembler becomes more useful to you.

The sample run shows what happens when you try to disassemble the monitor ROM located at low memory in the H8. A listing of this program is located in your H8 manual in the PAM-8 section for your comparison.

Here is what happens. After typing RUN it takes about 20 seconds for the address prompt to return. You then enter the beginning and ending address, in decimal, and hit return. The program does a nice job until it gets to address 33, at which point incorrect disassembly occurs. If you are following the program listing in the H8 manual you will see the this address contains a DB directive and is not an actual instruction. In this case, it contains the actual Heath part number for the ROM. The programmer who wrote the disassembler anticipated this and provided an ASCII representation for byte located at each address. Notice 44313 is located in these 5 addresses. The next addresses are disassembled improperly because the disassembly is skewed.

A Disassembled Program

A sample run appears just after the listing. Examining one line of disassembly, the first group of numbers contain the address being decoded. This address is in split octal with no leading zeros. The second group of numbers is the actual octal number in that address. The third character, if any, will be the ASCII character represented by the byte in that address. Sometimes it is unmeaningful in cases where the program disassembles properly but very often provides a good key when the disassembly goes awry. The fourth group of characters is the operator (mnemonic) which is the assembly language equivalent of the machine language byte in this address.

The last number indicates a 1, 2 or 3 byte instruction.

By now, you should be able to see the value and limitations of a disassembler. To use it properly, you will have to examine small sections at a time to see if it makes sense. If not, you will have to restart it at a new address.

So go ahead and disassemble Heath Basic if you would like. For practice you can disassemble the console driver, for which you already have a listing in your H8 manual to compare against.

Bob

```
00010 DIM A$(255),L(255)
00020 FOR I=0TO 255:L(I)=1:NEXT
00025 READ I:IF I=0THEN 30
00027 READ A:L(I)=A:GOTO 25
00030 FOR I=0TO 255:READ A$(I):NEXT
00060 INPUT "ENTER FROM, TO LIMITS";F,T:V=F
00080 N=V:GOSUB 1000:PRINT Q;:A=PEEK(V):FOR I=1TO L(A):N=PEEK(V-1+I):GOSUB 1000
00090 PRINT TAB(15);Q;CHR$(PEEK(V-1+I));:IF I=1THEN PRINT A$(A),L(A)
00100 IF I<>1THEN PRINT
00110 NEXT :V=V+L(A):IF V<=T THEN 80
00120 END
01000 Q1=INT(N/16384)*100000:N=N-((Q1/100000)*16384)
01010 Q2=INT(N/2048)*10000:N=N-((Q2/10000)*2048)
01020 Q3=INT(N/256)*1000:N=N-((Q3/1000)*256)
01030 Q4=INT(N/64)*100:N=N-((Q4/100)*64)
01040 Q=INT(N/8)*10:N=N-((Q/10)*8):Q=N+Q+Q1+Q2+Q3+Q4
01050 RETURN
05000 DATA 1,3,6,2,14,2,17,3,22,2,30,2,33,3,34,3,38,2,42,3,46,2,49,3,50,3,54,2
05010 DATA 58,3,62,2,194,3,195,3,196,3,198,2,202,3,204,3,205,3,206,2,210,3
05020 DATA 211,2,212,3,214,2,218,3,219,2,220,3,222,2,226,3,228,3,230,2,234,3
05030 DATA 236,3,238,2,242,3,244,3,246,2,250,3,252,3,254,2,0
10000 DATA "NOP","LXI B","STAX B","INX B","INR B","DCR B","MVI B","RLC"
10010 DATA "?","DAD B","LDAX B","DCX B","INR C","DCR C","MVI C","RRC"
10020 DATA "?","LXI D","STAX D","INX D","INR D","DCR D","MVI D","RAL"
10030 DATA "?","DAD D","LDAX D","DCX D","INR E","DCR E","MVI E","RAR"
10040 DATA "?","LXI H","SHLD","INX H","INR H","DCR H","MVI H","DAA"
10050 DATA "?","DAD H","LHLD","DCX H","INR L","DCR L","MVI L","CMA"
10060 DATA "?","LXI SP","STA","INX SP","INR M","DCR M","MVI M","STC"
10070 DATA "?","DAD SP","LDA","DCX SP","INR A","DCR A","MVI A","CMC"
10080 DATA "MOV B,B","MOV B,C","MOV B,D","MOV B,E","MOV B,H","MOV B,L","MOV B,M","MOV B,A"
10090 DATA "MOV C,B","MOV C,C","MOV C,D","MOV C,E","MOV C,H","MOV C,L","MOV C,M","MOV C,A"
10100 DATA "MOV D,B","MOV D,C","MOV D,D","MOV D,E","MOV D,H","MOV D,L","MOV D,M","MOV D,A"
10110 DATA "MOV E,B","MOV E,C","MOV E,D","MOV E,E","MOV E,H","MOV E,L","MOV E,M","MOV E,A"
10120 DATA "MOV H,B","MOV H,C","MOV H,D","MOV H,E","MOV H,H","MOV H,L","MOV H,M","MOV H,A"
10130 DATA "MOV L,B","MOV L,C","MOV L,D","MOV L,E","MOV L,H","MOV L,L","MOV L,M","MOV L,A"
10140 DATA "MOV M,B","MOV M,C","MOV M,D","MOV M,E","MOV M,H","MOV M,L","HLT","MOV M,A"
10150 DATA "MOV A,B","MOV A,C","MOV A,D","MOV A,E","MOV A,H","MOV A,L","MOV A,M","MOV A,A"
10160 DATA "ADD B","ADD C","ADD D","ADD E","ADD H","ADD L","ADD M","ADD A"
10170 DATA "ADC B","ADC C","ADC D","ADC E","ADC H","ADC L","ADC M","ADC A"
10180 DATA "SUB B","SUB C","SUB D","SUB E","SUB H","SUB L","SUB M","SUB A"
10190 DATA "SBB B","SBB C","SBB D","SBB E","SBB H","SBB L","SBB M","SBB A"
10200 DATA "ANA B","ANA C","ANA D","ANA E","ANA H","ANA L","ANA M","ANA A"
10210 DATA "XRA B","XRA C","XRA D","XRA E","XRA H","XPA L","XRA M","XRA A"
10220 DATA "ORA B","ORA C","ORA D","ORA E","ORA H","ORA L","ORA M","ORA A"
10230 DATA "CMP B","CMP C","CMP D","CMP E","CMP H","CMP L","CMP M","CMP A"
10240 DATA "RNZ","POP B","JNZ","JMP","CNZ","PUSH B","ADI ","RST 0"
10250 DATA "RZ","RET","JZ","?","CZ","CALL","ACI","RST 1"
10260 DATA "RNC","POP D","JNC","OUT","CNC","PUSH D","SUI","RST 2"
10270 DATA "RC","?","JC","IN","CC","?","SBI","RST 3"
10280 DATA "RPO","POP H","JPO","XTHL","CPO","PUSH H","ANI","RST 4"
10290 DATA "RPE","PCHL","JPE","XCHG","CPE","?","XRI","RST "
10300 DATA "RP","POP PSW","JP","DI","CP","PUSH PSW","ORI","RST 6"
10310 DATA "RM","SPHL","JM","EI","CM","?","CPI","RST 7"


*RUN
ENTER FROM, TO LIMITS? 0,45
 0               21        LXI D          3
                 371 Y
                 3
 3               41 !       LXI H          3
                 12

                 40
 6               303 C      JMP            3
                 73 ;
                 0
```

```
11              315 M       CALL            3
                132 Z
                0
14              26          MVI D           2
                0
16              303 C       JMP             3
                201
                0
21              315 M       CALL            3
                132 Z
                0
24              32          LDAX D          1
25              303 C       JMP             3
                244 $
                1
30              303 C       JMP             3
                45 %
                40
33              64 4        INR M           1
34              64 4        INR M           1
35              64 4        INR M           1
36              61 1        LXI SP          3
                63 3
                303 C
41              50 (        ?               1
42              40          ?               1
43              100 @       MOV B,B         1
44              112 J       MOV C,D         1
45              107 G       MOV B,A         1
46              114 L       MOV C,H         1
47              100 @       MOV B,B         1
50              303 C       JMP             3
                53 +
                40
53              365 U       PUSH PSW        1
54              257 /       XRA A           1
55              303 C       JMP             3
                143 C
                2
END AT LINE  120
*
```

BUFFALO AREA H8 OWNERS MEET

Pete Perrello talks with one of the early arrivals at the first gathering of the Heath users group of the Greater Buffalo area.

John Hodge holds a real software donation! Baked by one of the H8 families, this cake fitting of the occasion was programmed into the session during a coffee interrupt.

Error horns, tick counters, and klingons mixed it up with ham sandwhiches and chili, as Buffalo's first H8 users group got off to a successful start. Over 20 enthusiastic Heath owners were hosted by John Hodge, in his Orchard Park residence Sunday April 16th, including Pete Perrello, coorganizer of the event.

Three complete H8 systems were in operation during the day long meeting, one system loaned with Heath sales representative, Alex R. Crinzi from Heath's retail store in Amherst, N.Y. Heath getting kudos for compiling all the owners names and addresses from sales slips.

A good mix of hardware, and software interests kept things lively all day with people meeting people, and discovering new applications. Buffalo's new mass transit routing may have had its birth on a Heath!

After 7 seemingly short hours, the rag chewing came to a close, and a new twist to a lottery was demonstrated. A random number program with all the names entered came up with a winner delighted to receive our first door prize. A basic course EC-1100 donated by Heath Amherst was won by Jack Thompson of Kenmore, N.Y.. The group agreed to quarterly meetings, and will meet again in July.

For more information about this group, contact:

P. Perrello
6260 Seneca St.
Springbrook, N.Y. 14140

FACTORY ASSEMBLED VERSIONS OF ALL HEATHKIT COMPUTER PRODUCTS TO BE AVAILABLE SOON

The Heathkit Summer Catalog will announce the availability of a factory assembled version of the H11 computer to be followed by availability of the complete line of Heathkit computer products in assembled form by September of this year.

These assembled Heath computer products will appeal to the non-hardware oriented hobbyist or to the small businessman who is anxious to get his system in operation as quickly as possible. They will also be of particular interest to those HUG members who are considering selling small computer systems to businesses. Heath customer feedback indicates a large market for personal computers among small businesses and many hobbyists have the capability to put together a team who can design, program, install, and service such systems on a custom basis in their localities.

Assembled Heath computer hardware and system software is ideally suited to the small business application, particularly with the introduction of the new WH-17 Floppy. For information on a new limited distributor sales program for these Heath assembled computer products, contact Al Robertson, Director of Sales, here at Heath Company (telephone: 616-982-3206).

NEW HEATHKIT ELECTRONIC CENTER OPENS IN TEXAS

The newest Heathkit Electronic Center was opened in San Antonio, Texas on April 27, 1978 at 7111 Blanco Rd. 78216 and is managed by Mr. Al Eljaiek. If you are in the area, stop by and see this latest showcase of Heathkit Products and Accessories.

Heathkit Electronic Centers (owned and operated by Schlumberger Products Corporation) are in major market areas to serve you. Each of the 51 locations is a complete one-stop shopping center for Heathkit Products, featuring a showroom full of assembled demonstration models and a complete stock of all available kits. Plus, parts and service are available at each location.

Complete operational displays of all available computer products are on hand at each Heathkit Electronic Center for your inspection. Stop in soon at the center nearest you!

What it is!

In REMark Number 1 we offered a Heath universal programming pad to anyone who guessed the idea behind the HUG logo. We received several hundred guesses of which about half were correct. Although the rules stated that a correct answer was necessary, pads were sent to anyone who submitted any kind of an answer, just for being active.

What does the logo represent? Our stylist used the side view of one
H8 and placed i

NEW COMPUTER COURSE Uses Heathkit H8 and H9 System

National Technical Schools has recently announced new courses in computers. The courses include the Heathkit H8 Computer and Software, H8-5 Serial IO and Cassette Interface, plus H9 Video Terminal. For further information contact:

National Technical Schools
4000 South Figueroa St.
Los Angeles, Ca 90037

The Second Software Contest

This contest is devoted to Amateur Radio, since a significant number of computer enthusiasts indicated they were licensed and they also requested programs related to their "other hobby."

Rules are simple.

The contest is open to ET-3400, H8 and H11 owners. You must submit a source and object tape, if programmed in assembly language on the H8 or H11, or a "dump" of the program if it is in BASIC. Supply a hard copy listing and a sample run if possible. Write a short synopsis of how the program operates. Submit your program using the official HUG program submittal form found in your HUG Handbook.

The value of the program is based on:

- Program use and popularity as related to Amateur Radio.
- Program Stability.
- Ease of program use.
- Completeness of work.
- Use of minimum external hardware.
- Quality of documentation.

All winners will be determined by a committee at HUG and decisions are final. Schlumberger/Heath employees and relatives are not eligible for prizes. All programs will be considered for HUG library entry.

Deadline for submitting programs is July 31, 1978.

Winners to be announced by separate mailing.

1st Place HR-1680 High-performance SSB/CW Receiver (wired)
Plus HS-1661 Matching Speaker

2nd Place HW-8 QRP Transceiver
Plus HWA-7-1 Power Supply

3rd Place HD-1410 Keyer

Heath
User's
Group
Hilltop Road
St. Joseph MI 49085