REMark®

Volume 7, Issue 3 • March 1986
P/N 885-2074 Issue 74

$2.50

Official magazine for users of HEATH ZENITH computer equipment.

IBM AT Compatible Personal Computer only $2899

HS-241

NEW IBM AT Compatible Advanced Personal Computer Kit

Priced Hundreds of Dollars Less Than the IBM AT!

Only $2899.00 Less HUG Discount

Includes: • 512K RAM expandable to over 15MB • single 1.2MB 5.25" floppy disk drive • controller card for two floppy and three hard disk drives • MS-DOS operating system • 8 expansion slots • 6 MHz clock for high speed processing • compatibility with almost any PC video card for color, graphics, and high resolution video options. **Monitors from $79.95**

See the catalog for SuperSaver discounts on accessories.

Take your HUG Discount on Heathkit and Zenith Data System Products

HS-148-41 IBM-PC Compatible Computer Kit with HVM-122A Monitor

HS-148-41 IBM PC Compatible Computer with Disk Drive and 256K Memory

Kit Only $999.00

• MS-DOS operating system • accesses IBM software • optional 10 Meg, 20 Meg and 30 Meg Winchester hard disk drive available • 360K disk storage • 256K memory expandable to 640K • start-up diagnostic • famous Heath/Zenith quality

• Expand to 640K Memory for Just...$110.00

• Second Disk Drive Just $149.00

• Monitors from $79.95

ZP-150 *Personal Advanced Laptop

Sensational price on this new PAL* LAPTOP COMPUTER

7½ lb. briefcase size unit has Microsoft WORKS, WORD, MULTIPLAN, and BASIC...Modem...Calendar...Alarm...all built in. Much more, too. Act now for this unbelievable price.

Apply your HUG Discount $899.00

SYQUEST 10 MEG REMOVABLE WINCHESTER CARTRIDGE DRIVES

Includes drive, one 10 meg cartridge, controller board.

SPECIAL...$1195.00

Reg. $1395.00

FREE Installation

ZOOM/MODEM PC 1200 INTERNAL 1200/300 BPS Modem
For PC, XT, and AT computers. Hayes-compatible. Includes touch tone decoder, onboard RAM, battery backup **$299.00**

Masterpiece 4-Function Power Controller
Includes swivel base, 5 outlets, master switch, power surge protector, static ground **$99.95**

Your Choice dBase III or Framework
While quantities last **$449.00**

Smart Outlet Box with Increased Protection
Provides single switch control of up to seven electrical devices. UL approved with new, superior surge protection and noise filtering ..... **Special $49.95**

NASHUA DSDD Diskettes ..... Per Box **$12.95**

ZENITH data systems
AUTHORIZED DEALER

Fine Electronics Since 1962

Phone orders accepted.
Prices good through April 15, 1986.
Inquire about special corporate and government sales and service programs.

EXPERTS: CAD NETWORKING COMMUNICATIONS

Your TOTAL SERVICE computer center
Service • Support • Software • Accessories • User Training • Competitive Prices

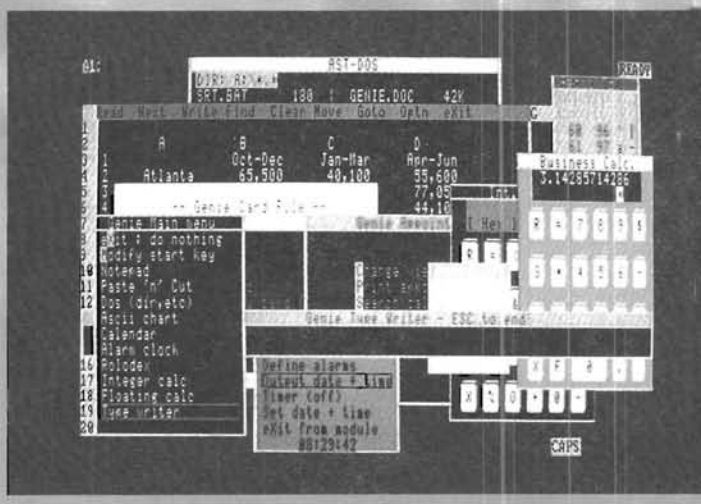
HZC-189

Heath® ZENITH®
Units of Veritechnology Electronics Corporation
Computers & Electronics

NOT PROTECTED

Genie™

Software Magic from Advanced Software Technologies



Shown here is Genie "popped up" on a Z-110 running Lotus 123. From the left are: The Genie main menu, the Genie rolodex style card file, the Genie notepad containing data cut from Lotus, the Genie DOS performing a directory command, the Genie alarm clock (at the bottom,) the Genie typewriter, Genie calendar, Genie Cut and paste, Genie Calculators, and the Genie Ascii table.

Genie is a memory resident application. This means that once you load Genie it is always available for you to use. Just hit the magic keys and Genie will appear (Shift-Shift: No function keys lost.) You can have Genie perform various tasks, and when you finish Genie goes away and you are back where you started.

Here is what you get with Genie:

- **KEY MAPPER** - Redefine any key on the fly, store long commands in a single key, and save many maps on disk.
- **TrueDOS™** - Run any MS-DOS program inside the window. FORMAT, SORT, BACKUP, compile, all without leaving your application.
- **CALCULATORS** - you get two calculators, one regular floating point calculator, and programmer's calculator for base conversions and bit operations.
- **NOTE PAD** - you'll never have to hunt for paper and pen again. Simply call up Genie's Note Pad and jot down what you need. Or expand the buffer to 56K: Instant editor.
- **CUT AND PASTE** - Cut text from any place on the screen and output it later. Cut long commands off the screen and into your KEY MAPPER, move data from your spread sheet to your word processor: Instant integration.
- **CALENDAR** - schedule appointments for any year up to 9999 keep track of expenses, search and print the calendar.
- **ROLODEX** - a name address and telephone number list that you can search any time. No limit on the number of cards. You can even output an address directly to your word processor.
- **ALARM CLOCK** - Have your Genie remind you of appointments, set alarms to ring at any time on any day. A window appears with a reminder of what each of the 8 alarms is for.
- **ASCII Table** - programmers never have to leaf through big books to find the ASCII value of a character.
- **TYPEWRITER** - Knock out quick memos any time, even send ESCAPE codes to your printer. Cut a portion of a spread sheet and paste into the typewriter for quick printouts.
- **SCREEN SAVER** - Automatic phosphor protector for your tube. Genie will even let you blank the screen manually to discourage peepers.
- **COMMAND STACK** - Lets you access up to the last 2K worth of commands you typed.

Best of all with Genie you only have to load what you use. If you don't need to use an ASCII chart don't load it. Need extra alarms? then load two alarm clocks. Your own personal Genie.

And now Genie user's have the advantage of Genie accessories, which can be installed and accessed like any Genie function - any time:

- a **Scientific Calculator** more powerful than most real calculators. Has a full range of math, science and statistical functions and very high accuracy.
- a **Spelling Helper™** To help you find the correct spelling without leaving your word processor.
- a **Print Spuffer™** (spooler/buffer) to speed and manage all your printing.
- a development pack to let you write your own pop up programs, including many routines for windowing, and source code for our Rolodex. Writing pop ups is a snap. (Please call for details)

Only Genie Gives you so much for so little.

ADVANCED SOFTWARE TECHNOLOGIES

452 West 47th Street
New York, NY 10036
(212) 247-0150

ORDERS ONLY CALL

1-800-437-4400 Ext 800

Yes please rush me the following software for my
_____ Z110/120 _____PC compat.

_____	Copies of Genie	@ $54.95	_____
_____	Spelling Helper(s)	29.95	_____
_____	Print Spuffer(s)	24.95	_____
_____	Scientific calc.	19.95	_____
_____	Genie pack (I want it all) Saving $40	$89	_____
		Sub total	_____
Check encl ☐		shipping	4.00
M/C ☐		NY Tax (8.25%)	_____
VISA ☐		TOTAL	_____

Card Number _____

Expires _____ Your Tel Num _____

Name _____

Address _____

Company _____

City/ST/Zip _____

Heath/Zenith Users' Group

REMark®

Staff

Manager . . . . . Bob Ellerton (616) 982-3867

Software Engineer . . . . . Pat Swayne (616) 982-3463

Bulletin Board and Software Developer . . . Jim Buszkiewicz (616) 982-3463

Software Coordinator . . . . . Nancy Strunk (616) 982-3838

Secretary . . . . . Margaret Bacon (616) 982-3463

Editorial and Advertising Assistant . . . . . Lori Lerch (616) 982-3794

Printers . . . . . Imperial Printing St. Joseph, MI

	U.S. Domestic	APO/FPO & All Others
Initial	$20.00	$35.00*
Renewal	$17.00	$30.00*

* U.S. Funds

Limited back issues are available at $2.50, plus 10% shipping and handling — minimum $1.00 charge. Check HUG Product List for availability of bound volumes of past issues. Requests for magazines mailed to foreign countries should specify mailing method and appropriate added cost.

Send Payment to: Heath/Zenith Users' Group
Hilltop Road
St. Joseph, MI 49085
(616) 982-3463

Official magazine for users of HEATH ZENITH computer equipment.

Volume 7, Issue 3 • March 1986

Index of Advertisers

This index is provided as an additional service. The publisher does not assume any liability for errors or omissions.

On The Cover: *Pictured is Heath Company's new Robot, HERO 2000. You won't believe some of the things it does. The article on Page 69 explains some of these exciting things.*

The H89 *SPEED* Center

4MHz mod — $45

An easy to install plug-in module. No trace cutting or soldering. Speed may be toggled with software. Includes a replacement Z80A (4MHz). Includes CP/M software support for Heath, CDR Systems and Magnolia. Call or write for info on HDOS support. Specify disk format.

6MHz mod — $59

Similar to our 4MHz modification, but increases the CPU speed to 6MHz. Requires some soldering on the CPU board. Includes a replacement Z8OB (6MHz). May require replacing additional parts. Some technical knowledge is recommended for installation. CP/M support only. Specify disk format.

CDR Super RAM 89

Make your H89 a speed whiz with this board and the included RAM drive software. With our 6MHz mod and the RAM drive, the Heath BIOS assembles (using MAKEBIOS) in less than 1½ minutes! Note our discount prices!

Main board w/o RAM - specify disk format.....$175
Expansion board w/o RAM.....$69
Main board w/512K RAM.....$239
System w/MEGABYTE RAM.....$369
Clock option.....$45
SCSI option.....$85

CDR Controllers

Double density disk controllers. Includes board, instruction manuals, ROMs and software. Allows control of 5 and 8 inch drives. Includes BIOS source code and some useful utilities. Please include your CP/M s/n when ordering.

For the H89 FDC-88OH.....$349

REP3 - Automatic Key Repeat

Stop wasting time - make full use of the repeat function just like the Z100. Simple plug-in installation on your keyboard assembly. Hold any key down for half a second and the key begins repeating. Combine this with our 4MHz mod and make WordStar fly! Provision for a defeat switch.

Kit.....$35
Assembled.....$45

ZCPR3

We are licensed by ECHELON, Inc. to distribute the Z-System for the Heath market. This includes the ZCPR3 system with supporting software utilities, the ZRDOS-plus BDOS replacement and applicable documentation. Comes already installed and includes a bootable disk ready to run.

ZCPR Gives you many features found in UNIX and MSDOS. The ZRDOS is not in the public domain and gives you automatic logon of disks. No more "BDOS ERROR ON B:" messages just because you forgot to do a ^ C after changing disks.

Available for Heath BIOS 2.2.03 and .04 and for CDR BIOS. Check for availability of other BIOS configurations. Call or write for a brochure on this excellent product.

ZCPR3 - specify format and hardware..... $98

WSPATCH — $20

Adds H19/H89 function key patches to versions 3.0 or 3.3 of WordStar. Key functions similar to the PIE editor. Includes provision for redefining the keys by the user. Also includes a printer driver for the Epson MX80/FX80 printers.

NEWWORD - the better word processor from NewStar Software

Faster than WordStar, same commands as WordStar, MailMerge feature is built-in, supports more printers, has an undelete function and excellent documentation. Includes WORD Plus spelling checker by Oasis - normally a $150 item by itself! We offer this package at discount and also include a function/cursor key patch for the H19/H89.

CP/M - specify format.....$115
MSDOS - specify computer.....$195

EMULATE

Allows the H89 to read/write to the following disk formats.

Actrix	Eagle II	Morrow MD	Superbrain Jr.
AMPRO	Epson QX-10	NCR DecMate 5	Televideo
Beehive Tpr	Fujitsu CP/M86	NEC PC-8001A	TRS80-1 CP/M
CDR	IBM CP/M86	Osborne 1	TRS80-3 CP/M
Cromemco	IMS 5000	Otrona	TRS80-4 CP/M
DEC VT180	Kaypro II	PMC MicroMate	Visual 1050
DEC Rainbow	Magnolia	Royal/Adler	Xerox 820
		Sanyo 1100	Zorba

Now includes 44 formats! Uses a modified BIOS which is included with the program. Allows reading of 40-track disks in an 80-track drive.

For H37 with Heath CP/M..... $59
For CDR BIOS 2.91.....$49
Check for Magnolia version.

DISKETTES

High quality private label diskettes at an economy price.

5" SSDD 10 hard sector.....$16/box of ten
5" DSDD.....$18/box of ten
8" SSDD.....$26/box of ten
8" DSDD.....$29/box of ten

ANALYTICAL PRODUCTS 209/564-3687

20663 Ave. 352 Woodlake, CA 93286

Specify Disk Format on Software

CALL OR WRITE FOR CATALOG PRICES SUBJECT TO CHANGE

Terms: Check or Money Order - VISA/MC - C.O.D.
Add $3 per order for shipping and handling
California residents add 6% tax

BUGGIN' HUG

"A Tight C/80 Environment" Correction

Dear Lori:

The appearance of my article, "A Tight C/80 Environment," in the January '86 issue of REMark pleased me to no end. But, there were two fairly significant typographical errors in it. It might do my ego a lot of good to blame somebody on the editorial staff, but I found both errors when I looked at my own copy of the manuscript.

Near the bottom of the text on page 17, the reader is directed to search for the string, "read_fil". If you try this with INGREP, it will quite correctly tell you, "No match found". While the C source code contains a function called "read_fil", it gets truncated to "read_fi" in the ASM file which is what you want to search for.

The other typo could very well go unnoticed. In the function printit() in Listing 1, there is an extra "/*," following the declaration, "int i;" as if to open a comment. There is no matching "*/". This is not a problem if you copy the source code without copying the comments; but if you copy it exactly as printed, the next two lines won't get compiled.

Knock off the extra letter on "read_fi", and the extra "/*" in printit() and everything should work just fine. My apologies for any head scratching this may have caused.

Sincerely,

Don Keller
1330 Eden Valley Road
Port Angeles, WA 98362

New Product Announcement

Dear HUG:

GEMINI users can now access data in IBM winchester disk subpartitions from Z-100 mode. GEMWIN has device drivers and an assignment utility that will allow from two to eight subpartitions to be read from or written to while in the Z-100 mode. This allows transferring data between the two modes of operations. Also, many programs which run in either mode only have to be kept in one partition saving disk space. GEMWIN requires an H/Z-100, 110 or 120 computer with a GEMINI emulator board and a Z-217 winchester controller and operates under MSDOS 2.1 or higher. GEMWIN sells for $44.95 + $2.00 shipping. Send check or money order only to J.J. Thompson, 281 Warren Avenue, Kenmore, NY 14217, (716) 873-0380 after 5 PM eastern standard time.

Would Like To Modify H-89 Without The Expense

Dear HUG:

I have owned an H-89A for the last four years or so and have found that excellent support is provided for it by the manufacturers who advertise in REMark and Sextant. Among other items, the Super 89 RAM board is an excellent product.

My reason for writing centers on the H-17 disk board and the limitation that every H-89A owner would have experienced with upgrading speed around that board. I am presently using one of the systems in which the speed of the RAM board is increased by using a module which turns off for disk accessing or I/O. The module works fine, but I find that programs that involve many accesses to the disk really slow the operation. I know that I could probably resolve this situation by obtaining either a CDR disk board or an H-37 disk board, but my limited use of the computer really wouldn't justify that expense. All the programs that I need are on the hard sectored format.

Pat Swayne in his article dealing with a 4 MHz modification for the H-89 in a REMark issue of approximately three years ago mentioned in his opening remarks that he knew of H-89 owners who had modified their system clock and had modified their monitor chip to change the H-17 speed parameters on that chip. I am wondering if any of the readers are among the persons who modified their machines in this manner, and if so, if they would be kind enough to share the necessary code modifications with me. Alternatively, would any readers familiar with the source code for the H-89A monitor chip point out to me which sections of the H-89A source code manual relates to the speed of accessing information from the H-17 disk drive.

Any help that readers could provide relating to their own experiences on this subject would be greatly appreciated. Thank you.

Philip Jones
222 Maclaren Street, Apt. 55
Ottawa, Ontario
CANADA K2P 0L8

Look Out For Paper Clips!

Dear HUG:

LOOK OUT FOR PAPER CLIPS! If they have been stored in a "ready access" paper clip holder, like the one on my desk, they MAY be magnetic. They may be magnetic enough to ruin a diskette! Most of these "ready access" paper clip holders use a magnetic ring at the top to hold the (steel) paper clips handily ready.

What will the gremlins think of next to glitch our data?

I have only had my Z-150 PC for a couple of months, but I am already addicted to it. Partly, thanks to "REMark" publishing many helpful articles and HUG making available the great low-cost software. Unfortunately, I have been so busy using my PC for business that I haven't had much time to enjoy it.

Sincerely,

Jon S. Wilson
32871 Via Del Amo
San Juan Capistrano, CA 92675

The Source Is The 5 Volt Bridge Rectifier Circuit

Dear Mr. Speidel,

I have just received my August 1985 issue of REMark, in which you ask for help in solving a video problem. About a year ago, the local Heathkit technicians helped me solve a very similar prob-

lem. On my H-89A, I had what looked very much like a horizontal "cloud" of lower intensity reverse video or reduced brightness normal video letters that would gradually roll upward on the screen. I also had problems with "garbage" characters replacing the valid characters I typed. If this is your problem, I may have a solution for you.

The source of this problem is in the 5 volt bridge rectifier circuit and could be caused by either of two problems. Look at the H-89A schematic, page three of three. Note that the yellow wires on the transformer secondary go to connector P101, pins 4 and 5. The circuit is continued through the circuit board foil strips to connector P103, pins 1 and 3. The first problem could be corrosion at the pin to circuit board interface. The direct approach is to simply bypass the circuit board by removing the yellow wires from P101 and soldering them directly to the bridge rectifier. Of course, the existing two orange wires must be removed from the bridge rectifier terminals. With the cover removed, all these wires are in the right rear corner (as viewed from above) of the power supply and are easily accessible. If this fails to solve your problem, then the bridge rectifier might be bad (second problem). When I bought a new BR-1 rectifier (Heath part number 57-67) about a year ago, it cost between $6.00 and $7.00. This solved my problem. Also, be sure to add more heat conductive paste to the bridge rectifier when it is installed.

If anyone is having a similar problem, I suggest that it be repaired as soon as possible. The 5 volt circuit powers many other components, and can cause secondary failure of these components.

Good luck, and I hope that this information will be of benefit to you.

Sincerely,

Edouard A. Piche
4127 Fox Hill Drive
Sterling Heights, MI 48077

ZP-150

Dear HUG:

Bill Gates' piece in the October issue is the second REMark promo for the ZP-150 in recent months. Gates writes less entertainingly than Jack Frank, but their point's the same: this is one great little machine.

Jack Frank convinced me back in June, and I bought one. It keeps its promises in every respect, but one.

Gates claims, in language similar to a line in the Heath catalog, that "the 32K bytes of memory give you enough room for the high-performance features that you've grown to expect on your office or home PC." (Mine is a Z-100, so I expect a lot).

It ain't so. MSWorks is a powerful, versatile — and memory hungry — system. The promised 32K shrink to about 23K as soon as you turn it on. Every part of the package, when used of course, nibbles a K here, a K there. This computer, like my mother, is prone to remember everything: a rare talent, sometimes welcome, sometimes not.

I'm using the ZP-150 at the moment on a project that requires sending text over a phone line to meet a weekly deadline. This involves composing in Word, saving a text file, and sending it with Telcom.

How much text can I send? With no other files in the ZP-150, i.e. with no way to use all those wonderful, versatile tools like a phonebook and a script to automate the file transfer, the "enough room" in the machine's pretended 32K limits the extent of my weekly scribble to 3 or 4 pages. The folk who read it no doubt rejoice; the discipline surely will improve my writing; but neither Frank, the catalog, nor Gates prepared me for life with so stringent a teacher of literary morality.

Two or three more 32K memory modules — at $190 a pop, says the catalog — will let the Iron Maiden unbend enough to begin to show her stuff. She's a charming, powerful, tough lady — no question. I just wonder whether Jack Frank or Bill Gates have actually asked her to dance in her 32K corset.

Sincerely yours,

Lewis Wilkins
3936 N. Delaware Street
Indianapolis, IN 46205

MBASIC Patch For HDOS

Dear HUG:

There have been several articles in REMark recently that involve the use of MBASIC on the H/Z89 machines. I do most of my work with HDOS rather than CP/M, so I use the latest available version of MBASIC on HDOS, which is version 4.82.

The recent programs, such as the Calc program in the March issue of REMark, are often written to be used only with MBASIC V5.21 under CP/M. I did not really understand why HDOS MBASIC was left out until I attempted to run one of these programs under HDOS and it did not work!

I tried using MBASIC version 4.7 for quite a while until I found that I needed the capability of using the INKEY$ function that was only available in V4.82. I was rather dismayed to find that this function, despite manual descriptions to the contrary, does not work in V4.82 under HDOS, so the programs which use the INKEY$ function will not run under HDOS and there is no easy way even to program around the deficiency.

This is where I became annoyed and decided to fix the problem, as I could find no reference to a patch in any of the magazines that I had. After some work, including a disassembly of the entire MBASIC, I came up with an answer. The following patch is that answer.

Follow the instructions for entering the MBASIC patch using the PATCH utility that was delivered with HDOS 2.0. This will enable the INKEY$ function in version 4.82 and will also renumber the version to 4.83, so you can tell if you are working with the patched version. I don't like having various unmarked versions lying about!

```
>PATCH
PATCH Issue #50.06.00

File Name?  MBASIC.ABS

Address?  134150
134150 = 315/377
134151 = 212/001
134152 = 134/000
134153 = 302/322
134154 = 164/
134155 = 134/
134156 = 312/332
```

```
134157 = 175/^D          Control-D
Address?  161150
161150 = 062/063
161151 = 015/^D
Address?  ^D
PATCH Issue #50.06.00

File Name? ^D
```

There, that's all there is to it. Try it out, but don't make the changes if the original data is not as I have specified it and lastly, don't patch your distribution copy. Many programs depend on the presence of the INKEY$ function, so I hope this added capability will be of some use to the HDOS people left out there.

Yours truly,

Ted Thompson
61 Sample Road
Ottawa, Ont.
Canada K1V 9T9

Typewriter Program Will Not Run When Trying To Use Backspace

Dear HUG:

The typewriter program, Version 6, page 42, will not run on my Z-150 when you try to use the backspace. I don't see why it should, since you are trying to get a character back from the printer by sending a backspace.

So, I wrote one that does work. It can not only backspace, but also insert and delete. Its name is Directyp, and it is enclosed.

Sincerely,

William R. Remington
210 South Road Lindamere
Wilmington, DE 19809

```
10 REM --- DIRECTYP ------------------------------------
20 GOSUB 7000
30 DEFSTR A, S: DEFINT I, J, M
40 DIM S(80): CLS
50 CLS: L = 1: I = 1     'L IS LINE   I IS CHAR COMING UP
60 LOCATE L, I
70 A = INKEY$: IF A = "" THEN 70
80 M = ASC(A)
90 IF M = 13 THEN GOSUB 1000            'PRINT A LINE
100 IF M = 91 OR M = 8 THEN GOSUB 3000 'BACKSPACE
110 IF M = 93 THEN GOSUB 4000    'REWRITE AFTER BACKSPACE
120 IF M = 9 THEN GOSUB 5000             'INSERT
130 IF M = 4 THEN GOSUB 6000             'DELETE
140 IF M = 27 THEN CLS: END
150 GOSUB 2000                           'ACCEPT A LETTER
1000 REM --- PRINT A LINE ----------------------------
1010 FOR J = 1 TO I - 1. LPRINT S(J);: S(J) = "":
     NEXT J: LPRINT
1020 L = L + 1  I = 1: RETURN 60
2000 REM --- ACCEPT A CHARACTER ----------------------
2010 PRINT A;
2020 S(I) = A
2030 I = I + 1: IF I > IMAX THEN IMAX = I  'RESETS THE
                                            RIGHT END OF
                                            LINE
2040 RETURN 70
3000 REM --- SUB BACKSPACES -------------------------
3010 IF I = 1 THEN BEEP· BEEP: GOTO 3060
3020 I = I - 1
3030 LOCATE L, I
3040 COLOR 0, 7: PRINT S(I),. COLOR 7, 0
3050 LOCATE L, I
3060 RETURN 70
4000 REM --- RECOVER TEXT AFTER BACKSPACE ------------
4010 IF I >= IMAX THEN BEEP  BEEP: GOTO 4050
4020 LOCATE L, I
4030 PRINT S(I);
4040 I = I + 1
4050 RETURN 70
5000 REM --- SUB INSERTS -----------------------------
5010 LOCATE ,,,3,6
5020 A = INKEY$: IF A = "" THEN 5020
5030 M = ASC(A)
5040 IF M = 14 THEN 5190
5050 IF M = 13 OR M = 93 OR M = 91 OR M = 8 OR M = 9
     THEN 5020
5060 IF M = 27 THEN END
5070 FOR J = IMAX TO I - 1 STEP -1
5080    S(J + 1) = S(J)
5090 NEXT J
5100 S(I) = A: PRINT S(I);
5110 IMAX = IMAX +1: I = I + 1
5120 COLOR 0, 7
5130 FOR J = I TO IMAX - 1
5140    PRINT S(J);
5150 NEXT J
5160 COLOR 7, 0
5170 LOCATE L, I - 1
5180 GOTO 5020
5190 LOCATE ,,,5,6: RETURN 70
6000 REM --- SUB DELETES ONE CHARACTER ---------------
6010 COLOR 0, 7
6020 FOR J = I TO IMAX - 2
6030    S(J) = S(J + 1)
6040    PRINT S(J);
6050 NEXT J
6060 S(IMAX - 1) = "": IMAX = IMAX - 1
6070 COLOR 7, 0
6080 PRINT " "
6090 RETURN 60
7000 REM --- SUB EXPLAINS ----------------------------
7010 CLS: LOCATE 1, 30: PRINT "This is DIRECTYP"
7020 PRINT :PRINT "The computer runs like a typewriter."
7030 PRINT :PRINT "Type in what you want. Backspace with
     [ or backspace. Restore with J."
7040 PRINT :PRINT "To insert a character, put the cursor
     where you want to insert, and tap ctrl-i."
7050 PRINT "To stop inserting, tap ctrl-n."
7060 PRINT :PRINT "To delete a character, put the cursor
     where you want to delete and tap ctrl-d."
7070 PRINT :PRINT "To print a line, tap return."
7080 PRINT :PRINT "To end, tap esc."
7090 PRINT :PRINT "To proceed, tap space bar."
7100 A$ = INKEY$: IF A$ = "" THEN 7100
7110 RETURN
```

Hoping Others Can Avoid Misfortune

Dear HUG:

I'm writing to let you know about some serious problems I've had with an add-on board for my Z-150, hoping that other users can avoid my misfortunes by being forewarned.

In June 1985, I ordered a Tecmar Captain Board from Software Wizardry. The board arrived promptly, but no amount of fiddling could get it to work properly with my machine. Getting no useful advice from the people who sold me the board, I finally talked to engineers from Zenith and Tecmar, and discovered something that no one at Software Wizardry knew about: there are several different versions of the Captain board, only one of which will work with the Z-150. In its original version, the Captain assumes that there is 256K memory in the machine, which is fine for IBMs. The problem is that the Z-150 has 320K on the motherboard, causing memory address confusion.

There is apparently a modified version of the Captain board with a changed PAL chip that is supposed to overcome the Z-150

incompatibility. It is Tecmar's no. 811050, and is presumably what one would get directly from Zenith. I say "apparently" because I am still waiting to receive the correct board from Software Wizardry, months after placing the order and wasting a lot of time because of their mistake. From now on, I will buy add-ons directly from Zenith.

Yours,

James Curtsinger
University of Minnesota
Dept. of Genetics and Cell Biology
250 Biological Sciences Center
1445 Gortner Avenue
St. Paul, MN 55108-1095

Qualification Of Kenneth Mortimer's Note

Dear HUG:

Kenneth Mortimer's note in the September '85 issue about line numbering in FORTRAN needs some qualification.

On all main frames I've ever used, columns 73 to 80 of FORTRAN statements were reserved for line numbers if one wished to use them. In fact, during the last few years my experience has been that main frame FORTRAN compilers automatically insert the information: they put the first three letters of the program name into columns 73 to 75, and line numbers into the remaining five columns often in multiples of ten.

Things are different with the personal computer FORTRAN compilers. On some of them (for example, the very excellent SSS FORTRAN-4 compiler under CP/M) you cannot use columns 73 to 80 for this purpose. The problem seems to be that because these compilers are capable of reading squeezed FORTRAN statements which pay no attention to continuing from line to line, these compilers would try to read the information in columns 73 to 80 as if it were part of the FORTRAN.

I have written a special program which will handle FORTRAN programs transferred from main frames to make them suitable for PCs. The program strips the line numbers AND ALL THE BLANKS from the end of each line of FORTRAN code.

Sincerely,

H. Harry Spencer
P.O. Box 725
New Brunswick, NJ 08903

Purchased EZPLOT And Own A Gemini Printer?

Dear HUG:

Please print the following for HUGgies that purchased EZPLOT and own Gemini printers.

Gemini printers are nearly command compatible with Epson printers. Unfortunately, in the recently released HUG program, EZPLOT, one of the commands used to perform graphics screen-dump to Epson printers is one that is not compatible with Gemini printers. Specifically, the command for setting line spacing differs slightly between the two printers. For graphics, a line spacing of 1/9" is desired. The necessary command string in HEX for Epson is 1B 33 18. The Gemini command string in HEX is 1B 33 10.

The following modification to EZPLOT (HUG P/N 885-3023-37 and HUG P/N 885-6003-37) will provide graphics dump support for the Gemini printers at the expense of losing Epson support.

1. Use DEBUG.COM to implement the patch.
2. Use normal CAUTION in performing this task. Always assume the worst will happen and provide yourself with some protection against the inevitable. In other words, make a working copy of EZPLOT.COM and perform the following operations on that copy.
3. Depending on your version of EZPLOT, perform the following:

For the Z-DOS version. (H/Z-110 and 120 computers only):

```
A>DEBUG EZPLOT.COM        ;Load EZPLOT.COM into DEBUG
-E57B7                    ,Enter new data at address 57B7H
xxxx:57B7    18.10        ;Change 18H to 10H. That's all!
-W                        ;Write edited version back to disk
-Q                        ;Quit and return to DOS
```

For the MS-DOS version. (H/Z-150 and 160 computers only):

```
A>DEBUG EZPLOT.COM        ;Load EZPLOT.COM into DEBUG
-E601A                    ;Enter new data at address 601AH
xxxx:601A    18.10        ;Change 18H to 10H  That's all!
-W                        ;Write edited version back to disk
-Q                        ;Quit and return to DOS

************************CAUTION************************
*  IF YOU DO NOT FIND AN 18H AT THE PRESCRIBED ADDRESS *
*                    MAKE NO CHANGE                    *
************************CAUTION************************
```

4. Test out your patched version. Now when you wish to get a hardcopy of your plot use the EPSON option on the DUMP menu. It should work fine. Enjoy.

Thank you.

Sincerely,

Tom L. Riggs, Jr.
215 S. Brookwood Drive
Auburn, AL 36830

Gilchrist's Cinp And Cout

Dear HUG:

Allen Gilchrist's cinp and cout routines in the July "Favorite Subroutines" column brought to mind a couple of other solutions to the same problem. Self-modifying code is not ROM-able, and usually risky. And the Z-80 chip has a powerful instruction that needs no modifying.

Whitesmith's C compiler, in order to make its code ROM-able, builds the 8080 out (port) instruction on the stack, calls it, and on return restores the stack pointer. This is cleaner, but takes a lot of doing. When I first saw it, I was appalled and did the job with the Z-80 "out (c),r" instruction, where r is any register. The resulting code is simplicity itself.

```
cinp(port)
{
#asm
    POP  D      ret adr
    POP  B      port in c
    DB   355Q   prefix
    DB   150Q   in L,(c)
    PUSH B      restore stack
    PUSH D      ret adr
```

Continued on Page 83

A Winchester For The '89

Part Two

Peter Ruber
P.O. Box 502
Oakdale, NY 11769

The Magnolia Connection

Magnolia Microsystems of Seattle, WA is probably responsible for more sophisticated enhancements for the H-89 computer than any other independent supplier — and that includes Heath/Zenith. In fact, Magnolia (which was founded in 1976) actually predates both Heath Company and Zenith Data Systems.

They began as a retail operation selling CP/M computers, such as the Altos, and were the first operation of this kind in the Seattle area. When Heath launched the H-89 in 1979 as a small, relatively low-cost computer that the market needed, Magnolia initiated an evaluation of the unit in relation to their customers' needs. The H-89 lacked a standard Operating System (it came with HDOS), so Magnolia implemented ORG-0 CP/M and introduced it at the West Coast Computer Faire in 1980.

Seeing that the '89 had a far-reaching sales potential beyond the hobbyist market, Heath/Zenith rushed to complete their own version of CP/M in 1981. Unfortunately, the accompanying documentation package did not do credit to their usually high standards; and it took them another two years to provide the updated version 2.2.04 (which provided support for the Z-89-11 Multi I/O card) and a manual that was a model of clarity and detail.

Magnolia was also responsible for yet another first: the implementation of a Winchester hard disk system for the '89 as early as 1980 — a full two years before Heath/Zenith offered the Z-67 subsystem — with the 77313 Corvus host adaptor. The Corvus interface (actually designed by IMI, whose 8" drives and controllers Corvus used) predates the Shugart SASI-bus. It was contemporaneous with the 8" SA1000 interface from Shugart, but at a different functional level — IMI put the controller inside the drive itself. Shugart competed by developing a controller using their own (SASI) bus (which went on to become an industry standard and, eventually, an IEEE (SCSI) standard). Along the way, IMI also designed 5" drives with their own (non-ST506) interface, with an "IMI-bus" controller which Corvus also used.

The Magnolia Corvus Winchester interface appeared shortly after their release of CP/M. All at once, the H-89 had gone from a computer system primarily designed for the hobbyist, to a full-blown business machine capable of holding its own among its peers.

In 1981, when Heath/Zenith was taking "forever" in releasing their Z-89-47 interface and Z-47 8" drive subsystem, Magnolia updated their Corvus hard disk interface by adding the Remex intelligent floppy I/O to their 77313 board, which now came to be called the 77314, and beat Heath to the market by several months.

This was quickly followed by their 77315 (CAMEO) and 77317 (XCOMP) interfaces, which are still available from stock, but no longer actively promoted. The XCOMP controller was designed for both the SA1000 and ST506 drives. The CAMEO controller supported one of the early Winchester monster units — a 14" 5+5 MB Cartridge Subsystem. The one truly outstanding feature of these early Magnolia hard disk interface cards was the addition of their own proprietary bus expansion technique.

As the abilities of the H-89 grew, several shortcomings in the design became more obvious. Aside from only having a 90 watt power supply, it did not support direct memory access, and it only had three expansion ports available. Due to internal space limitations, expansion cards were greatly restricted in size, which in turn limited the number of I/O ports the computer could have resident at any given time.

By enhancing the BIOS of their CP/M implementation, writing new software driver utilities, as well as preparing a variety of BOOT ROMS, Magnolia was able to open up a whole new world of I/O expansion for the '89. This wasn't done at the expense of the I/O ports Heath/Zenith had built into the computer. They remained intact, especially the disk I/O port slots at P504 and P506. Instead, Magnolia's proprietary expansion bus confined itself to the center slot (P505), through which they implemented their hard disk interfaces, as well as the 3-Serial I/O ports that otherwise would have been lost with the removal of the H-88-3

Serial card. This explains why Magnolia expansion cards always resemble an aerial view of an overcrowded parking lot. Brad Gjerding and his designers had a knack for stuffing four functions into a space ideally suited for only one.

I suppose its reasonable to ask — if a small firm with a handful of employees and relatively modest resources could do so much to enhance the usefulness of the '89, why didn't Heath/Zenith with its overwhelming engineering staff and financial clout do more than it did to expand the '89? At best, their efforts appeared more to patronize a loyal band of customers than a consummate desire to sophisticate their product. As soon as the '89 hit the streets, they were hard at work on the next generation of computers, incorporating the latest advances in board design and IC technology, and assembly-line automation that would result in a faster, more powerful computer at a lower cost. The big profits were in selling systems — not add-on cards.

When Heath/Zenith introduced their Z-89-67 SASI interface card and the Z-67 Winchester subsystem in 1982, Magnolia had their own SASI interface card in the final stages of development — the 77320. Since the SASI standard is a hardware specification, the Z-89-67 and the 77320 cards are compatible with each other as is their respective software. That is, Z-67 software will work with the 77320.

When I asked Brad Gjerding if he could recall any limitations the 77320 board had, he replied: "We've come to regret not including an option to use one of our expanded I/O addresses, so people could have the 77320 board in their computer together with both Z-17 and Z-89-37 controllers. But, that's 20-20 hindsight. However, we did include the standard Heath serial ports at the bottom of the card. Things got a little crowded and we had to tack one of the three serial port connectors on the back of the card."

When you install any Magnolia Winchester host adaptor card in the '89, you can only have one floppy disk controller resident at the same time. In practical terms, however, once you've upgraded to the Z-37 disk controller, there really isn't much point in tolerating the limited storage abilities of the hard sector controller. And, with a Winchester, you seldom use even your Z-37 controller, except to transfer programs to the Winchester and to perform periodic back-ups of your hard disk files. In daily use over the last four months, I can't recall having turned on my quad-density drives more than a dozen times. You're spoiled quite easily when you install a Winchester on the '89. You can BOOT from any of the partitions you have set up in an average of 12 seconds, as opposed to nearly 45 seconds from a floppy drive.

Five years have elapsed since Magnolia implemented CP/M and the first Winchester hard disk system for the '89. During this intervening period, Heath's hard disk subsystem was a viable product for only two years, because the emergence of their 16-bit system relegated the 8-bit '89 to secondary status. And, as this article is being written (September 1985), the '89 has been discontinued after a six-year reign. This has placed Magnolia Microsystems in the enviable position of being the only manufacturer of hard disk host adaptor cards. Unquestionably, their biggest seller was the Corvus 77314 interface, because the Constellation multiplexer allowed several computers to share one rather expensive hard-disk drive — a system they supplied to many companies around the country.

Magnolia's 77320 SASI interface competed quite well with Heath's Z-89-67 host adaptor because their implementation of

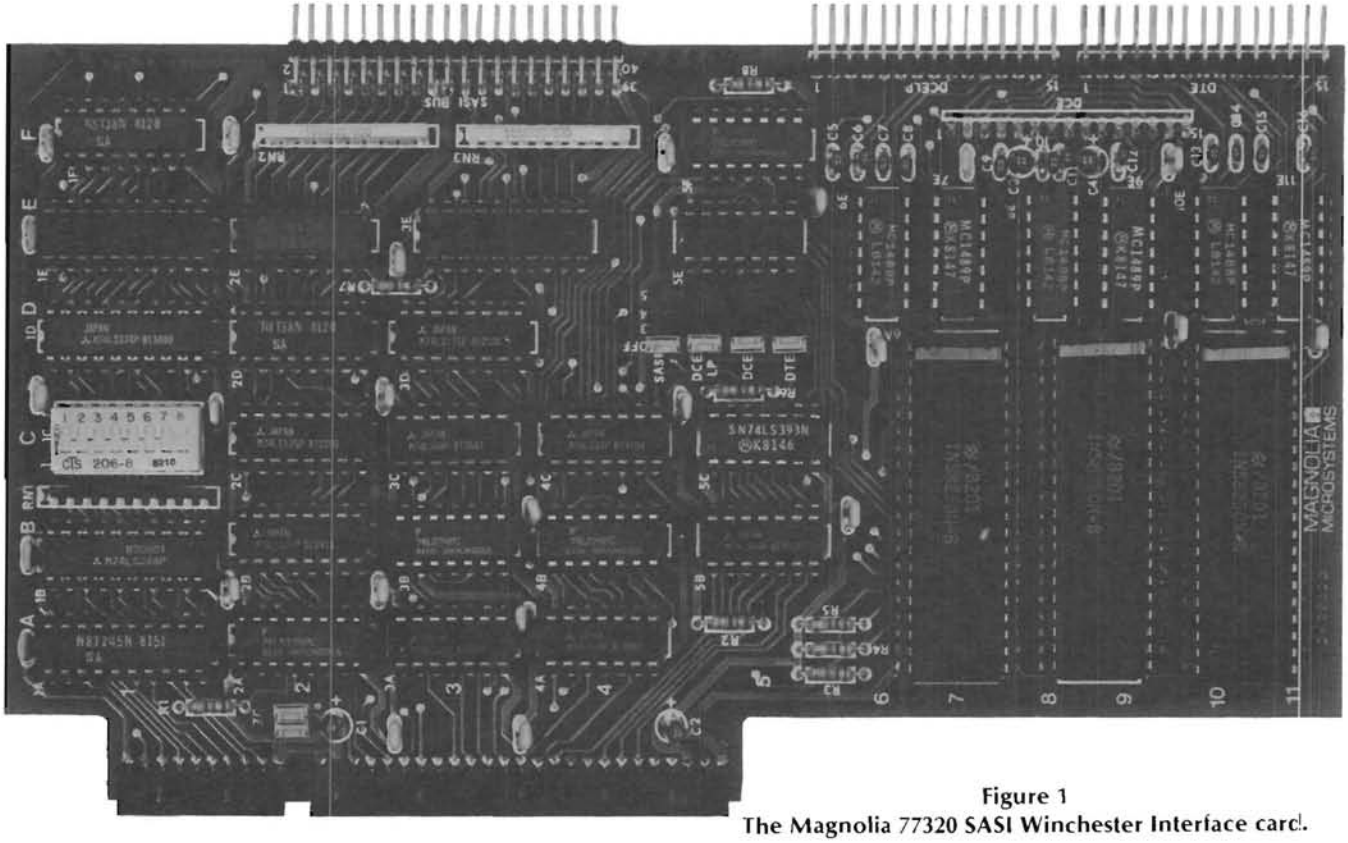


Figure 1
The Magnolia 77320 SASI Winchester Interface card.

CP/M allowed greater flexibility in hard disk drive selection, whereas Heath's would only support the mammoth Memorex 101 8-inch Winchester I discussed in Part One of this series. Also, Heath's Winchester BIOS was sold as a separate software package. It would have been nice if they had incorporated the Winchester BIOS into CP/M 2.2.04 rather than consigning it to the old products heap. So, we are left with the 77320 SASI interface as the universal communicator between the '89 and a hard disk drive.

I had hoped to obtain schematics and other technical data for this article, but I was informed this information was not available. "We support our products at the 'functional interface' (black box-to-black box) level, complete with system software, and customers don't need to worry about technical details, such as actual cable pinouts. If you have a Corvus drive, it works with the '314 and our software. If you don't, it won't . . ."

In correspondence around the country, I have learned that Magnolia has come to rely more heavily on their dealers to provide the necessary technical support required by the user, because these dealers quite often are their own hard disk system integrators and Magnolia isn't in a position to know what drives and controller combinations are being offered in a subsystem. If you purchase a complete system from Magnolia, it behooves them to provide you with the help you need to get it operational. But if you merely purchase the SASI host adaptor and the CP/M software (which comes free with the SASI), and scout around for the rest of your system at the lowest possible price, Magnolia can't afford to spend a lot of time holding your hand.

Magnolia's SASI host adaptor and CP/M is optimized for the Xebex S1410 and the newer low-power S1410a hard disk controller card. I am told it will work with other hard disk controllers which I hope to test out in one of the forthcoming articles in this series. On the hard disk drive level, any ST506 standard drive is usable. But this doesn't mean that you should grab any surplus bargain you see advertised.

A primary rule you should remember is to purchase only a drive for which documentation is available. In order to use Magnolia CP/M on a Winchester subsystem, you must create a BIOS and LINK it into your system volume which contains important information about the drive you're going to use — specifically, the number of read/write heads, number of cylinders, tracks per cylinder, etc. Without this information, you cannot partition the hard disk drive and then Sysgen it as a bootable device.

The one negative feature of the Magnolia CP/M Winchester utilities is that you are only allowed two partions on a Corvus system. I think that when this standard was established, most Corvus drives available were in the 5-10 MB range. The 77320 SASI interface will allow up to four partitions on a drive, which offers the user somewhat more flexibility in the number and types of programs he can have resident on a single drive, especially when you are using a drive with 20 MB or more of storage capability.

On the positive side, Magnolia's 77320 SASI utilities will support eight hard disk controllers, each of which (if you are using the Xebec) can handle two Winchester drives. Simple mathematical extensions will deduce that you could, conceivably, run sixteen Winchester drives off one '89, if you're willing to spend a small fortune in the process. Most 16-bit computers aren't capable of a hard disk network of this magnitude. I wonder if Magnolia realized the ultimate power they had made available to the H-89 computerist when they developed their SASI utilities. No doubt they had a multi-user office environment in mind, because they were the first and only firm to develop a controller network that catapulted the '89 into the business world.

Getting Started Under Magnolia CP/M

My first encounter with Magnolia CP/M was caution mixed with fear. I had been told that it was the "only way to go" and "once you've used Magnolia's CP/M, no other version on the market would ever be satisfactory." As I'm usually suspect of broad statements about a product's superiority, I wasn't quite certain what to expect when my software package arrived.

The documentation is handsomely prepared and stuffed into a fat, gray binder. Depending on the type of hard disk system you are planning to install, several supporting booklets pertaining to the hardware installation, the Monitor ROM and the software utilities you will be working with are also supplied. You will also receive a functional replacement for the standard Heath/Zenith MTR90 4k Boot ROM, as well as new I/O Port Decoder ROMs replacing part numbers 444-61 and 444-83. There are "a", "b" and "c" versions of all these ICs, covering all of Magnolia's Winchester options.

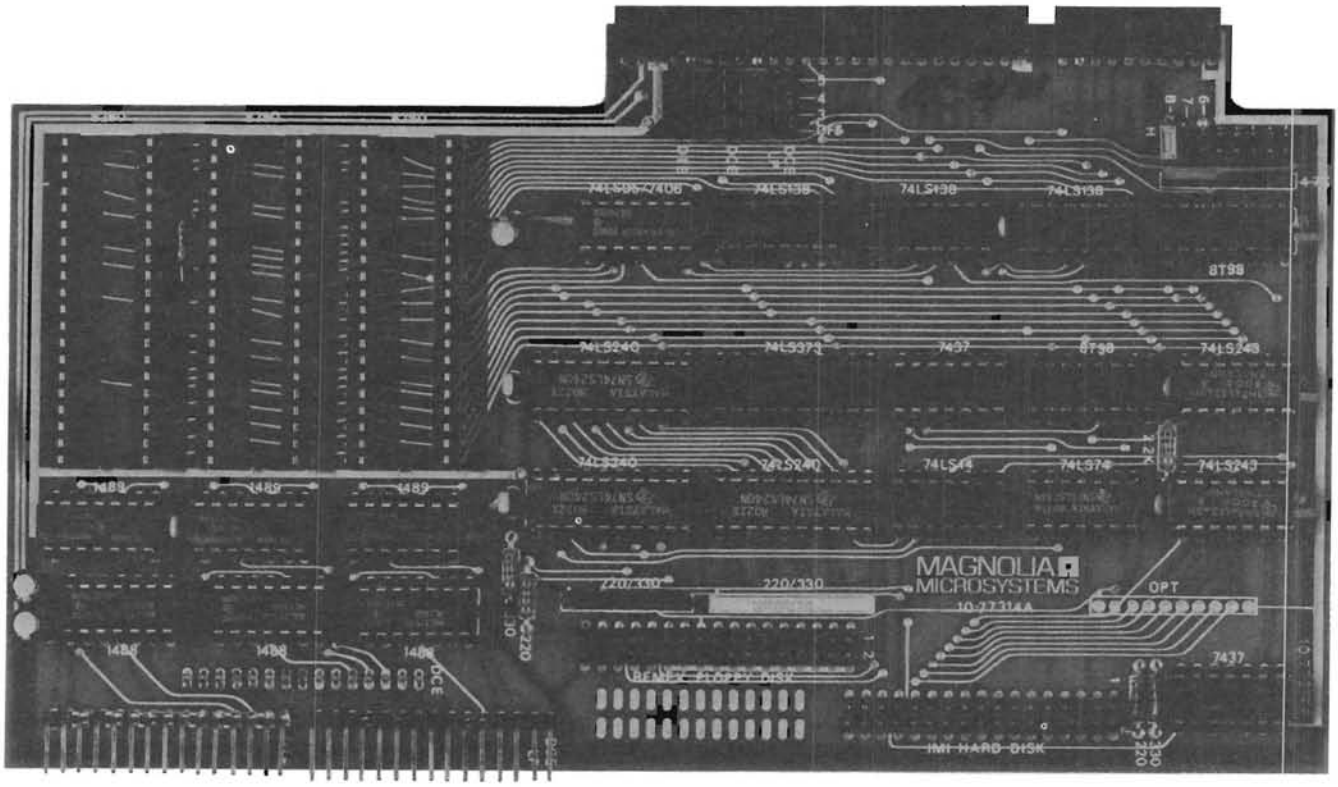
None of these will affect the operation of the standard Heath/Zenith version of CP/M or even HDOS. You will not be able to use your Winchester with H/Z Operating Systems, but you can certainly Boot these from your floppy drives — with a minor change. That is, the symbolic representations of the SY: and DK: floppy disk devices under HDOS and the alpha designations of CP/M have been converted to a numerical format.

Depending on how you have set up the switches on S501 on the '89s CPU board to designate the primary Boot drives, the procedure with Magnolia CP/M requires that you specify the assigned number of the drive you wish to Boot from. According to Magnolia's very elaborate scheme, device numbers have been assigned to cover the Heath/Zenith Z-37 soft-sector controller and H-17 hard-sector controller, the Z-47 and Z-67 subsystems, Magnolia's own 5"/8" double-density controller, their 128k RAM card, and their Cameo, Xcomp, Corvus and SASI interfaces.

We shall ignore the basic setting up procedures for a CP/M System Volume disk, because of the similarities to Heath/Zenith's CP/M. But I shall mention that you will now be able to add a fourth drive to your Z-37 controller and increase the amount of physical data you can store on your 48 or 96 tpi double-density disks because of the way the track/sector format has been allocated. Also, the FORMAT procedure offers a screen display as each track is formatted, and these are automatically verified so that you can instantly spot a bad disk. Most of the set-up procedures are handled by screen menus, and you use your cursor and special function keys to move around the menu and to configure the associated hardware you have on-line.

I think it is appropriate to mention to those '89 owners who are not familiar with both the Heath/Zenith and Magnolia CP/M version that data and program files from either version cannot be transferred to the other because MMS and H/Z handle the logical-physical head positioning of double-sided disks in different ways. In simplistic terms, Magnolia CP/M writes to one side of a disk first, then the other side, with side two offset by about 4 tracks. Track 0 on side one begins at the outside edge of the disk and writes to the center hub. Side two begins to write (and read) from the center hub to the outside edge. This was done for compatibility between Magnolia's 5"/8" double-density controller and the H-17 single-density controller. However, the Z-37 double-density controller follows an alternating side method of

Figure 2
The Magnolia 77314 Corvus (IMI) Winchester Interface card.



read/write: First track 0 on side one, then track 0 on side two, and so forth.

Those who are contemplating switching from Heath/Zenith CP/M to Magnolia and wish to place their working files to a Winchester drive, should first transfer their files to an MMS CP/M formatted floppy disk via a conversion utility program before loading them onto a hard disk. One such program is called EMULATE, which is distributed by Analytical Products, 20663 Ave. 352, Wood Lake, CA 93286. This utility will read and write to more than 4 different CP/M formats and is worth investigating.

Now, before I wander too far afield from the intended scope of this article, we will now go through the procedure for setting up a hard disk using the Magnolia CP/M, 77320 SASI interface, and the Xebec hard disk controller. The hard disk data listed below is for both the Microscience HH-612 and Shugart SA-712, which are new half-height, low-power 10 MB Winchester drives. Part of the set-up procedure has been condensed from a booklet published by Floppy Disk Services (39 Everett Dr., Bldg. D, Lawrenceville, NJ 08646) and entitled INSTRUCTION MANUAL FOR INTERNAL AND EXTERNAL HARD DISK SYSTEMS FOR THE H/Z-89. It is one of the best idiot-proof manuals I have ever read, and it will be discussed in Part Four of this series when I will examine their internal hard disk system for the '89.

1. Turn on the '89. It should beep twice and the hard disk will begin to wind up. The LED on the hard disk may or may not flicker or be lit for the first 30 seconds. The Boot Syntax is Boot 0 for H/Z-17, Boot 46 for Z-37 or Boot 33 for MMS double-density.

2. Format and Sysgen a blank disk in drive B:. Copy all the utilities from your Master MMS CP/M disk. Reset the '89, insert the System Volume you have just made and PIP the files from the SASI diskette to it.

3. Reboot your disk and run DEFSASI3.COM. After the program has loaded, the cursor will position itself at the first line titled SUBSYSTEM DATA. Hit the Return key and enter the following data:

```
Controller number      Ø
Controller mfg         Xebec   (enter #1 to get this)
Model                  S1410   (enter #1)
Version                104683  (enter #2)
Logical unit number    Ø       (enter #Ø)
Drive mfg              MSCI    (enter #1 and type in MSCI)
                               (or Shugart if applicable)
Model                  HH-612  (enter #1 and type in HH-612)
                               (or SA-712 if applicable)
Physical sector size   256     (enter 256
```

Now hit the BLUE key on the '89 to save this info and return to the main menu.

4. Move the cursor to the DRIVE CHARACTERISTICS section using the up/down arrows on the keypad and hit the Return key. Using the guide below, enter the data as given:

```
Logical unit number      Ø
Type of media            F
Number of cylinders      306
Number of heads          4
Sectors per track        32
Control byte             14 (use 17 for SA-712)
Interleave factor        32
Expected format time     30
Exp. disk test time      10

Reduced write cyl        Ø
Write precomp cyl        Ø
ECC data burst length    11
```

Hit the BLUE key to save the data and return to the main menu.

5. Position the cursor to the UPDATE DEFSASI3.DAT and hit return to save the data to the disk.

6. Position cursor to DRIVE INITIALIZATION and hit return. The cursor will position itself in the desired starting location. Hit return for each item. Some of the functions are instant. Others will take a few minutes. All tests must finish with NO ERRORS. When complete, hit the BLUE key to exit to the main menu.

7. Move cursor to the WRITE M320F210.HEX file and hit return to create this file on the disk.

8. Then move cursor to EXIT TO CP/M and hit return.

9. Now we must tell the system to recognize that the hard disk is present. This is accomplished by linking the new hex module as follows:

```
A>LINK M320F210 MOVCPM <ret>
```

10. When the link is complete, immediately run MOVCPM. Then run SYSGEN. Since the new system is already in memory, it isn't necessary to "get" the system, only put it. In other words "RETURN" for source and "A" for destination. And since you have just re-sysgened drive A, you must SHIFT-RESET and reboot.

11. Now run the program DRIVES. This program may take several seconds to complete. Look for 2 devices, 50 and 51 in brackets. These are the numbers assigned to your new hard disk system. Make a note of the "letters" (not the numbers). They will be used in the next few steps. These numbers and letters will vary depending on the options in your particular system. If all goes well, the system reboots, the new hard disk will appear as the letter I: (eye). REMEMBER! The letter may be different on your system. In our examples, we will use the letters "I" and "J".

12. Type the following: SCAN I:<ret>. Then, follow the directions on the screen. You will only have to hit return once for this operation. It may take from 5 to 15 minutes to perform this operation.

13. When this function is complete, and no errors have been encountered, you must now SCAN J:<ret>. This will not take as long as I: since this partition is smaller than the other. Remember that the default partition is 8.1 MB for the first partition, and 1.5 MB for the second as CP/M can only address 8.1 MB maximum under this system. If you wish to partition the hard disk in a different way, refer to the Magnolia Manuals. If everything has gone well, both partitions should not show any error. Now you may PIP all files (*.*) to the hard disk by typing PIP I:=A:*.* (during this operation any bad sectors found will be "mapped" out).

14. In order to boot directly from the hard disk, we must set up the system to do so. Since you have just copied all the files from your master to the hard disk, log onto the first partition, I:<ret>, per the examples in the two previous sections. Remember that your assignment letters may vary.

15. Run the SETUP.COM program and select the Logical/Physical Drive Assignments. Hit return.

16. Under MMS CP/M, the first 5.25" drive (using the MMS controller) is designated as 33, then 34, 35, 36. If this is the case, the screen will display number 33 in parentheses. If you're using the Z-37 controller, the numbers would be 46, 47, 48, 49. In order to make the hard disk the first and second logical drives, enter 50 when the A: designator is on the screen. Then move the cursor down one and enter 51. Then down again and enter number 33, 34, etc., or 46, 47, etc., for the controller you are using. If you see any error messages at this time, simply ignore them. If your system is different from our examples, consult the Magnolia manual for assistance.

17. Now press the "f3" key. This will clear any error codes. Then hit the BLUE key to update and exit to CP/M. To link the new boot module onto the hard disk, proceed as follows:

    While logged on I:, type:

```
I >LINK B320 MOVCOM <ret>
```

    When the link is completed, run MOVCPM. Then Sysgen and put the system from memory to drive I: Remember that the system is in memory, so just hit "return" for the question "Get system".

18. Your new hard disk is now ready to boot. Hit SHIFT/RESET and enter "B" at the MMS: prompt, followed by the letter "E". This will echo as two letters — "EE". Hit return and your new hard disk should boot up.

A final note of importance: When you first install your MMS 77320 SASI controller and the appropriate ROMs that come with this interface package, you must alter the dip switches on S501 on the CPU board according to the documentation.

And, as mentioned earlier, obtaining the manufacturer's documentation for the hard disk drive you plan to purchase is of the utmost importance. The set-up parameters are vital, if you are to link them to your system volume. Without this information you will be unable to communicate with the hard disk.

If possible, obtain a copy of the Xebec S1410 Owner's Manual. Many system integrators don't always supply this when you order the controller either separately or as part of a complete subsystem. Some, I've learned, don't have them to sell. They can be purchased from Xebec, P.O. Box 512, 432 Lakeside Drive, Sunnyvale, CA 94086. It provides a valuable source of information pertaining to pin-outs, error messages, SASI interface schematics, programming data, parameter information on some two dozen different ST506 compatible drives, and much more.

In one of the future articles in this series, we will attempt to explain the significance of the hard disk parameters that your software looks for during set-up.

For further information on the many Magnolia CP/M software products and fully operational Winchester subsystems MMS has to offer, write to:

MAGNOLIA MICROSYSTEMS, INC.
4039 - 21st Avenue West
Seattle, WA 98199

Our next installment will look at the Quikstor Winchester from Quikdata and the new standard Heath/Zenith CP/M and HDOS hard disk software they now have available. I welcome any comments and suggestions on this series. Please enclose a stamped reply envelope if I can assist you with any specific or technical information relating to your H-89 Winchester. *

More Megabytes for the Buck!

Move up to Winchester Performance for Only $895

Here's the ultimate solution to your H/Z89 or H8 data storage problems, a Winchester Hard Disk. No other single upgrade for your system can deliver such a performance leap.

- Super Fast Disk Drive Access Speeds
- Have All of Your Data On-Line Simultaneously
- Can Eliminate Hundreds of Floppy Disks
- Allows Better Organization of Your Data
- Better Data Integrity than Floppy Disk Storage

Includes Z37 Compatible Floppy Disk Controller!

System #1 - 10 Megabytes with Internal Mounting $895

- The Very Latest 3.5 Inch Low Power Technology
- Mounts in the Front of Your H/Z89 Cabinet
- Ideal for H/Z89's that are Frequently Transported
- Additional Drives can be Attached with an External Cabinet
- This Mounting cannot be Combined with our Interactive Graphics Controller Product

System #2 - 10 Megabytes with External Cabinet $995

- Super Fast 5.25 Inch Half Height Drive
- Elegantly Mounted in an External Color-Matched Cabinet
- Cabinet Features Room for a Second Drive
- Compatible with the Interactive Graphics Controller

System #3 - 20 Megabytes with External Cabinet $1095

- Super Fast 5.25 Inch Half Height Drive
- Elegantly Mounted in an External Color-Matched Cabinet
- Cabinet Features Room for a Second Drive
- Compatible with the Interactive Graphics Controller

All Three Systems Feature:

- Up to 16 Data Partitions for HDOS, CP/M, or Both
- Cold Boots HDOS and Heath CP/M (Requires MTR-90 ROM)
- Expandable to 3 Hard Disk Drives and 4 Floppy Drives
- Full ECC Error Correction for Very High Data Integrity
- 2 Centronics Parallel Printer Ports
- Independent Power Supply
- Installs Professionally and Easily
- Uses Only One Bus Slot (H/Z89 Left Side)
- Six Month Limited Warranty on All Components

These systems include a Z37 compatible floppy disk drive controller and software for no extra charge. Just plug in your existing drives, or order a drive to mount in the extra space available in our external cabinet and you'll have double density performance for your floppy drives!

System Expansion Options

Half Height 40 Track Floppy Drive	$195
Half Height 80 Track Floppy Drive	$225
Additional 10 Megabyte Hard Disk Drive	$445
Additional 20 Megabyte Hard Disk Drive	$495
Additional External Cabinet with Power Supply	$195

SigmaSoft AND SYSTEMS

"Support is the most important feature."

17000 Dallas Parkway, Suite #207, Dallas, TX 75248, (214) 380-6187

Our engineers would love to answer all of your questions about our products. Please give us a call to check current prices. Please include $10 for UPS shipping and specify H/Z 89 or H8. Texas residents please add 6% sales tax.

MasterCard VISA

ZPC Update #4

Pat Swayne
HUG Software Engineer

This is the fourth in a series of support articles for ZPC, a program that allows H/Z-100 (dual processor) computers to run some IBM PC and/or H/Z-100 PC software. ZPC is available from HUG as part no. 885-3030-37, and is described in the new HUG Software Catalog Update. A companion ZPC Support Disk (described in the January REMark) is available as part no. 885-3034-37.

In this edition of ZPC Update, I will discuss a fix for the LPRINT command in compiled BASIC programs, and operation of the following under ZPC: Turbo Pascal 3.01, WordPerfect 4.1, Compiled QuickBASIC programs, PrintMaster, PC-Write 2.4, PC-File, and a "civilian" version of MultiMate 3.3.

LPRINT In Compiled BASIC Programs

If a program is compiled with the IBM BASIC compiler, the Zenith GW-BASIC compiler, or the MicroSoft QuickBASIC compiler, LPRINT statements used in the compiled program will not work under ZPC. That is because the compiled program attempts to open the printer as a device using "LPT1" as a file handle. The PC version of MS-DOS recognizes both "LPT1" and "PRN" as valid handles for the printer, but the Z-100 version only recognizes "PRN", so the programs do not run under ZPC (a device error message is generated). One solution to this problem is to insert a small program into memory that intercepts calls to the DOS to open file handles. The program checks for the handle "LPT1", and changes it to "PRN" if it is found.

The BASIC program listed below, when run, will generate a file called "PRNFIX.COM" that can intercept DOS calls and change "LPT1" to "PRN". You can run this program using ZBASIC or the Z-100 or PC versions of GW-BASIC. The program will create PRNFIX.COM and then exit back to the system.

```
10 REM THIS PROGRAM CREATES PRNFIX.COM
20 OPEN "O",1,"PRNFIX.COM"
30 FOR I=1 TO 72
40 READ BYTE:PRINT #1,CHR$(BYTE);:NEXT I
50 CLOSE #1:SYSTEM
60 DATA &HEB,&H27,&H80,&HFC,&H3D,&H74,&H5,&HEA
70 DATA &H0,&H0,&H0,&H0,&H56,&H8B,&HF2,&H81
80 DATA &H3C,&H4C,&H50,&H75,&H10,&H81,&H7C,&H2
90 DATA &H54,&H31,&H75,&H9,&HC7,&H4,&H50,&H52
100 DATA &HC7,&H44,&H2,&H4E,&H0,&H5E,&HEB,&HDF
110 DATA &H0,&H33,&HC0,&H8E,&HD8,&HBE,&H84,&H0
120 DATA &HC4,&H3C,&HC7,&H4,&H2,&H1,&H8C,&H4C
130 DATA &H2,&HE,&H1F,&H89,&H3E,&H8,&H1,&H8C
140 DATA &H6,&HA,&H1,&HBA,&H29,&H1,&HCD,&H27
```

To use PRNFIX, load ZPC first, and then load PRNFIX. It will do it's "thing" automatically from then on. If you load ZPC using a batch file, add a line to load PRNFIX also.

Following this article, the source code of PRNFIX is listed, for those who would rather assemble it than create it from a BASIC program.

Turbo Pascal 3.01

Turbo Pascal version 3.01 will not run under ZPC because it contains a time delay controlled by the hardware timer interrupt, which is not available under ZPC. The software timer interrupt is available, however, and works just as well. It can be patched in, and Turbo will work under ZPC and will continue to work on a real PC. To make the patch, create a file called TURPCH.BAT that contains this line:

```
DEBUG TURBO.COM <TURPCH.DAT
```

Create a file called TURPCH.DAT that contains these lines:

```
E1A6
70
E1CC
70
W
Q
```

Copy TURPCH.BAT, TURPCH.DAT, and DEBUG.COM (from your MS-DOS disk) to your Turbo disk, log on to the disk, and enter

```
TURPCH
```

at the system prompt, and hit Return. The patch will be installed for you. **Note:** This patch is not required for Turbo Pascal version 3.00, or for any lower versions.

Word Perfect 4.1

To run WordPerfect under ZPC, first create a file called WPPCH .BAT that contains these lines:

```
REN WP.EXE WP.BIN
DEBUG WP.BIN <WPPCH.DAT
REN WP.BIN WP.EXE
```

Now, create a file called WPPCH.DAT that contains these lines:

```
F2A5B,2A66,90
A2A59
JMP 2A67
PUSH AX
MOV AL,20
INT 51
POP AX
INT 10
JMP 2AAA

A2A6B
LOOP 2A67

A2AA8
JMP 2A5B

A90
MOV AX,DS
ADD AX,1000
MOV DS,AX

G=90,97
F16D6,16E0,90
F21C6,21D0,90
ADS:FB20
JMP FB2F

W
Q
```

Copy WPPCH.BAT, WPPCH.DAT, and DEBUG.COM (from your MS-DOS disk) to your WordPerfect system disk, log on to the disk, and enter

```
WPPCH
```

at the system prompt, and hit Return. A patched WordPerfect will be created that you can run under ZPC. When WordPerfect requires a Control-Function key combination, press the F0 key followed by the required function key. When it requires an Alt-Function key combination, press the Help key followed by the required function key.

The QuickBASIC Compiler

Programs compiled under the QuickBASIC compiler will not run under ZPC unless they have been patched. Since the patching is done to the compiled program, not to the compiler, the address of the patch is going to vary from program to program. Because of this, I cannot present an exact patch, but I will show you a procedure to follow for locating the areas to patch. This procedure is for programs compiled with the /O switch (programs that run by themselves, without runtime support).

Assuming that you have a compiled program called PROG.EXE, first rename it so that the extension is .BIN.

```
A>REN PROG.EXE PROG.BIN
```

Load in the program with DEBUG and search for the hex number string BB, 0F, 05:

```
A>DEBUG PROG.BIN
S100,F000,BB,0F,05
1234:1280
```

In this example, the code was found at address 1280. Verify that this is the correct patch area by disassembling the code here. It should look like this:

```
-U1280
1234:1280 BB0F05        MOV     BX,050F
1234:1283 26            ES:
1234:1284 803F00        CMP     BYTE PTR [BX],00
1234:1287 7407          JZ      1290
1234:1289 33C0          XOR     AX,AX
1234:128B 07            POP     ES
1234:128C F9            STC
1234:128D E9D200        JMP     1362
1234:1290 26            ES:
1234:1291 FE07          INC     BYTE PTR [BX]
```

There will be more lines listed than we have shown here. The numbers may be different, but the basic instructions should be the same. Note the addresses of the JZ and INC instructions in the disassembled code, and alter the code at these points:

```
-A1287
1234:1287  JMP 1290          Replace JZ with JMP

-F1291,1292,90               Fill INC with NOP's (90)
```

Now search for the numbers 88, 1E, 0F, 05:

```
-S100,F000,88,1E,0F,05
1234:1377
```

Verify that this is the correct patch area by disassembling:

```
-U1377
1234:1377 881E0F05      MOV     [050F],BL
1234:137B 9D            POPF
1234:137C 7503          JNZ     1381
1234:137E EB6B          JMP     13EB
```

There will be many more lines listed than the ones we have shown here. Alter the code at the beginning of this area (the MOV instruction) as follows:

```
-F1377,137A,90
```

Note that the fill covers all 4 bytes of the MOV instruction. Now, write the patched file back to disk, quit DEBUG, and rename it back to .EXE.

```
-W
Writing 4567 bytes
-Q
A>REN PROG.BIN PROG.EXE
```

The program will now be ready for use under ZPC. Each compiled QuickBASIC program would have to be patched this way.

PrintMaster

PrintMaster will run without patching under ZPC (level 3), but when you select an item from any of its menus (by moving a

pointer with the arrow keys), the name of the item becomes unreadable. This is because PrintMaster is attempting to display the name in reverse video, and the method used involves a graphic character that is improperly defined in ZPC. You can patch the character tables in ZPC (any release of ZPC3 or ZPC3A) as follows: First, create a file called FIXGRF.BAT that contains this line:

```
DEBUG ZPC.COM <FIXGRF.DAT
```

Now, create a file called FIXGRF.DAT that contains these lines:

```
E2E8
18 3C 3C 18
EBE0
18 3C 3C 18
W
Q
```

Copy these files along with DEBUG.COM to your ZPC system disk, log on to the disk, and enter FIXGRF at the system prompt, and hit Return. The graphic character will be fixed, and the menu display of PrintMaster will be correct.

If you run PrintMaster and find that it does not complete something you are printing out, there is a patch you can apply to PrintMaster to correct the problem. (This problem only occurs with certain printers, especially very slow printers.) First, create a file called PMPCH.BAT that contains these lines:

```
REN PMMAIN.EXE PMMAIN.BIN
DEBUG PMMAIN.BIN <PMPCH.DAT
REN PMMAIN.BIN PMMAIN.EXE
```

Now, create a file called PMPCH.DAT that contains these lines:

```
A69CE
MOV DL,AL
MOV AH,5
INT 21
XOR AH,AH
NOP

W
Q
```

Copy these files along with DEBUG.COM to your PrintMaster disk (Disk I), log on to the disk, and enter PMPCH at the system prompt, and hit RETURN. PrintMaster will be patched to work correctly with your printer.

PC Write

To run the public domain word processing program, PC Write (version 2.4) under ZPC, create a file called PCWPCH.BAT that contains these lines:

```
REN ED.EXE ED BIN
DEBUG ED.BIN <PCWPCH.DAT
REN ED BIN ED.EXE
```

Create a file called PCWPCH.DAT that contains these lines:

```
F7C1B.7C1F,90
E7C25
90
E7C34
90
A77C6
JZ 77BF

W
Q
```

Copy PCWPCH.BAT, PCWPCH.DAT, and DEBUG.COM to your PC Write disk, log on to the disk, and enter PCWPCH at the system prompt, and hit Return. PC Write will be patched for use under ZPC, but it does not run as efficiently under ZPC as some other PC word processors, such as WordPerfect or WordStar.

PC-File

PC-File (a public domain filing program) can be run under ZPC without patching if you first run PC-SETUP and select option 2 from the first menu.

"Civilian" Multimate

The version of Multimate sold by Heath/Zenith has been altered to display a "Restricted Rights Legend" message (because it is sold to the government) on the screen when it starts up. I found that the "civilian" version of Multimate, that does not display the special message, cannot be patched for use with ZPC using the patches supplied for the Heath/Zenith version. If you have the other version, you can patch it by creating a file called MMPCH .BAT that contains these lines:

```
REM THIS FILE PATCHES MULTIMATE (CIVILIAN VERSION)
REN WP.EXE WP.BIN
DEBUG WP.BIN <MMPCH.DAT
REN WP.BIN WP.EXE
```

Create a file called MMPCH.DAT that contains these lines:

```
FFB20.FB24,90
FFB64.FB6D,90
FFC47.FC4B,90
FFC52.FC5A,90
FFC75.FC79,90
FFC84.FC88,90
FFCE4.FCED,90
EFD57
90
EFD6A
90
W
Q
```

Copy MMPCH.BAT, MMPCH.DAT, and DEBUG.COM to your MultiMate System disk, log on to it, and enter MMPCH at the system prompt, and hit Return to install the patch.

Assembly source listing PRNFIX.ASM.

```
        PAGE    ,132
;       PRNFIX -- THIS PROGRAM CHANGES OUTPUT TO "LPT1"
;       TO OUTPUT TO "PRN"
;
;       BY P  SWAYNE, HUG SOFTWARE ENGINEER
;
;       COPYRIGHT (C) 1985 BY HEATH/ZENITH USERS' GROUP

JMPF    MACRO
        DB      0EAH            ;DEFINE FAR JUMP
        ENDM

DUMMY   SEGMENT STACK
DUMMY   ENDS
CODE    SEGMENT
        ASSUME  CS:CODE,DS:CODE,ES:CODE,SS:CODE
        ORG     100H

START:  JMP     SHORT SETUP     ;SET UP PROGRAM
INT21   CMP     AH,3DH          ;OPEN FILE FUNCTION?
        JZ      INT21A          ;IF SO, PROCESS IT
INT21X: JMPF                    ;ELSE, EXIT
I21ADR  DW      0,0             ;INT 21 ADDRESS
INT21A: PUSH    SI              ;SAVE SI
        MOV     SI,DX
        CMP     WORD PTR [SI],"PL" ;CHECK FOR "LPT1"
        JNZ     NOTLPT
        CMP     WORD PTR 2[SI],"1T"
        JNZ     NOTLPT
        MOV     WORD PTR [SI],"RP" ;IF SO, REPLACE WITH "PRN"
```

```
        MOV     WORD PTR 2[SI],"N"
NOTLPT: POP     SI
        JMP     INT21X
        DB      Ø
ENDRES  LABEL   NEAR            ;END OF RESIDENT CODE

SETUP   XOR     AX,AX
        MOV     DS,AX           ;PUT DS AT Ø
        MOV     SI,OFFSET 21H*4 ;POINT TO INT 21 VECTOR
        LES     DI,DWORD PTR [SI] ;GET INT 21 VECTOR
ITSIN:  MOV     WORD PTR [SI],OFFSET INT21 ;SET NEW VECTOR
        MOV     2[SI],CS
        PUSH    CS
        POP     DS              ;FIX DS
        MOV     I21ADR,DI       ;SET UP EXIT TO OLD INT 21
        MOV     I21ADR+2,ES
        MOV     DX,OFFSET ENDRES ;POINT TO END OF RES  CODE
        INT     27H             ;EXIT, PROGRAM RESIDENT

CODE    ENDS
        END     START                                    *
```

The FORTRAN Formula — 6

Dick Stanley
P.O. Box 9512
Alexandria, VA 22304

Because of some unavoidable problems, there has been a significant lapse since the last FORTRAN Formula article was written, even though it hasn't been quite so long since it was published. So, before getting into the details of this installment, let's catch up on the mail.

First, thanks to all who have taken the time to write and telephone about this series. I'm pleased that it is proving valuable to so many people. I am taking your comments and suggestions seriously, even if I haven't had time to respond to each of you individually.

One problem has arisen which is becoming more common, and will take some work to overcome. The much publicized "decline of CP/M" is no where so much in evidence as in the lack of software available on CP/M disk formats. Even the current Heathkit catalog doesn't have software in H-89 disk formats. If you haven't found a FORTRAN compiler yet, this can be a problem. Many of those "closeout specials" referred to in an earlier article have been sold. So where do you go to get a FORTRAN compiler?

If you are interested in the language merely for familiarization, then a low-cost pseudo-compiler like Utah FORTRAN (Formerly, Nevada FORTRAN) is probably adequate. It is still available in CP/M formats from Ellis Computing (5655 Riggins Ct. Ste. 10, Reno, NV 89502 (702) 827-3030). On the other hand, serious work in FORTRAN really requires a true compiler.

The Microsoft FORTRAN-80 compiler is still available from some suppliers, such as First Capitol Computer (1106 First Capitol Drive, St. Charles, MO 63301). However, you can expect to pay close to list price for it (around $395). Nevertheless, if you are serious about using FORTRAN, this is a good compiler that has been thoroughly exposed to many users. It is not without bugs, but those that exist are known and well-documented.

The last remaining bastion of CP/M support seems to be centered around the Osborne and Kaypro computers. One supplier that still carries a wide variety of CP/M software — and even continues to develop new items — is Spite Software (P.O. Box 8918, Portland, OR 97207 (503) 224-0137). Although I haven't seen any FORTRAN compilers in their latest catalog, they would know where to find one, if anyone would.

Before you rush out to buy a compiler, it would probably be in order to review the comments on FORTRAN compilers that appeared in "The FORTRAN Formula — 1" (REMark, October 1984).

Now, on to the installment in our pursuit of stock market profits and FORTRAN expertise!

More About Moving Averages

In "The FORTRAN Formula — 5" (REMark, November 1985) you learned the concept of a moving average. Moving averages are used in many fields, not merely stock market analysis. This is because it is often important to be able to extract a trend from seemingly random data. Clearly, that is always desirable in the stock market. You want to know if a stock is moving up or down. And you also want to know when the up or down motion changes direction.

A moving average is a mathematical low-pass filter. The longer the duration of the moving average, the longer the term of the trend it reveals. If you calculated a 200-day and a 30-day moving average of the same variable — say, the Dow Jones Industrial Average — you should not be surprised to find that the two moving averages may not be trending in the same direction. If the 200-day average yields higher and higher values day after day, you can comfortably state that the DJIA is in a long-term uptrend. Long-term because 200 days is a long moving average, remember?

While the 200-day line is trending higher, however, the 30-day line may trend down, reverse itself and head upwards again. This reflects only that the shorter-term (higher frequency) trend revealed by the short term moving average is more variable than the long-term average. Life is full of short-term trends riding on top of long-term trends, so this is not a startling revelation.

A moving average changes direction when the most recent data point falls below the most recent value of the moving average. If the moving average of the DJIA is trending upward and yesterday's closing value was 1450.02, if the DJIA closes today at 1449.56, the upward trend has been broken. You can observe that this is true by calculating a few moving averages and observing when the value of the average changes from up to down. Knowing this criterion for determining when a moving average changes direction is important in the use of these averages for following trends.

Doing It Again And Again . . .

If you review the program we have developed so far, you will see that there at least 16 quantities that are candidates for analysis using moving averages. We may also want to calculate moving averages of different durations to check trends of varying persistence. Surely, you won't want to write the code to calculate a moving average each time you need to calculate one. If nothing else, that would make the code much longer than it needs to be. And you really don't want to be locked into a certain duration of average simply because you wrote the code for say, a ten-day duration.

The way to get around this problem is to write a small program to calculate the value of a moving average whenever it is needed. You are already used to this in FORTRAN. Whenever you need to calculate the value for the sine of an angle, for example, you simply write a statement like

```
A = SIN(ANGLE)
```

This assignment statement causes the value of the sine of ANGLE to be assigned to the variable A.

What happened here? You called a subprogram named SIN to calculate the sine of the angle, and passed it the value of the angle stored in the variable called ANGLE. The subprogram did the needed calculations and returned the proper result to your program. A subprogram that performs in this way is called a function.

The SIN function subprogram was written by the folks who wrote the compiler, because calculating sines is a common need for FORTRAN programmers. However, the language is structured to allow you to write your own functions. And that is just what we are going to do to calculate moving averages.

Before starting to write a function subprogram, though, let's examine how we get information into and out of it. In the example above, you saw the basic limitation on output: a function must return one, and only one, value to the calling program. That value is obtained by referring to the function. The value calculated by the function need not be stored in a variable; it can be a part of a calculation. Consider the calculation below:

```
A1 = SQRT(B7) + SIN(ANGLE3)/2
```

SQRT and SIN are both function subprograms supplied with the compiler. (SQRT calculates the square root of the argument given it). Here, neither function result is placed in a variable, but the result of each is operated on mathematically and the result is placed in a variable.

So far, we have only seen instances where one value — called an argument — is passed from the calling program to the function subprogram. However, there is no reason this must be so.

What must we tell a function so that it has enough information to calculate the moving average of values stored in an array? Unless we decide to write a function to calculate a moving average of only one duration, we need to pass to the function both the name of the array in which the values to be averaged are located, and also the duration of the average. This is done by putting these arguments in the parentheses after the function name, separated by commas, like this:

```
FUNCTION(ARG1, ARG2)
```

Writing The Function

Now that you know the basics, Figure 1 shows the code for a function to calculate a moving average. Let's take a look at this function subprogram, line by line (that's why there are reference numbers on each line in Figure 1).

```
      FUNCTION AVGM (ARRAY, LENGTH)          1
      DIMENSION ARRAY(60)                    2
      SUM = 0.0                              3
      DO 3000 J=1,LENGTH                     4
3000  SUM = SUM + ARRAY(J)                   5
      AVGM = SUM/FLOAT(LENGTH)               6
      RETURN                                 7
      END                                    8
                                             ^
                                             |
   These numbers are for reference only, --+
     they are NOT part of the code
     Do not enter them in your code
```

Figure 1
A function subprogram to calculate a moving average.

Line 1

Right away, we notice that this line begins with the word FUNCTION. That is mandatory. The FORTRAN reserved word FUNCTION tells the compiler that everything that follows this declaration is a function subprogram, until the compiler is told the end of the subprogram has been reached. Next comes the function name, by which we will call it to do its calculations from the main program. I chose to name this function AVGM for two reasons: (1) the name suggests what it does (AVG for average, M for moving), and (2) the choice of name coincides with the type of result we expect to obtain from the procedure, namely floating-point (real). Had I named the function MAVG or MOVAVG, it would have returned an integer value for the moving average, which is not what is desired. If this seems confusing, review the rules on naming FORTRAN variables in the second article of the series.

Finally, enclosed in parentheses are the variables that will be provided to the subprogram by the main program. The names used here are "dummy" variables. That means that these variable names are used only to write the function subprogram. When the subprogram actually executes, these names will be replaced — here and throughout the subprogram — by the names or numbers you provide when you call the function. There are two reasons for using dummy variables. One, they provide a convenient way to write the subprogram without knowing what the names of the variables in the main program are. Two, they give generality to the function. If there were two arrays, called DJIA and TRIN (funny how we chose those names, isn't it?), we could obtain their ten-day moving averages by using the following statements in the main program:

```
C This calculates the 10-day moving average of the DJIA
DJIA10 = AVGM(DJIA, 10)
```

```
C This calculates the 10-day moving average of TRIN
TRIN10 = AVGM(TRIN, 10)
```

If you had used the actual array names in writing the subprogram, you would have needed a function for each array. That is not too helpful, so the use of dummy variables permits more general use of the procedure.

Notice that the dummy variables are named in accordance with the type of variable that is to be passed. ARRAY is a variable to pass the name of an array. LENGTH is an integer variable to pass the length of the moving average to the function. Since it is the index of a DO loop, this value must be an integer, so the variable is named accordingly.

Line 2

If an array name is passed to a function, as we want to do here, the array must be dimensioned inside the subprogram. This is done by dimensioning the dummy variable within the subprogram, as has been done here. Arrays must also be dimensioned in the main program, and the dimensioning inside and outside the subprogram must be consistent. It happens that all the arrays that we may want to calculate moving averages of in the STOXIN program are of size 60. Therefore, we can dimension the dummy array to 60 elements because all the main program arrays are already dimensioned to 60 elements. We cannot dimension the dummy array to be either larger or smaller than the real arrays, or problems will arise.

Line 3

To form the average, we are going to accumulate a total in a variable called SUM. This statement initializes SUM to be zero before we begin, to make sure we get the right result.

Line 4

This statement begins the DO loop that examines the elements of the array beginning with element 1 and ending with the element whose number corresponds to the value of LENGTH.

Line 5

This line accumulates the total of the desired array elements in the variable SUM. The DO index, J, is used as a pointer to each element in turn. The last array value used will be the one numbered equal to the value of LENGTH, which is just what is needed to calculate a moving average of that length.

Line 6

Here the subprogram calculates the value of the moving average by dividing the accumulated total of the average elements by the duration of the average. Since the length of the average is an integer variable, we must convert it first to floating-point before dividing SUM by it. This integer to floating-point conversion is performed by the FLOAT function, which is included in the library supplied with the compiler.

This line illustrates a vital point about the function subprogram. The name of the function MUST appear on the left side of an assignment statement somewhere within the subprogram. The name must be identical to the name of the function on the first line. This step is needed so that a value can be determined for the function, and that value assigned to a variable of the same name so it can be passed back to the calling program. Remember, a function can receive many inputs, but it can return only a single output.

Line 7

Every function subprogram must contain at least one RETURN statement. This statement returns control to the main program at the next executable instruction after where the function was invoked. If there are several routes through a function, depending on the values of the inputs, there may be several RETURNs, one for each path. There must be at least one, however, no matter what.

Line 8

A function subprogram must end with an END statement. This tells the compiler that the end of the subprogram has been reached. There can be only one END statement, and it must be the last statement of the subprogram.

You should now see how the function is constructed, and the basic items it must contain. The calculations may be considerably more complex than those done in this example. The beauty of the function subprogram is the ability to call for complex or repetitive calculations from the main program by merely invoking the function name and giving it the required arguments. This is unlike BASIC, where most versions will permit functions of only a single line.

From the above discussion, you have also seen how to use a function subprogram. You merely place it on the right side of an assignment statement, and its value is returned and used as defined by the assignment statement. This is exactly the same way that you calculate a square root (using the SQRT function) or a cosine (using the COS function), so the concept is familiar.

Getting More General

The function subprogram just defined in Figure 1 is a useful subprogram for calculating moving averages. At the moment, there does not appear to be a need in the STOXIN program to calculate moving averages based on arrays of length other than 60. However, in further development of that program, that might change.

If the size of the arrays to input to AVGM changes, another function must be written to accommodate arrays of that size. Or must it? Can we pass the size of the array being used to the function as one of the arguments?

Yes, we can. And if we do, we make the subprogram considerably more flexible and more general. Figure 2 shows AVGM rewritten to pass the array size, as well as the other arguments to the subprogram.

```
      FUNCTION AVGM(ARRAY, LENGTH, ISIZE)
      DIMENSION ARRAY(ISIZE)
      SUM = 0.0
      DO 3000 J=1,LENGTH
3000  SUM = SUM + ARRAY(J)
      AVGM = SUM/FLOAT(LENGTH)
      RETURN
      END
```

Figure 2
More general function to calculate moving averages.

Notice that in Line 1 a new argument, ISIZE, has been added. This argument provides the subprogram with the size of the array being used, so that it can be properly dimensioned in the subprogram. This is an integer variable, and is therefore, named so that the default type assignment of FORTRAN will treat it as an integer. In Line 2, ISIZE is used as the argument to the DIMEN-

SION statement, so that array has the same dimensions inside and outside the subprogram.

In either version of AVGM, notice that if LENGTH > ISIZE (or LENGTH > 60 in the Figure 1 version), an error will be returned. Clearly, we cannot count beyond the end of an array. This is not a problem with subprograms, but a routine consideration when dealing with arrays — you must be careful that everything is properly sized.

Moving Right Along

By now, you have acquired the expertise to write your own basic FORTRAN programs. The best way to learn how is to do it, so here's what needs to be done to try out the AVGM function:

1. Write a short program to input values to an array of whatever size you choose. You can either input them from the keyboard, or you can generate them from within a program. If you input from the keyboard, however, you can use real data from something that interests you.
2. Write another short program to print out the values you have stored, so that you can verify what is in the array. The listing may be either on the CRT or on the printer, but it should be of a form that lists the array element number and then the value stored in that element, similar to:

```
3       37.6
4       33.5
5       38.8

        etc
```

3. Write a short program segment to calculate moving averages of the array elements. The averages should be of different lengths, and your program should accept inputs from the keyboard about the length of the moving average desired. Confirm the accuracy of the subprogram by manually calculating a few of these moving averages manually or with a calculator. If there is a difference, something is wrong!!

Examples of all the above types of subprograms are contained in previous installments of this series. In fact, the consolidated STOXIN listing contained in "The FORTRAN Formula — 5" (REMark, November 1985) illustrates everything except the function subprogram needed to calculate the moving averages.

When you are sure the calculations work, modify your program so that it contains two arrays having different names. By entering the name of the desired array, the desired moving average duration, and the size of the array, you can observe the functioning of the AVGM subprogram in calculating moving averages from whatever array it is told to handle.

The next article in the series will present one solution to the above problem. The input routines that you will write are not going to be wasted — we have not yet provided a means to fill the arrays used in STOXIN with real data, and that is how it will be done. There are many ways to solve the problem, not just one, so don't be disappointed if your solution and your friend's don't agree. If it works, it is a valid solution.

We are almost done laying the foundation for our FORTRAN endeavors. Far from being a dying language, FORTRAN continues to be one of the most widely used programming languages in the world. It is still being used in the development of many state-of-the-art systems, and promises to be prominent for a long, long time. Once the groundwork is finished, we will see how to maximize the benefits of FORTRAN, and also how to overcome its apparent shortcomings, such as string handling. We'll also point to large stores of public domain FORTRAN software, and review some of the better items. The fun is just about to start — stick around. ✱

Evaluation Of Word Finder Synonym Finder

Bill Schiffbauer
Senior Technical Writer
Heath Company

I was recently asked to evaluate the Word Finder Synonym Finder software package from Writing Consultants. Being a Senior Technical Writer for the Heath Company, I eagerly accepted. I used Synonym Finder with WordStar Version 3.31 on a Heath HS-148-41 Computer with optional second disk drive.

I found this software package very easy to install and use. To install Word Finder, you must run the file WFINSTAL.EXE. An interesting feature is the user registration portion. WFINSTAL prompts you to enter your name and address. Only upon entering this information are your software Registration and Serial Numbers displayed. Writing Consultants requests that you enter these numbers on your License Agreement and return the form to them. They use these numbers to verify ownership for program support and any future upgrades or enhancements. After registering your software, WFINSTAL prompts you to select the word processor you will be using. To make this selection, simply move the arrow to the word processor of your choice and press the RETURN or ENTER key. Word Finder supports eight dedicated word processors. It also includes a "Generic word processor" to allow you to try a word processor not on their dedicated list.

ENHANCE.EXE allows you to customize Word Finder. You can change the default disk drive, change the command used to open the window of synonyms, and change its opening screen message to remind you of the command you have selected. Since their default command is easy to use and did not conflict with any of my WordStar commands, I used the default command of CTRL-F6.

To use Synonym Finder, simply enter SF and press the ENTER or RETURN key. Once loaded into memory, your system will display a general message telling you that Synonym Finder is loaded into memory. At this point, load your word processor and you're ready to begin.

After Synonym Finder and WordStar were loaded, I was ready to begin. To use Synonym Finder, simply place the cursor at the beginning of the word you wish to find a synonym for and press CTRL-F6. Synonym Finder then displays the selected word in reverse video and displays a window of synonyms for the highlighted word. The display also includes information concerning the synonyms: synonyms are presented by part of speech (n = noun, v = verb, adj = adjective, adv = adverb). If a word has more than one meaning, semicolons separate each meaning group. If you decide to replace a word with a displayed synonym, place the cursor at the beginning of the synonym and press the RETURN or ENTER key. Synonym Finder automatically replaces the word in your text file.

The following is a list of my observations (good, bad, or indifferent) while using Synonym Finder:

1. During installation (WFINSTAL), the program randomly produced "9s" on my display. Backspacing eliminated the 9s and produced no additional problems for WFINSTAL.

2. The first time I selected a synonym, the program randomly produced approximately two lines of "9s." These were deleted using WordStar.

   IMPORTANT: It should be noted that I have never again observed these two problems.

3. Once you replace a word with a synonym, WordStar does not automatically reform the paragraph (if necessary). Since you immediately return to the WordStar edit mode, you can use CTRL-B from WordStar to reform the paragraph. This is not really a problem; being accustomed to CorrectStar, I thought

Synonym Finder may react accordingly (automatic paragraph reform).

4. If you use CorrectStar with WordStar, your disk may not have enough space to include the Synonym Finder files. Therefore, I created two WordStar disks, one with WordStar and CorrectStar and one with WordStar and Synonym Finder. This is not a problem. However, it does mean "disk swapping" to run both CorrectStar and Synonym Finder.

5. Synonym Finder displays only the root of the word when looking for the participle or past tense of a verb. For example, if you are looking for a synonym for the word "previously", Synonym Finder displays alphabetically the 30 closest words. Among these is the word "previous." If you continue, it displays synonyms for the word "previous." If you select the synonym "premature", Synonym Finder replaces the word "previously" with "premature." You must then add the "ly" suffix while in WordStar. This also happens with words ending in "ed", "ing", etc.

6. When you look for a synonym for a word in "special print" (for example: bold, italics, postscript, etc.), you must place the cursor under the first letter of the actual word. For example, if you look for a synonym for ^PBINTRODUCTION^PB (marked for bold print), you must place the cursor under the letter "I". If you place the cursor under ^P instead of the "I", Synonym Finder will not be able to find a synonym.

7. Writing Consultants warn you that words identified for special print (bold, italics, supscript, etc.) may not always be replaced in the same mode with WordStar 2000. I did not experience this problem with WordStar Version 3.31. The word I replaced was always replaced in the same mode.

Synopsis

I found Synonym Finder very easy to install and use. Although some of my observations may sound derogatory, none of the "problems" listed outweigh the benefits of Synonym Finder. To incorporate some of the features that I think would be nice (not necessary, but nice) would be a monumental task. A feature I would like to see incorporated is to allow the user to add words to Synonym Finder's dictionary (similar to CorrectStar's dictionary). I'm sure that Synonym Finder will be a great tool for my writing career. I would recommend this product to anyone for recreational or professional writing. *

Computers In Europe

D. C. Shoemaker
HQ USEUCOM Box 897
APO NY, NY 09128

A surprising number of you reading this article will have the opportunity to travel to Europe, either in the course of business or possibly at the invitation of a relative. Uncle Sam, perhaps. If you're a dedicated, dyed-in-the-wool computer hobbyist, one of your first questions will probably be "Can I take my system with me?" Since you're already either a Heath/Zenith owner, or at least giving it some serious thought (after all, you're reading this magazine), the short answer is yes, you can take it with you. The longer answer, as usual, is more like "You can take it with you, but . . ." It's the "but" we need to look at.

The problem that causes most computer users trouble first is power. People with Radio Shack, Apple or IBM computers can't just waltz into Europe, plug their system in and begin to compute. Any computing done thusly will be extremely energetic and notably brief, since in Europe the power is 220 volts at 50 Hertz (cycles), not the 110 volts 60 Hertz you're used to at home. To be sure, you wouldn't be able to just plug in and blow up. There's a different type plug and socket system used in Europe, so that your computer can't be connected without an adapter. More on the physical problems later.

Fortunately, since you're a Heath/Zenith owner, you have a tremendous built-in advantage you may never have considered. You may not even be aware of it, but your Heath-designed computer product, no matter what it might be, comes equipped for dual voltages and line frequencies. Usually at the flip of a switch or two, you can convert from 110 to 220 volts, and from 60 Hertz to 50 Hertz. In some cases, you may have to open the computer, printer or terminal and resolder a jumper, or even move a wire from one terminal to another. Those of us who have been world travelers for some time, consider this capability to be one of Heath's best features in the long run.

If you built your component from a kit, your assembly manual told you how to configure the kit as you were building it (you do still have the manuals, don't you?) If you bought the equipment already built, your operating manual will have the same information.

Basic reference material is a must. The equipment manuals are obvious, as is your software documentation. How about reference books? If you think you'll need an Intel or Zilog manual, get it now. If you think you might want a reference on BASIC programming, get it now. The hardest way to buy books is by mail. It's slow, frustrating, and 50 percent of the books advertised are worthless.

What if you have some non-Heath components? For instance, the ZVM-135 is an excellent color monitor that thousands use with their H/Z-110 and 120 color computers. That's not a Heath product, it's from Zenith. There's a crucial difference. Most Zenith products are provided for either 110 volts 60 Hertz or 220 volts 50 Hertz. No switch selection. Many of the Heath/Zenith-labeled printers are actually made by someone else (Epson, for instance) and are in the same category. What to do about those?

There are two aspects to this problem. First, does the equipment really require 50 Hertz, or was that put on the label to conform to European market requirements? Only the maker knows for sure. Ask them if you can. In many cases, you'll find that the equipment doesn't really care; the alternating current is immediately rectified to direct current, then put to work. Printers usually fall into this category. In such cases, it won't usually make any difference whether you use 50 or 60 Hertz. Using the wrong frequency power can occasionally cause overheating problems in transformers, so it's a good idea to ask to be certain, whenever possible. Naturally, some equipment is already so marked, saying 50/60 Hertz. (An aside: Current usage dictates the term Hertz, with a capital H, in honor of Wilhelm Hertz, a 19th century German physicist whose work on electricity led to the practical application of alternating current. His name has replaced the older term "cycles." For some reason, we do not honor the Italian scientist Alesandro Volta, the discoverer of "animal," or current electricity, as distinct from static electricity, in the same fashion.)

If it turns out that the equipment really does require 60 Hertz and just won't work properly on 50, the maker may possibly have a modification kit to allow you to make the conversion. That's another good reason to ask. If not, then you might look into the possibility of selling or trading in the offending piece. Frankly,

there's very little well-designed computer-related equipment that I've come across, either in the US or here in Germany, that can't be operated on 50 Hertz. One important exception: 8" disk drives. These devices have AC motors, unlike their smaller DC brothers, so they're out. They may run, but not reliably. And since reliable storage is the reason you bought 8" disk drives in the first place, leave the 8" drives at home, perhaps with a trusted friend. Second, remember who it was who told you that 8" drives were more reliable, better, etc., than 5-1/4" ones. He or she is living in the past, and passing out old advice. If you're contemplating assembling a system from basic components to bring with you, you should think seriously about staying with the 5-1/4" drives because to accommodate the European power frequency means buying 50 Hertz drives, and that's even worse than bringing 60 Hertz drives with you, since they would be useless upon your eventual return.

Lest you dream about bringing your computer system to Europe and selling it before you return, a word of caution. It may seem like a good idea to bring, or later buy, European-standard components with that in mind, but you run the risk of running afoul of customs problems, both in the country you visit and the US. Selling something as expensive as a computer without paying the import duty can net you a hefty fine, and possibly a jail term (gaol in England.) And in the case of the H/Z-100 series computers, I well recall that before Heath would ship my kit, I was required to sign a statement to the effect that I would not sell the computer before my return to the US. The Department of Commerce at one time looked on the H/Z-100 series as sufficiently high technology to restrict its sale outside the US. I do not know if this is still the case; I rather suspect not, but it's hard to be certain.

If you've been able to determine that your hardware will operate on 50 Hertz, the next question is voltage. If at all possible, run everything on the same voltage. This reduces the hazzards of ground loops and other circuitry problems that could result in a very energetic signal passing from one piece of equipment to another. It's quite possible to develop a 110 volt potential difference in the grounds of two pieces of equipment, one of which is on 110 volts and the other on 220. Another good reason for keeping the voltages the same is that you can use a power strip that lets you plug everything into one circuit. Handy for emergency power-down. More on this later.

It isn't earth-shakingly critical whether you elect to use 220 volts or 110. If you stay with 110, there will be one less thing to worry about when you configure your equipment for the trip in either direction. On the other hand, you'll need a transformer. This poses several problems in and of itself. First, you'll need one that can handle the total current of your system. Computers don't draw a lot of current, but it's enough that the transformer will be fairly large and heavy. If you have an H/Z-110, an H-25 and a color monitor, you should have a transformer rated for at least 1000 watts. That's a lot more than your system will draw, but it keeps the transformer running cool, and allows for a possible expansion without going out to purchase another transformer. Just one 1,000 watt transformer can cost up to $100.00 in Germany. It will be a very nice one, with its own switch and fuse, but it's still $100.00. And it will be useless to you upon your return to the US, so you'll probably sell it before you leave. Of course, you could use it for other things when you're not computing, like your US-made hair dryers, microwave ovens and the like. But since other family members may need their devices, it's inconvenient having to switch back and forth, and expensive to have several transformers. Transformers are really a bother, all in all.

My personal experience leads me to suggest that it's better to set the power supplies of your Heath equipment to use the 220 volt power you have available. This way you can dispense with the transformer and not worry about how much current you're going to need when you add that nine-track drive and laser printer. However, in Europe nothing is either easy or simple for the American computerist; you also have to deal with power plugs. This is a problem peculiar to the Europeans. They don't have electrical standards in the sense that we do in the US. Most countries are internally consistent (but not always; England must have a dozen different plugs), but often differ from one country to another. Since you'll most likely be staying in one place while you're in Europe, you would probably be wise to make up a power strip (or buss or bar). Ideally, it should contain some sort of surge protection, but at a minimum there must be provision for proper grounding.

If you're unsure of how to wire a power strip with surge protection, you might refer to Byte magazine's article in the December 83 issue, page 36. Either find someone who can build it for you or gather the parts and bring them with you. It won't be difficult to find someone who can show you how to do the wiring, or who can do it for you. You won't be able to find all the parts, however. The plug, for example, probably can't be found in the US (you'll see why when you get to Europe!) Just bring the surge protection devices and conventional three-pin US-type AC sockets. Of course, you can just buy a conventional power strip, such as the ones sold by Heathkit dealers or Radio Shack stores. They can be obtained with surge protection. The plug on the end of the cable can be replaced in Europe.

My next main point follows from this. Much, if not most, of what you will need is not available anywhere in Europe. The Europeans as a whole are several years behind the US in personal computing. This is partly due to cost; a computer the equivalent of my H-120-22 would cost a German enthusiast well over $5,000. One reason is that much state-of-the-art computer equipment does not come from within the Common Market, but from the US. Even the Japanese equipment is expensive, and not all that readily available. Heath/Zenith computers are imported. Some other big computer names come from Ireland or England. There are a few US names in wide evidence, the biggest being Commodore. So unless you plan to trade in your Heath for a Commodore, come prepared.

Common items like disks are readily available, and generally less expensive. I pay less for Memorex disks than you do, regardless of what discount house you use. When you add postage, the cost difference can be $5.00 a box or more. Other common items like printer paper are also readily available, but are of slightly different sizes. There's a European standard for paper (there seems to be a European standard for everything!) that hasn't caught on very widely in the US, but it's coming. If you don't mind the odd size, you'll have no problem. Paper products, in general, cost more in Europe than in the US (almost all bulk wood products are imported.) I generally have paper shipped to me from the US. You should figure on about $40.00 a box for 16-pound paper, delivered. What applies to paper applies to things to put it in. If you generate a lot of listings that you like to keep in binders, bring some binders, too. The European ones won't fit US paper.

Ribbons for your printer may or may not be available. If yours is an IBM-related product, or a Diablo or one of the hundred or so MX variants, you're probably all right. You won't find ribbons for your H-25 or some of the other "exotic" printers common in the

US. The solution for the H-25 and some of the others, naturally, is mail order. For other printers, the best advice is to bring what you'll need for the duration of your stay. They won't cost all that much, and the peace of mind will be worth it.

Spare parts are a mixed lot (no pun intended.) Most common electronic parts are available if you can find an outlet. Most large cities have parts stores. Most small ones do not. And of course, there's no running down to the local Radio Shack at 8:00 PM for a fuse or a resistor. Most European shops open at 9:00 AM and close at 5:00 PM, if not before. If they're open on Saturday, it's usually 9:00 to 12:00 in the morning, and closed Sundays, virtually without exception. While I wouldn't go so far as to recommend buying a basic load of resistors and capacitors, you definitely should bring three or four of every kind of fuse you need. And be certain that the values are correct for 220 volt operation. Remember that Heath generally recommends two values, one for 110 and one for 220. Get the right ones.

Other parts may or may not pose problems. I've had no trouble with common IC chips. In fact they're often cheaper, especially 64K memory chips. Others can't be found at all, and a few are tremendously expensive. Fast microprocessors are hard to find and costly. Resistors, capacitors, and most hardware are easily found, if you can locate a store. However, mail order is often the only practical resort.

There are some parts that are difficult to mail order and would be worth stocking up on. A good example is parts for your disk drive. A kit of the parts most likely to fail is available from Workman & Associates, 112 Marion, Pasadena, CA 91106. You might want to drop them a line, tell them what type drives you have, and get a complete price list (they have some other neat stuff, too.) You'll find that in many cases, Tandon, for instance, the drive maker won't deal with you at all, but for some reason it's often difficult to find a retailer who stocks Tandon drive parts. I ordered a kit from Workman; I haven't used it, but I sure feel better for having it.

A few words about mail ordering from Europe might not be amiss. If you have a major credit card, that's often the easiest. Most US parts dealers, and of course, Heath themselves, will take cards. And since you often won't know just what a widget will cost, including postage, that's by far the easiest way to order. COD is, as you might expect, virtually non-existent. And you can forget UPS. We have UPS in Europe, but it's not the same. If you prepay your order, a postal money order is faster than a check. Many dealers don't ship until the check clears, and that can take up to ten days for a check drawn on a European account. A money order is more like cash, and most vendors will ship immediately on an M.O. Naturally, this is only available through US facilities, the military and so forth. If you won't have access to that, you're almost forced to use plastic money.

All this holds true for software, too. You'll find little you can buy in Europe, in part because of the rarity of Heath/Zenith computers, and in part because most European countries have signed up for the computer revolution to the extent that programs are now available in the native language. Until quite recently, this wasn't so. In fact, one of the greatest ways to meet Europeans who not only shared your interest in computers, but spoke very good English was to join a local computer club. I still recommend that highly, but you also meet lots of European computerists who don't need to know English to compute. The professionals still do, though.

Books and magazines are the same story. Three or four years ago in Germany, you could find all the US publications, in English, anywhere they sold computer-related items. That's largely disappeared. There are numerous excellent magazines in German, and most important books are available in German. Most likely, if you want a specific book, you'll have to mail order it just like the software you need. Don't forget your subscriptions. I recommend at a minimum all the Heath-specific periodicals you can find. They will often be your only source of ready information about what's going on in the world of Heath/Zenith computers. A subscription to REMark is a must.

Mail time from the US to Europe is quite fast, provided the items come directly to you at a civilian address. If they go into the military mail system (the Army Post Office, APO, or the Navy's Fleet Post Office, FPO) it's another story. Allow ten days each way, then you can be pleasantly surprised if something comes sooner. Once out of ten times.

Watch out for customs problems. If you have a piece of equipment shipped directly to you at your European address (not through the APO) it will have to clear that nation's customs. What happens depends on the nation, the value of the item, how the customs inspector is feeling that day, and possibly the phase of the moon. Germany is generally very good; if you can demonstrate that it's for your own use, and you're in Germany on Government business (ours, not theirs), there's usually no problem. Other countries may be more troublesome. It seems generally better to bring it with you (personally or in your household goods) than it is to buy it later and have it shipped to you. If you're in the military, use your military address and avoid all the trouble. The delay in the mail can be worth it.

Customs works both ways, and you can have problems on the US end, too. For instance, if you return a piece of defective equipment, or one you merely wish to exchange, be sure to indicate on the customs declaration (the form you'll have to fill out at the post office) that the item is "Returned US Merchandise." Use those words, and you should have no trouble. Forget, and US customs may impound the item, causing endless delay.

Naturally, wrap anything you send back as if it were a really valuable piece of equipment, which of course it is. Air mail is probably the best, not only from the standpoint of time, but for better handling, as well. Less exposure time, I suppose. Air freight is the best, but it's expensive. To ship an H-89 in an Anvil professional case from the east coast to Frankfurt, Germany, will cost about $300.00. If you need it fast, and want it working when you get it, that may not be too expensive. That's how I shipped mine, and it was worth it to me. A trip to the airport will be necessary to pick up the computer (no delivery unless you arrange for it yourself.) If you're not fluent in the language of the country, be sure to take a friend who is. Customs is tough enough without the language barrier.

When you wrap, double cartons and lots of plastic peanuts will help. Send back only what you must. If you're certain that your H-19 terminal problem, for instance, lies in the timing and processor board, then send just that back. Try not to ship the CRT at all. A little correspondence with Heath's technical consultants can save a lot of shipping cost. And worry. Even better, try to keep the original shaped styrofoam packing material. Try to make sure that the carton is waterproof. If you have to ship your disk drive back, be sure to put an expendable disk or a cardboard simulacrum in the drive and close the door. This is especially essential for double-sided drives, where the heads would bang against

each other without something between them. Most reputable repair shops will do the same when they return the drive. It wouldn't hurt to remind them.

If you keep all these things in mind, you can take it with you to Europe, and you can expect it to run well and faithfully. A little foresight (and a few friends in the US you can turn to for help with purchases) will make all the difference. There are lots of others here in the same boat, ready and willing to help. Computers are a great hobby and useful tools, and that upcoming trip to Europe needn't interrupt your fun. Come and enjoy! *

Files, FATs, And Directories

William M. Adney
P.O. Box 531655
Grand Prairie, TX 75053

Although the original article that I planned was related to MS-DOS subdirectories, I/O redirection, and command pathing, it became obvious that the combination of subjects took more space than I originally thought. In addition, a number of people have asked me about various items related to files, file allocation tables (FAT's), and how all of this is related. Since the use of subdirectories is certainly not obvious until you understand the use of the ROOT directory, we will spend some time looking at the way MS-DOS generally handles files.

Unless otherwise noted, all comments in this article will apply to both the Z-100 and the Z-150 series (including all Zenith compatible IBM PC) computers. Many of the techniques used by MS-DOS are similar to those used in CP/M, since MS-DOS is based on a number of CP/M concepts. Due to space and the popularity of MS-DOS systems, I will not discuss the differences between MS-DOS and CP/M. Suffice it to say that although the end result is very similar, the technical implementation is significantly different in the way that CP/M handles file and directory related information. And by the way, I will also use DOS (Disk Operating System) and MS-DOS to include both MS-DOS and PC-DOS. Although CP/M (Control Program for Microcomputers) is also a DOS, any specific comments on CP/M — whether the 8 bit or 16 bit variety — will talk about CP/M to differentiate it from DOS.

The concepts presented in this article are, in some cases, complicated, however, it is important to know how MS-DOS works with files so that you can understand what can (and sometimes does) go wrong. Many of the error messages that are displayed by the CHKDSK program can be confusing. These errors can be caused by application program bugs or errors within the DOS itself. Although some of these problems can be corrected by the CHKDSK program with the /F (Fix errors) parameter, it is important to understand the real meaning of the error message and the probable fix that CHKDSK will make. For example, if you see the error message: "x lost clusters found in y chains", what should you do? And how does CHKDSK recover from this error? What should you do next?

What Is A File?

A File is a special group or set of related information. In mainframe data processing, we also use Data Set which essentially means a set of related data. Both names, File and Data Set, can usually be interchanged since they mean basically the same thing.

There are two basic types of files: Program (or system) files and Data files. Program files include those system programs that came with the DOS distribution disks such as FORMAT.COM, CHKDSK.COM, and COMMAND.COM. In addition, program files include application programs such as WordStar, SuperCalc, and dBase. In these kinds of discussions, I also like to include the idea that batch files, like AUTOEXEC.BAT, are also included as a special kind of "program" file. In general, program files include just about any file that is used in your system. That includes files that have a file type of EXE, COM, BAT, or SYS (e.g. the BIOS, MSDOS.SYS or the configuration file, CONFIG.SYS).

Data files, on the other hand, include just about everything that is created by an application program: correspondence created by a word processor, spreadsheets or databases. These files can be named by the user. For example, a data file can contain a letter to the Heath/Zenith Users' Group. That fits our definition since the letter is a set of related information. Similarly, a SuperCalc spreadsheet file (e.g. BUDGET.CAL) is also a data file.

Many of the introductory texts discuss data files in terms of the common manila file folder. Indeed, many of the so-called user friendly computer interface displays, such as that used on the Apple Macintosh, use an icon (picture) of a file folder to represent a data file. File folders generally have names indicated on the tab. Similarly, files in all computers also have file names.

File Names

In talking with a number of microcomputer users, I have found that file names are especially puzzling, particularly to new users. Files are groups of information (analogous to the common office file folders as previously mentioned) that are stored on a disk.

Most microcomputer systems allow the use of file names which consist of two parts: the filename (a required combination of characters) and the file type (usually optional). I have found it useful to think of a file name (note the space between the two words) as being in the following format:

filename.typ

The entire entry — filename.typ — is the complete specification of the file name. Note that my use of "filename" (not separated by a space) indicates the first eight characters of the file name. MS-DOS allows a maximum of 8 alphanumeric (combination of letters and numbers) characters for the filename and 3 alphanumeric characters for the file type. Note that the file type — sometimes called an extension — is not a required part of the user assigned file name in MS-DOS.

There are some limitations in choosing a file name. Aside from the limit of 8 characters in the filename and 3 characters in the file type (note that a period MUST be used to separate the filename from the file type), the following characters cannot be used in filenames or types:

< > , ; : = ? * [ ] + | / \ space tab

Most of these characters are used in the command lines for MS-DOS (and CP/M); without these restrictions, the DOS would be unable to tell where the command name ends and a file name begins.

In addition to the special characters restriction, there are specific conventions that must be followed when choosing a file name:

- The file name must be from 1 to 8 alphanumeric characters.
- The file name must begin with an alphabetic character. Note that some operating system versions no longer require this convention, however, it is still recommended for compatibility reasons.
- The file type (extension) generally does not have to be specified for most files. Exceptions to this rule include files used for programming (e.g. BAS or ASM) and batch files (BAT).
- If specified, the file type can be from 1 to 3 alphanumeric characters — except for the reserved characters described above — and must be separated from the filename portion by a period (verbally referred to as "dot", i.e. "filename-dot-type".

All of the file names thus far discussed (in the form of filename .typ) are "unambiguous" because the file name has been completely specified. UNAMBIGUOUS FILE NAMES (abbreviated as ufn) are a completely specified and unique reference to a single file.

MS-DOS also provides the capability to use "ambiguous" file names. AMBIGUOUS FILE NAMES (abbreviated as afn) can refer to any number of different files using the "masking" capability of the wildcard characters ? (question mark) and * (asterisk).

The ? character is understood to mean "match any character at this location" and is, therefore, strictly dependent upon its exact location within the afn description. The * character will match all characters at the position of the asterisk plus any positions that follow to the right, up to the period separator or the end of the file name portion. And the wildcard specification *.* will match all files on the current or specified drive.

For example, let's say that we wanted to obtain a directory display (DIR command) on all BASIC program files on the current drive. The command DIR *.BAS will show all files with a BAS file type. If we change that slightly to DIR *.BA?, we will see BAS files, BAT (batch) files, and BAK (backup) files among others. From a technical perspective, it is interesting to note that MS-DOS "expands" the asterisk wildcard to question marks. For example, when you enter the command DIR *.BAS, the real directory search is based on DIR ????????.BAS where the asterisk is expanded to allow a search for all characters in each of the eight possible positions. All of these conventions for file names are the same as discussed in my FlipFast books. More on disk directories later.

Disks And Disk Directories

Now that we have looked at a file, it is time to consider a collection of files that are contained on a floppy disk or a hard disk partition. Many introductory texts discuss the concept that a disk is similar to a file cabinet. That is, it contains a collection of files. Since we know that each file has a unique file name, MS-DOS must have a way of finding the physical location of a file on the disk. This is accomplished by using the disk directory as a starting point. A Directory contains the file name and various other information related to the file such as date, time, and actual file length. As it turns out, each directory entry is 32 bytes (or characters) in length (that applies to both DOS and CP/M).

For our purposes, the most important information is contained in the first eleven bytes which contain the 8 byte filename and the 3 byte file type. The filename and type are left justified in the directory, and spaces are inserted in unused positions. Just as a side note, the term VTOC — Volume Table of Contents — is used in many mainframe systems to describe the disk directory which contains similar information. I think that VTOC is really more descriptive of the real contents and use of the file related information.

The Disk Directory

Aside from the file name, there are quite a few interesting things in the disk directory. An ATTRIBUTE BYTE — located at the 12th byte — defines the characteristics of the disk files. Most files are created with a "zero" attribute byte. In addition, that byte can be modified by the ATTRIB command in DOS 3.1 to allow for Read/Only (01) or Read/Write (00 - the normal attribute). Other attributes — hidden, system, label — are assigned by certain programs. FORMAT, for example, asks for a volume label which has the attribute of 08 in this position. And while I am discussing attributes, I will simply note in passing that a subdirectory has a 16 (10 hex) attribute.

The 32 byte disk directory entry exists for all defined files on the disk. This includes the BIOS (e.g. IO.SYS) and the system kernel (e.g. MSDOS.SYS), as well as COMMAND.COM and the volume label if one has been defined.

Among other things, the disk directory also contains the date that the file was created or updated, the time that the file was created or updated, a "pointer" to the File Allocation Table (FAT), and the actual size of the file. There is a common misconception that all forms of DOS are more efficient in disk space usage than CP/M. This appears to have happened because the DIR command shows the actual file size instead of the actual disk space used for the file.

As noted in my column in the July 1985 issue of REMark (Blocks and Clusters), both CP/M and DOS allocate file space in block units. For DOS, these block allocation units are technically called clusters. Double-sided, double-density DOS disks typically allo-

cate file space in 1k clusters. That means that the required disk space for any file smaller than 1024 bytes is still 1024 bytes, regardless of the actual file size. It just happens that DOS checks the file size and writes the actual value to the disk directory, and that actual value is displayed by the DIR command. So, that 33 byte AUTOEXEC.BAT file really takes 1024 (or 1K) of physical disk space. Similarly, that 2049 byte file really takes 3072 bytes (3 clusters) even though it is only one byte over 2048 bytes (2 clusters).

The final important item of information contained in the disk directory is the Starting Cluster of the file. The starting cluster is also a pointer into the File Allocation Table which is how DOS "locates" files that are larger than one cluster.

The File Allocation Table

Most FATs are constructed with 12 bit (1.5 byte) entries, and the first three bytes (two entries) of the FAT are reserved according to the Microsoft standard. The first byte is an "indicator" (called the FAT ID) of the disk characteristics, usually an FD (hex) for a DS/DD 9 sector disk or FF (hex) for a DS/DD 8 sector disk. By convention, the next two bytes are FF FF (hex). All of the remaining 1.5 byte FAT entries are for files which are initialized to 000 (hex) by the FORMAT program. As files are created, these numbers are updated to indicate the clusters for that file. Remember that the disk directory provides the starting cluster which also points to a FAT entry.

As a standard, FF7 (hex) is reserved to mark a bad cluster. The FF7 is created by the FORMAT program which is why you will lose an entire cluster (e.g. 1024 bytes) if a bad sector (i.e. 512 bytes) is found on the floppy disk. Although there are some other standard values used in the FAT, the other common value is FFF (hex) which signals the last cluster of the file.

For example, let's take the 33 byte AUTOEXEC.BAT file that I mentioned earlier. It has, say, a starting cluster of 0C5 (hex). If we looked at the FAT in the appropriate location, we would find an FFF (hex) to indicate that there are no more clusters for the file. This is the simple case. For larger files, additional cluster numbers (called a CHAIN) are recorded in the FAT, and again, the end of the file is indicated by FFF (hex). Although Microsoft also defines the values FF8–FFE (hex) as valid for indicating the last cluster in the file, it is my experience that these values are not used in practice.

The FAT is obviously a table which is used by DOS to keep track of the disk space. FAT entries may range from 000 to FFF (hex or 4095 decimal) for 1.5 byte entries. The first three bytes (two FAT entries — 000 and 001) of the FAT are reserved.

Following that, data clusters are numbered sequentially beginning with 2 to a number that is one greater than the number of clusters on the disk. For 8 sector, DS/DD disks, the number is 316 decimal. For 9 sector, DS/DD disks, the number is 355 decimal. If you really understand disk formats, you will be able to show that a standard 9 sector disk has 354 clusters available for use — that should be the same as the value reported by CHKDSK (354 clusters × 1024 bytes/cluster) or 362,496 bytes.

Regardless of the physical characteristics of the disk, FATs may contain valid cluster numbers ranging from 002–FF6 (hex). The remaining values — FF7 to FFF (hex) — are reserved. If you subtract the number of reserved values (000, 001, FF7, FF8, FF9, FFA, FFB, FFC, FFD, FFE,and FFF hex), you will find that there are a total of 4086 possible entries (002 to FF6 hex) in a FAT. And if you look at Table 3 for a hard disk in my July 1985 column, you will notice a distinct relationship between the Partition Size and the Minimum File Allocation Size. For example, the 8 MB partition size has a minimum file allocation size of 2048 bytes. If you take the maximum number of entries in the FAT (4086), and multiply it by the 2048 bytes, you will be able to show that you have enough entries to handle an 8 megabyte partition. You can also play that game to show that the same thing is true for larger partitions because the minimum file allocation size is correspondingly larger.

That clearly means that we have discovered a limitation — the maximum of 4086 entries for a FAT that uses 1.5 byte entries. As the hard disk partition sizes get larger, so do the minimum file allocation sizes. In the worst case, DOS would use 16,384 bytes for our 33 byte AUTOEXEC file in a 32–64 megabyte partition, not a very efficient use of disk space. But what happens in larger disks, like the 20 megabyte and larger sizes used by the Z-241 and Zenith compatibles (e.g. the IBM PC AT)?

The size of the FAT entry had to be expanded from 1.5 to 2 bytes. Some semblance of compatibility was maintained by simply adding an F (hex) to the maximum value so that the range includes 0000–FFFF (hex) or 0–65,535 decimal 2-byte FAT entries. If the standard floppy disk DS/DD cluster size (i.e. 2 sectors per cluster) was maintained, it is obvious that it would be easy to have a 64 megabyte partition with a 1K cluster size. Even if the standard Zenith 0–8 megabyte cluster size (2 kilobytes) was maintained, that still allows a 32 megabyte partition. By the way, I do not currently know what the standard cluster size is for the Z-241 and compatibles.

So much for the directories and FAT's. Let's see what happens when a file is erased. And then I will give you the secret as to how some programs can recover these erased files.

When A File Is Erased . . .

Two things happen to a file when you use the ERASE command. First, a special character (E5 hex) is placed in the first byte of the directory entry which replaces the first character of the file name. Then the file space is deallocated — a technical term which means that all of the entries in the FAT (called a chain) are set to zero except for the entry indicating the last cluster in the chain (the FFF hex entry). Let's take a look at the possibilities in this situation.

The first possibility is that the file is smaller than one cluster, typically 1024 bytes on a floppy disk. As usual, the starting cluster is found in the directory entry. The file is smaller than one cluster so we find an FFF hex in the appropriate FAT entry indicating the last cluster in the file. In this special case, all that needs to be done to "unerase" the file is to simply change the first character in the disk directory to a valid ASCII (printable) character. That is an easy task with DEBUG.

For files larger than one cluster, the task becomes more complex. Since the file chain in the FAT was set to zero, there is apparently no way to recover the file. The programming secret to recovering an erased file is that you must understand how DOS allocates (and deallocates) space for disk files.

How DOS Allocates File Space

DOS allocates file space as data is stored on a cluster by cluster basis. Since the user does not need to know or care how much file space is required, this technique is called dynamic allocation — dynamic in the sense that the DOS determines how much space is actually needed for a given file. In a mainframe system, specialists must calculate how much space a file requires (on disk

or tape), and specify that space requirement in Job Control Language (JCL). Failure to accurately judge the space requirements results in the "B37" or "E37" type of error which means that not enough file space was specified. The job abends (abnormal end) and must be rerun when this happens. Fortunately, we do not have to worry too much about that except when the disk (or directory) itself runs out of space.

One of the interesting points related to space allocation is that DOS uses clusters as they are available without regard as to whether all clusters are physically located (usually sequentially) together — in one CONTIGUOUS file. In a freshly formatted disk, all space obviously exists in one chunk. Files copied to that disk (using the COPY command) are cleanly arranged in contiguous (physically adjacent) clusters.

As files are updated and lengthened, DOS uses the first available cluster(s) for the additional space which are, most likely, scattered throughout the disk. If you use the CHKDSK *.* command on a disk which has been in service for some time, you will probably see the information message: "filename.typ Contains nn non-contiguous blocks" which means that the clusters are not adjacent to each other in that file. From a user perspective, it is important to know that since you may be able to improve the performance of your system by copying files on that disk to a new one. That improvement is due to the fact that the disk drive heads will not have to be moved to various locations on the disk — they can read the file with minimum movement.

That's another good reason to back up your files on a periodic basis. And although the DISKCOPY command is good for some backups, it is important to know that it copies disks on a sector-by-sector basis — even to the point of duplicating bad cluster marks (FF7 hex) in the FAT. The point is, use the COPY command to eliminate the noncontiguous files reported by CHKDSK.

The Secret Of Recovering Erased Files

As part of the file space allocation algorithm, DOS allocates space using what I call a "Last Erased, First Used" (LEFU) technique. That is, clusters which are freed (set to zero in the FAT) as a result of the ERASE command are the first ones used by DOS when additional file space is required. Obviously, DOS "remembers" the last clusters that were erased which is the programming secret to recovering erased files. Although the concept itself is simple, the programming required to accomplish that task is not trivial. The LEFU space allocation algorithm is the primary reason that most "recovery" program documentation states that files may not be recoverable if you have done ANYTHING to the disk (especially update another file) since the ERASE command was used.

CP/M uses a similar space allocation algorithm although the recovery process is much easier since there is no FAT (and corresponding zeroed FAT entries) to contend with. The CP/M disk directory entry actually contains a "disk map" which is similar in concept to the FAT. Erased CP/M files can be simply recovered because the ERA command only sets the first byte of the directory entry to E5 (hex).

Files, FATs, And Disk Directories — A Final Look

Now you know almost everything there is to know about disk directories, FATs, and files. A floppy disk (or hard disk partition) can be viewed as a file cabinet. A file is a single unit of related data within that file cabinet. Unlike a file cabinet, we have a special index — called a disk directory — to the contents of that disk. The disk directory lists the contents of the file cabinet (i.e. disk) by file name and provides a "pointer" to the initial location of the first chunk (cluster) of the file. Again, unlike a file cabinet, our files are stored in a limited space called a cluster. Large files are stored in multiple clusters (locations) which are recorded in the File Allocation Table (FAT).

Those of you who are technically inclined will note that I have purposely left out some of the intricate technical details, such as how to read the cluster number in the FAT or the date and time in the disk directory. This article was intended to be an introduction to the subject, not a discussion of high order bits in low order bytes and the process for changing that into a usable form. That assumes a knowledge of assembly language and other details that are way beyond the scope of what is presented here.

I have received more than a few letters asking where I managed to learn all of this kind of information. Some of you have been surprised to learn that much of this is available in the various manuals and other documentation provided with the Heath and Zenith systems. Although I have spent a considerable amount of time with DEBUG and some C programs to determine some of these details, the basics are contained in the system documentation.

On To Other Things

I had occasion to talk on the phone with a gentleman the other day who, with tongue firmly embedded in cheek, accused me of being partial to Heath and Zenith computers. It's true! And I make no excuses about it. In my opinion, the Heath and Zenith hardware and software is probably the best you can find anywhere at any price, period.

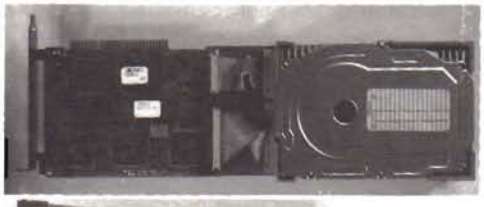
Furthermore, my experience is that support from Heath and Zenith is the best available. Not that it's perfect, it's just the best available. For example, I recently assembled an H-151 computer system for a friend. He bought the system at the Fort Worth Heathkit Store on my recommendation. We took the unit into the Dallas Heathkit Store to have the disk controller aligned since it became obvious that my low input resistance VOM was not up to the task. No problem! That was the day before Christmas, and we had the computer back in about an hour. As it happened, the technician was not very busy, so it was easy for Don Murray (the store manager) to provide that service. And Don came out on the deal too since he sold me an SM-77 digital voltmeter (DVM) which I bought to replace my VOM.

While I recognize that I may not be a typical user, I also had that kind of service BEFORE I wrote my first REMark column or published my first FlipFast book. Paul Fabares, at the Anaheim Heathkit Store, was responsible for that. The point is, I defy you to find that kind of service from IBM, Computerland or any other computer store at ANY time. My experience is that, in most cases, the other large chains are not very helpful in solving problems.

I also recognize that that kind of service is not always possible and you should too. But the fact that most Heathkit stores will make every effort to help you resolve any problem is something that is rare these days. That not withstanding, I have seen occasional stories about bad service from Heathkit stores, but I think that is the exception. And of course, it is just not possible to satisfy everyone all of the time, but I think that the Heath Company and the Heathkit stores make every effort to do so.

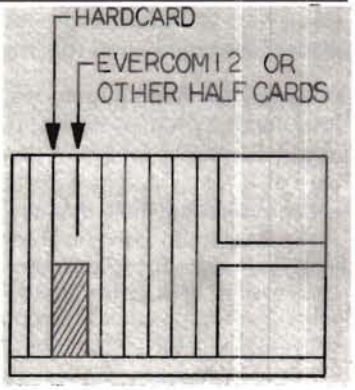
Lest you believe that I am blindly loyal to Heath and Zenith, there are some definite gripes that I have about each. While most of the prices in the Heathkit catalog provide a good value for the money, there are some rather spectacular exceptions. For example: the Z-205-1 64K RAM upgrade ($79.99), the Z-219-1 Video

The H/Z Upgrade Experts!



20M HARDCARD

Just drop the Hardcard in your H/Z PC's (including the portables), 20 million characters of storage space will be at your disposal. The hardcard is just like a conventional hardisk, except it does not require complicated installations.
The Hardcard is totally H/Z compatible, it supports all MFM-150 boot commands, just like the HS-317s.



HARDCARD $885.00

HARD DISK KITS

The hard disk kit comes with everything you need to upgrade your H/Z pc's (Expansion board is required for H/Z-138 and 148).
The Hard disk kit is totally compatible; unlike others, our hard disk kit is **THE ONLY** hard disk that supports the full MFM-150 boot commands.

20M — $545.00
30M — $789.00

60M BACKUP

The 60M tape backup is complete with a stand-alone tape drive, a controller, cables, an industrial standard tape cartridge, and software. It is FAST, RELIABLE, easy to install, simple to use, and extremely versatile. The software supports both image and file by file backup.

only $895.00

1200 BAUD MODEM

The Evercom 12 is fully Hayes and Bell 300/212A compatible. Features latest technologies for extremely reliable communication. The included Bitcom software provides VT100, 3101, and user defined terminal emulation. Bitcom supports text and Xmodem transfer, and automatic dial and logon.

only $179.00

EXPANSION SYSTEMS

Add expansion slots, drives and tape backup to your H/Z-138, 148, 150's and 160's. We have three versions available.

SLIMLINE	3½ slots
HALF-SIZE	4 slots
FULL-SIZE	6 slots

PLEASE CALL FOR PRICE!

8 MHZ UPGRADE for H/Z 150/160 Please Call

RAMDRIVE

command line Installation
From 2K to 700K+
Remove without reset.
Run faster than a hard disk.

$44.95

FAST HARD DISKS

For H/Z-200 by SEAGATE
40 MS AVE. access

20M — $696.00
30M — $855.00
40M — $945.00

WONG'S ADVANCED TECHNOLOGIES, INC.

ORDER TOLL FREE
(800) 626-1030

3201 Loyola Drive
Kenner, Louisiana 70065

INSIDE LOUISIANA
(504) 464-7379

We accept pre-paid, COD orders. Credit card purchases please call. Shipping charges minimum: $3.00 for the first 3 lbs. via UPS ground. Louisiana residents add 8% sales tax. No return merchandise will be accepted without an RMA number. Return merchandise are subject to a 15% restocking fee.
Prices are subject to change without notice.

RAM upgrade ($74.99), and the Z-205-4 256K RAM upgrade ($199.00). Not to mention the cost of hard disk upgrades for the various systems.

I can buy 64K RAM chips locally for less than a dollar a chip. The 256K RAM chips are available just about anywhere for about three dollars each or less. And 20 megabyte hard disks (e.g. HS-317-20 $1195.00) for the PC compatibles are available most places for about half of the listed price.

As a user, it is difficult for me to reconcile these prices no matter how much I would prefer to buy things through Heath. And as a user, you should be aware that, particularly in the case of the hard disk, it may well be worthwhile to pay a little more because of the service and the guarantee, but it's still a lot of money. As far as the RAM chips are concerned, I buy them outside of the normal Heathkit distribution channels. Paying over seven times the current market price for 256K RAM chips is not my idea of a good value. I can buy quite a few replacements in the event of a problem for the price difference even though they may not have the guarantee.

Since this seems to be the month for airing some personal complaints, I don't want to leave Zenith out. The Z-100 MS-DOS version of WordStar (MP-463-10) is probably the single worst implementation of any program on any computer. Although I can appreciate the technical complexities involved in porting WordStar from the PC version to the Z-100, that's no excuse in my opinion.

As a user, I want good performance from my software, and the Z-100 WordStar is just too SLLOOOWWWW, primarily in the video display. And I really don't care what the technical problems are. Comparatively speaking, the CP/M and PC versions of WordStar are blindingly fast, even in a floppy disk system. In fact, this implementation of WordStar is a distinct rival for Multimate (SS-5063-1) in terms of poor performance, except that Multimate seems to have bugs as I reported in my July 1985 column. About the only nice thing I can say about the Z-100's WordStar is that it is relatively bug free which is perhaps due more to the fact that the original had very few bugs than to the Z-100 modifications.

I used to think that it was my imagination about MS-DOS WordStar on the Z-100 until I started seeing all of the various patches that appeared in REMark last year. It's nice to know that I wasn't the only one who noticed the problem.

How To Shoot Yourself In The Foot

Without a doubt, Digital Research, Inc. (DRI) — developers of CP/M — should get an award for continuous self-inflicted injuries, otherwise known as the "Shooting Yourself in the Foot Award". This happens when you are out on Main Street, start to draw your gun, it sticks in the holster, and you manage to blow a rather large (and painful) hole in your foot. Ouch!

This started with the now famous story about Gary Kildall's failure to meet the IBM representatives when they appeared at DRI to discuss an operating system for the IBM Personal Computer. You may recall that it has been reported that he was out flying his personal airplane for pleasure. He continues to claim that it was a business trip, but regardless, DRI does not have the "standard" operating system today — Microsoft does. In this particular case, DRI not only managed to shoot a hole in its corporate foot, it actually blew the whole foot completely off. Chalk up one large and incredibly painful lost foot. Double Ouch!

While limping around on this badly injured stub of a leg, DRI managed to produce some good products. CP/M Plus was developed for the 8-bit micros, and CP/M-86 was provided for the 16-bit machines. By this time, DRI had managed to locate an artificial foot, and was learning to walk on it. In fact, DRI had learned to use the artificial foot quite well. Even to the point of developing a new operating system for the 16-bit machines called Concurrent CP/M or Concurrent DOS, whichever you prefer.

DRI was now back in contention for microcomputer operating systems, sort of... They then managed to shoot themselves in the foot again by discontinuing marketing and support of CP/M Plus which was a rather spectacular improvement over the original 8-bit CP/M. Unfortunately, this shot in the foot resulted in a fire to the artificial limb, becaue of their failure to recognize the incredible number of 8-bit systems being used. Due to the burns sustained, DRI had to have the leg amputated just below the knee. That really hurts! Triple Ouch!

DRI had to learn to walk on this artificial leg and developed the Graphics Environment Manager — GEM for short. GEM has had a lot of good reviews, and I personally think it is a very nice improvement over the original CP/M system interface. But, you guessed it — another shot in the foot! As it turns out, DRI was challenged on Main Street by Apple Computers for reportedly infringing on a copyright related to the icons used on Apple's Macintosh and Lisa computers. This shot in the foot was accomplished by DRI's agreement to change the GEM interface so that it would not "infringe" on Apple's copyright. The bad news is that this shot again caused a fire in the artificial limb that resulted in such severe burns that the rest of the leg had to be amputated. During the shootout, DRI also sustained critical head and facial injuries by falling flat on its corporate face. Rumor has it that part of the face was irrevocably lost in the shootout which happened late last year.

Today, DRI is still in the hospital, and status reports on its health tend to vary. The leg "injury" is nearly healed, but there was apparently significant injury to the facial tissue. Severe facial scars will probably be seen for some time. A completely new artificial leg is being developed for DRI, so we'll see if they can learn to walk on it without blowing it off again.

I think this latest fiasco with Apple is one of the most ridiculous things that I've ever heard. Apparently the issue centered on the use of the exact icons used by Apple. For example, a trash can is used to represent the "erase" command. How many ways can you represent something to be thrown out? Come on now, that's a bit much! Even worse, DRI apparently agreed to change the GEM interface because of that. And to add insult to injury, Apple did not even develop the icon in the first place. The basic idea was developed at the Palo Alto Research Center (PARC). Isn't it interesting that the people at PARC have been quiet? Although I'm no lawyer, it seems to me that this whole thing creates a rather disturbing precedent. Like I said, DRI is still in the hospital...

PC Emulators For The H/Z-100

As I write this at the end of December, the latest word on the UCI Easy PC is that I don't have one yet. And if you are considering an emulator for your '100, you might find it useful to wait until you get some opinions on both the Gemini and the Easy PC.

And Finally . . .

Many of you are taking the time to let me know that you like to see the information that I have been writing on general computer-related subjects. I appreciate that. And again, if you have suggestions on additional topics, please let me know. In most cases, I do not intend to discuss a topic at a level that requires a

significant amount of technical knowledge, primarily assembly language. Some of the concepts are complicated enough without adding another dimension that requires an understanding of assembly language. That is more appropriate to a book like my FlipFast series.

For those of you who will be attending the International HUG Conference in August, you might want to let me know if you have some special topic of particular interest. Indications are that I will be attending again this year, and if all goes well, I will have a couple of discussions scheduled. Any suggestions for discussion topics would be appreciated. Like last year, I intend to keep the subject on a "how-to" level to help you get the most out of your computer system.

The next topic in the "general information series" will include a discussion of the promised MS-DOS subdirectories which was part of my session at the HUG Conference last year. And by the way, you might want to read my article in the March-April issue of Sextant if you are interested in "How to Use the MS-DOS PROMPT Command".

If you have any questions about the information in this article, please let me know. As usual, you should enclose a stamped, self-addressed envelope if you would like a personal reply.

Products Discussed

Sextant (Magazine — 6 issues) $14.97/year
Sextant Publishing Company
716 E Street S.E.
Washington, DC 20003
(202) 544-0900

Digital Voltmeter (SM-77) $119.95
WordStar
Z-100 MS-DOS (MP-463-10) $299.00
Multimate
Z-150/IBM PC (SS-5063-1) $495.00
Heath/Zenith Computer Centers
Heath Company Parts Department
Hilltop Road
St. Joseph, MI 49085
(800) 253-7057 (Heath Catalog orders only)

*

1986 APR 1986

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

1986 MAY 1986

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

1986 JUN 1986

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

=== OKIDUMP === Z-100 GRAPHICS SCREEN DUMP

Dual purpose screen dump utility. One program, two functions, three outputs. Dump your Z-100 graphics screen to any OKIDATA Microline Series printer (includes 92,93,192,193). OKIDUMP will dump your graphics from a monochrome or color monitor. OKIDUMP can also do an inverse graphics dump. Or it will do a text only dump from your screen. Works like PSC using F-12 key; but lets you select the type of output, without changing the screen.

The graphics output is a dot by dot transfer. It is printed sideways on the OKIDATA and is full screen size.

$24.50 plus $1.50 postage and handling.
C & H Consultants, 284 Alexander Dr., Warner Robins, GA 31093

HUG Members only!

HUGE HUGGER DISCOUNTS now available on these selected products...exclusively for members only!

That's right. We're offering HUG members these close-out, last call and limited quantity specials from Heath Company.

The items listed are not necessarily representative of complete systems. Please call your nearest Heath/Zenith Computers & Electronics Center or Heath Company toll-free at 1-800-253-7057 for more information before ordering.

Watch for additional MEMBER ONLY discounts in upcoming issues of REMark®!

REMark Discount List

	Item	Last Adv'd Price	February Issue Price	HUG Member Price	Qty Left
H-19-3	H-19 to H-88 Mod Kit	220.00		199.95	23
H-88-1	Floppy Disk Accessory	150.00		99.95	73
H-88-5	Cassette I/O Card	25.00		24.95	80
H-88-9	Drive Mounting & Inst.	25.00		24.95	40
H-89A	W/O CRT, Ser I/O, Disk Cont.	-------		399.95	53
HCA-5-77	H-77 Dust Cover	14.00		9.95	3
HCA-5-89	H-89 Dust Cover	14.00		9.95	1
HCA-9	Print Stand For H-125	99.00	89.95	79.95	24
HDC-100	H/Z-100 Dust Cover	16.00		9.95	3
HDC-125	H/Z-125 Dust Cover	16.00		9.95	4
PA-100-1	USR Telpak (PC&ZDOS)	99.00		79.95	16
PC-121-50	TEC Captain 320K (PC)	399.00		349.95	6
PC-131	USR PCM-5 Modem (PC)	349.95		299.95	65
PC-140	DG Magic RAM Expn. 0K	129.00	99.95	49.95	9
PC-200	USR Modem Card (S100)	399.00		349.95	62
PC-310	DG Super "89 CPU	699.00	499.95	459.95	33
PCA-120-1	6-PAK Plus Game Acc'y.	39.95	34.95	29.95	20
PP-110	Trippe Emer. Power Sys.	349.00		299.95	1
PP-210	DP P132 Color Printer	1795.00		1199.00	1
PP-215	EPSON JX-80	699.95		499.95	49
PP-260	DTC 380Z Daisy Printer	1295.00	899.95	799.95	14
PP-400	Koalapad Touch Tablet	99.95		89.95	168

	Item	Last Adv'd Price	February Issue Price	HUG Member Price	Qty Left
PP-401	WICO 50-0110 Joystick	39.95		34.95	12
PPA-210-1	Black Ribbon	12.95		6.99	38
PPA-210-2	Primary Color Ribbon	24.95		12.99	91
PPA-210-3	Process Color Ribbon	24.95		12.99	69
PPA-215-1	Ribbon: Epson JX-80	15.95		12.95	36
PPA-260-2	Cut-Sheet Feeder/380Z	574.95	399.95	379.95	3
PPA-260-3	Ribbons (3Pk) /380Z	9.95	5.99	5.49	245
PPA-260-4	Cable - RS-232/380Z	49.95		34.95	28
PPA-260-5	Cable - Parallel/380Z	49.95		34.95	20
PPA-260-6	Wheel - Brougham 10	26.95		19.99	49
PPA-260-8	Wheel - Anelia Prop.	26.95		19.99	31
PPA-260-9	Wheel - Script 10/12	26.95		19.99	42
PPA-260-20	Wheel - Quadro 15	26.95		19.99	68
PPA-270-1	Trac Feed - Stylewriter	149.95	99.95	89.95	5
PPA-270-2	Sheet Feed - Stylewriter	249.95	174.95	149.95	44
PPA-270-4	Black Ribbons (3Pk)	16.95		9.99	283
PPA-270-5	Red Ribbons (2PK)	6.95		4.99	40
PPA-400-1	KOA Speedkey Software	59.95		49.95	40
ZA-100-4	S-100 Extender Board	59.00		29.95	346
ZA-219	Amber CRT /H-121	75.00		19.95	584
ZG-219	Green CRT /H-121	75.00		19.95	140
ZW-219	White CRT /H-121	75.00		14.95	152

Heathkit®
Heath Company

A subsidiary of Zenith Electronics Corporation.
Some items may not be available at all store locations.
Prices, product availability and specifications subject to change without notice.

CP-236

Put Whiz In Your Programming

A Review Of Software Wizardry's Utility Program For The Z-100

Joseph Katz
103 South Edisto Avenue
Columbia, SC 29205

Whiz is a memory-resident background utility for the Z-100. Like SideKick, the ancestor of such programs, Whiz waits patiently in RAM (Random Access Memory) while you run other programs. Then, when you need one of the Whiz functions, pressing the BREAK key and a number key makes it pop up. Press the ESC key when you're finished, and Whiz retreats into the background again. If you've seen SideKick or any of its other descendants, you have a general idea of Whiz.

But a quick comparison gives you only a general idea. Next to one of those flashy, multicolor beauties, Whiz seems pale and wan. It's monochromatic. It has no menu or alternate ways of executing functions. Nor does it have frivolities such as SideKick's telephone dialer, useless unless you have a Hayes Smartmodem and needless even if you do. Yet, it's Whiz I run before I do any programming on my Z-100.

The reason is that Whiz really shines as a programmer's tool on this wonderful programmer's microcomputer. Of course, it's useful as a general-purpose background program too, just as the Z-100 is marvelous as a general-purpose microcomputer. When you turn to writing or debugging programs in any language — C, BASIC, Pascal, assembler, or whatever — though, you'll begin to appreciate the real value of Whiz, just as you do the Z-100.

What's ironic is that SideKick is supposed to have started out as just such a toolkit written by Borland International's programmers for their own use. At some point, as it was being developed commercially, though, Borland seems to have gone into attract mode and added bells, whistles, and flash to catch the biggest possible market. In the process, the program's original use apparently was pushed back. Perhaps that's the explanation for that silly dialer. Of course, the ASCII table in SideKick must seem just as irrelevant to the non-programmer interested only in running a spreadsheet or accounting system.

So while most SideKick descendants are a potpourri of functions, Whiz is truly useful to programmers. Intentional or not, its avoidance of flashy color combinations, menu, and several different ways to invoke functions conserve RAM. What makes other background programs look so nice devours memory the way Pakman chomps food particles and critters when you're on a winning streak.

Keeping in mind, then, that my perspective on Whiz is as a programming tool, let's take it briefly by the numbers — those you press after hitting the BREAK key.

Press 0 after you press BREAK and you get the help screen, which lists four of the six Whiz functions: notepad and calculator; appointment calendar and calculator; alarm clock; ASCII table. The remaining two functions were added after the manual was printed, and are explained in an update file on the distribution disk.

Press BREAK 1 and you get a window on the combination notepad and calculator. The window shows 10 lines of 60 columns each, but the notepad will hold 40 lines of 80 columns each. You scroll through its contents using the arrow keys, just as you would expect. Unlike SideKick, which uses modified WordStar editing commands, Whiz uses the Z-100 editing keys for deleting and inserting lines or characters, backspacing, and homing the cursor, and F keys for things such as inserting the system date or time, writing the notepad to a file, reading from a file into the notepad, or reading into the notepad from a screen display produced by another program.

You can't capture graphics characters that way, only normal text. But it's a good way to save an inspiration that you've found inappropriate for the current program, but is just the right routine for something waiting in the wings.

It's also a good way to print that routine, although the manual doesn't say so. Remember that MS-DOS treats the printer as a file. So when you need to print a section of program quickly, you can use some of these Whiz features to set up the printer, as well as do the printout. Have the printer setup code in a disk file, read it into the notepad, then save it to a file named "PRN:". That will dump it to the printer. Then, you clear the notepad, capture the program section you want to print, and save it to "PRN:".

BREAK 2 gets you a window on the combination appointment calendar and calculator. You have only the window — 10 lines by 40 columns — for each date, but unlike other programs, Whiz lets you use the space any way you like: appointments, notes, and even sums due, payable, or earned. The editing keys and many of the F keys do the same as in the notepad, and additional F keys allow moving forward or backwards in the calendar by day or week. Press F6 and you'll be prompted for a specific date to display.

The calculator is the same for both the notepad and the appointment calendar. Despite the manual's statement that it's "a full floating point algebraic expression evaluator," it offers a fixed decimal position with six decimal places. The rest of the statement is accurate, however, and there is a wide range of algebraic functions. What I miss are functions to calculate things such as AND, OR, and XOR, and the ability to manipulate binary and hexadecimal radices.

BREAK and 3 brings the alarm clock. SideKick doesn't have this function, but it's useful if you're trying to get in some programming time between appointments. Turn on the alarm switch, set the time you want the alarm to go off, then get back to business. When the time comes, the alarm sounds no matter what you're doing — unless, of course, you've turned off the computer.

For the ASCII table, you press BREAK 4. It displays ASCII characters with their decimal (0-255) and hexadecimal (00-FF) codes. That's easier than thumbing through a printed ASCII chart. It's downright essential whenever you want to program even simple graphics displays, because the ASCII table shows you the graphics characters in whatever font you have in the current ALTCHAR .SYS. If you've ever tried working from those printed representations of graphics characters in the Z100 User's Manual, you'll know why Whiz's kind of ASCII table is essential.

The two functions not listed on the help screen are real prizes for anyone who has gotten into trouble while programming — which is everyone who programs. They let you bail out of trouble when another program — perhaps one you're writing — goes wrong.

Maybe the video display has gone crazy, leaving you no choice but to reboot the computer. Not anymore: with Whiz loaded before the trouble happened, just hit BREAK 9 and the console is reset to power-on configuration.

Or maybe what happened is the cursor got locked on line 25, in which case all further display scrolls maddeningly on that one line until you reboot. Not anymore: with Whiz loaded before the trouble happened, just hit BREAK 8 and the cursor jumps out to home position.

Those are two features unique to Whiz and uniquely valuable when programming on the Z-100. For Whiz is not simply a SideKick clone, but a background utility uniquely matched to this unique microcomputer — matched not only to its power, but also to its quirks under version 2.0 of MS-DOS, which is what Whiz requires. Say you have (as I do) a pair of 8" drives on your Z-100 and forget to turn them on before booting the computer (as I do). Then you're in a Whiz notepad and try to save its contents to a file on Drive D: Whiz intercepts the usual MS-DOS "Abort, Retry, Ignore" prompt, which of course would knock out Whiz, and handles the error in a reasonable way of its own.

Whiz has fairly ordinary requirements. It won't run on Z-DOS, but requires MS-DOS 2.0. It will run on a Z-100 with as little as 192KB RAM, but benefits from having 448KB or more. It will run on a floppy disk system, but having a hard disk means you never have to say you're sorry to run out of storage space. It cares not at all if you have a monochrome or color monitor.

You can get Whiz from Software Wizardry, 1106 First Capitol Drive, St. Charles, MO 63301, or by phoning toll free 1-800-TO BUY IT. The price is $49.95 and Whiz is protected only by the buyer's integrity. *

HUG Price List

The following HUG Price List contains a list of all products in the HUG Software Catalog. For a detailed abstract of these products, refer to the issue of REMark specified.

Part Number	Description of Product	Selling Price	Issue
HDOS HARDCOPY SOFTWARE			
885-1008	Volume I Documentation	9.00	
885-1013	Volume II Documentation	12.00	
885-1015	Volume III Documentation	9.00	
885-1037	Volume IV Documentation	12.00	8
885-1058	Volume V Documentation	12.00	
MISCELLANEOUS HDOS COLLECTIONS			
885-1032	Disk V H8/H9	18.00	8
885-1044-[37]	Disk VI H8/H89	18.00	
885-1064-[37]	Disk IX H8/H89 Disk	18.00	
885-1066-[37]	Disk X H8/H89	18.00	10
885-1069	Disk XIII Misc H8/H89	18.00	
GAMES			
HDOS			
885-1010	Adventure Disk H8/H89	10.00	4
885-1029-[37]	Disk II Games 1 H8/H89	18.00	8
885-1030-[37]	Disk III Games 2 H8/H89	18.00	8
885-1031	Disk IV MUSIC H8 Only	20.00	25
885-1067-[37]	Disk XI H8/H19/H89 Games	18.00	12
885-1068	Disk XII MBASIC Graphic Games	18.00	10
885-1088-[37]	Disk XVII MBASIC Graph. Games	20.00	14
885-1093-[37]	D&D H8/H89 Disk	20.00	16
885-1096-[37]	MBASIC Action Games H8/H89	20.00	18
885-1103	Sea Battle HDOS H19/H8/H89	20.00	20
885-1111-[37]	HDOS MBASIC Games H8/H89	20.00	23
885-1112-[37]	HDOS Graphic Games H8/H89	20.00	23
885-1113-[37]	HDOS Action Games H8/H89	20.00	23
885-1114	H8 Color Raiders & Goop	20.00	23
885-1124	HUGMAN & Movie Animation Pkg	20.00	41
885-1125	MAZEMADNESS	20.00	41
885-1130	Star Battle	20.00	45
885-1133-[37]	HDOS Games Collection I	20.00	59
885-8009-[37]	HDOS & CP/M Galactic Warrior	20.00	32
885-8022	HDOS SHAPES	16.00	45
885-8026	HDOS Space Drop	16.00	49
885-8032-[37]	HDOS Castle	20.00	59
CP/M			
885-1206-[37]	CP/M Games Disk	20.00	11
885-1209-[37]	CP/M MBASIC D&D	20.00	19
885-1211-[37]	CP/M Sea Battle	20.00	20
885-1220-[37]	CP/M Action Games	20.00	32
885-1222-[37]	CP/M Adventure	10.00	35
885-1227-[37]	CP/M Casino Games	20.00	38
885-1228-[37]	CP/M Fast Action Games	20.00	39
885-1236-[37]	CP/M Fun Disk I	20.00	55
885-1248-[37]	CP/M Fun Disk II	35.00	69
ZDOS			
885-3004-37	ZDOS ZBASIC Graphic Games	20.00	37
885-3009-37	ZDOS ZBASIC D&D	20.00	50
885-3011-37	ZDOS ZBASIC Games Disk	20.00	52
885-3017-37	ZDOS Contest Games Disk	25.00	58
UTILITIES			
HDOS			
885-1022-[37]	HUG Editor (ED) Disk H8/H89	20.00	20
885-1025	Runoff Disk H8/H89	35.00	
885-1060-[37]	Disk VII H8/H89	18.00	
885-1061	TMI Load H8 ONLY Disk	18.00	
885-1062-[37]	Disk VIII H8/H89 (2 Disks)	25.00	
885-1063	Floating Point Disk H8/H89	18.00	
885-1065	Fix Point Package H8/H89 Disk	18.00	10
885-1075	HDOS Support Package H8/H89	60.00	
885-1077	TXTCON/BASCON H8/H89	18.00	
885-1079-[37]	HDOS Page Editor	25.00	15
885-1080	EDITX H8/H19/H89 Disk	20.00	
885-1082	Programs for Printers H8/H89	20.00	
885-1083-[37]	Disk XVI Misc H8/H89	20.00	11
885-1089-[37]	Disk XVIII Misc H8/H89	20.00	20
885-1090-[37]	Disk XIX Utilities H8/H89	20.00	22
885-1092-[37]	Relocating Debug Tool H8/H89	30.00	14
885-1098	H8 Color Graphics ASM	20.00	19
885-1099	H8 Color Graphics Tiny PASCAL	20.00	19
885-1105	HDOS Device Drivers H8/H89	20.00	24
885-1116	HDOS Z80 Debugging Tool	20.00	27
885-1119-[37]	BHBASIC Support	20.00	29
885-1120-[37]	HDOS 'WHEW' Utilities	20.00	33
885-1121	HDOS Hard Sec Sup Pkg 2 Disks	30.00	37
885-1123	XMET Robot Cross Assembler	20.00	40
885-1126	HDOS Utilities by PS	20.00	42
885-1127-[37]	HDOS Soft Sector Support Pkg	30.00	45
885-1128-[37]	HDOS DISKVIEW	16.00	46
885-1129-[37]	HDOS CVT Color Video Terminal	20.00	46
885-8001	SE (Screen Editor)	25.00	28
885-8003	BHTOMB	25.00	28
885-8004	UDUMP	35.00	28
885-8006	HDOS SUBMIT	20.00	31
885-8007	EZITRANS	30.00	30
885-8015	HDOS TEXTSET Formatter	30.00	42
885-8017	HDOS Programmers Helper	16.00	42
885-8024	HDOS BHBASIC Utilities Disk	16.00	46
CP/M			
885-1210-[37]	CP/M ED (same as 885-1022)	20.00	20
885-1212-[37]	CP/M Utilities H8/H89	20.00	21
885-1213-[37]	CP/M Disk Utilities H8/H89	20.00	22
885-1217-[37]	HUG Disk Duplicator Utilities	20.00	26
885-1223-[37]	HRUN HDOS Emulator 3 Disks	40.00	37
885-1225-[37]	CP/M Disk Dump & Edit Utility	30.00	40
885-1226-[37]	CP/M Utilities by PS	20.00	40
885-1229-[37]	XMET Robot Cross Assembler	20.00	40
885-1230-[37]	CP/M Function Key Mapper	20.00	42
885-1231-[37]	Cross Ref Utilities for MBASIC	20.00	43
885-1232-[37]	CP/M Color Video Terminal	20.00	46
885-1235-37	CP/M COPYDOS	20.00	54
885-1237-[37]	CP/M Utilities	20.00	55
885-1245-37	CP/M-85 KEYMAP	20.00	63
885-1246-[37]	CP/M HUG File Manager & Utilities	20.00	64
885-1247-37	CP/M-85 HUG Bkgrd Print Spooler	20.00	67
885-5001-37	CP/M-86 KEYMAP	20.00	51
885-5002-37	CP/M-86 HUG Editor	20.00	52
885-5003-37	CP/M-86 Utilities by PS	20.00	54
885-5008-37	CP/M 8080 To 8088 Trans. & HFM	20.00	64
885-5009-37	CP/M-86 HUG Bkgrd Print Spool	20.00	66
885-8018-[37]	CP/M Fast Eddy & Big Eddy	20.00	43
885-8019-[37]	DOCUMAT and DOCULIST	20.00	43
885-8025-37	CP/M-85/86 Fast Eddy	20.00	49
ZDOS			
885-3005-37	ZDOS Etchdump	20.00	39
885-3007-37	ZDOS CP/EMulator	20.00	47
885-3008-37	ZDOS Utilities	20.00	47
885-3010-37	ZDOS Keymap	20.00	51
885-3022-37	ZDOS/MSDOS Useful Programs I	30.00	63
885-3023-37	ZDOS/MSDOS EZPLOT	20.00	63
885-3026-37	MSDOS SMALL C Compiler	25.00	65
885-3030-37	ZDOS/MSDOS Z-100 PC Emulator	40.00	68
885-3031-37	ZDOS/MSDOS Graphics	20.00	69
885-3034-37	ZDOS/MSDOS ZPC Support Pkg	10.00	72
885-8029-37	ZDOS Fast Eddy	20.00	53
885-8035-37	MSDOS DOCUMAT and DOCULIST	20.00	70
H/Z100 ZDOS/MSDOS - H/Z150 PC MSDOS			
885-3012-37§§	ZDOS HUG Editor	20.00	52
885-3014-37§§	ZDOS/MSDOS Utilities II	20.00	54
885-3016-37§	ZDOS/MSDOS Adventure	10.00	57
885-3020-37§	MSDOS HUG Menu System	20.00	62
885-3021-37§§	ZDOS/MSDOS Cardcat	20.00	63
885-3024-37§	ZDOS/MSDOS 8080 To 8088 Trans	20.00	64
885-3025-37§§	ZDOS/MSDOS Misc. Utilities	20.00	64
885-3029-37§§	ZDOS/MSDOS HUG Bg. Print Spool	20.00	66
885-3032-37§	MSDOS Halley's Comet Locater	20.00	70
885-3035-37§§	MSDOS SPELL5 & SPELL5F	20.00	72

§ All program files run on both

§§ Program files run partially on both

PC/IBM COMPATIBLE			
885-6001-37	MSDOS Keymapper	20.00	59
885-6002-37	CP/EMulator II & ZEMulator	20.00	59
885-6003-37	MSDOS EZPLOT	20.00	65
885-6004-37	MSDOS CheapCalc	20.00	67
885-6005-37	MSDOS Skyviews	20.00	67
885-6006-37	MSDOS Cardcat	20.00	69
885-6007-37	MSDOS DND (Dung. & Dragons)	20.00	70
885-8033-37	MSDOS Fast Edit	20.00	62
885-8037-37	MSDOS Grade	20.00	70
PROGRAMMING LANGUAGES			
HDOS			
885-1038-[37]	Wise on Disk H8/H89	18.00	
885-1042-[37]	PILOT on Disk H8/H89	19.00	
885-1059	FOCAL-8 H8/H89 Disk	25.00	13
885-1078-[37]	HDOS Z80 Assembler	25.00	21
885-1085	PILOT Documentation	9.00	
885-1086-[37]	Tiny HDOS PASCAL H8/H89	20.00	13
885-1094	HDOS Fig-Forth H8/H89	40.00	18
885-1132-[37]	HDOS Tiny BASIC Compiler	25.00	59
885-1134	HDOS SMALL-C Compiler	30.00	63
CP/M			
885-1208-[37]	CP/M Fig-Forth H8/H89 2 Disks	40.00	18
885-1215-[37]	CP/M BASIC-E	20.00	26
BUSINESS, FINANCE AND EDUCATION			
HDOS			
885-1047	Stocks H8/H89 Disk	18.00	
885-1048	Personal Account H8/H89 Disk	18.00	
885-1049	Income Tax Records H8/H89 Disk	18.00	
885-1055-[37]	MBASIC Inventory Disk H8/H89	30.00	
885-1056	MBASIC Mail List	30.00	
885-1070	Disk XIV Home Fin H8/H89	18.00	
885-1071-[37]	MBASIC SmBusPk H8/H19/H89	75.00	17
885-1091-[37]	Grade/Score Keeping H8/H89	30.00	14
885-1097-[37]	MBASIC Quiz Disk H8/H89	20.00	18
885-1118-[37]	MBASIC Payroll	60.00	30
885-1131-[37]	HDOS CheapCalc	20.00	47
885-8010	HDOS Checkoff	25.00	32
885-8021	HDOS Student's Statistics Pkg	20.00	44
885-8027	HDOS SciCalc	20.00	50
CP/M			
885-1218-[37]	CP/M MBASIC Payroll	60.00	31
885-1233-[37]	CP/M CheapCalc	20.00	47
885-1239-[37]	Spread Sht. Contest Disk I	20.00	
885-1240-[37]	Spread Sht. Contest Disk II	20.00	
885-1241-[37]	Spread Sht. Contest Disk III	20.00	
885-1242-[37]	Spread Sht. Contest Disk IV	20.00	
885-1243-[37]	Spread Sht. Contest Disk V	20.00	
885-1244-[37]	Spread Sht. Contest Disk VI	20.00	
885-8011-[37]	CP/M Checkoff	25.00	32
885-8036-[37]	CP/M Grade	20.00	70
ZDOS			
885-3006-37	ZDOS CheapCalc	20.00	47
885-3013-37	ZDOS Checkbook Manager	20.00	54
885-3018-37	ZDOS Contest Spreadsheet Disk	25.00	58
885-8028-37	ZDOS SciCalc	20.00	50
885-8030-37	ZDOS Mathflash	20.00	55

Continued on Page 83

HUG NEW PRODUCTS

HUG P/N 885-8039-37 MS-DOS
DPATH $20.00

Introduction: DPATH is an MS-DOS utility which provides a data directory path search facility. Once loaded into memory, DPATH remains resident, and provides directory searching for data and overlay files in much the same way that the MS-DOS 'PATH' command causes searches for executable files.

Requirements: DPATH was designed for use on a Z-100 computer running MS-DOS version 2, but can be easily modified for use on any machine running MS-DOS version 2 or MS-DOS version 3. A version is supplied for use on Z-100 PC systems, and any other PC, XT or AT compatible.

The following files are included on this distribution disk:

DPATH	.COM	Z-100 version of DPATH
DPATHPC	.COM	Z-100 PC version of DPATH
DPATH	.DOC	Documentation file
DPATH	.ASM	Assembler source for DPATH
STRUCT	.ASM	Structure macros for DPATH.ASM
README	.DOC	Startup info

Program Author: Gordon Buchanan

DPATH.COM — An MS-DOS utility program that can help you to organize data and programs within the MS-DOS hierarchial file structure. DPATH searches for data files and program overlays in much the same way that the MS-DOS PATH command searches for executable command files. Once loaded into memory, DPATH becomes part of DOS and provides a directory search facility that is available to all subsequently executed programs.

When DPATH is used in conjunction with the PATH command, all programs, program overlays, device drivers, configuration files, global databases, help files, etc., can be stored in one or more user defined "system" directories, and removed completely from all application directories. The files in your system directories will be accessible from anywhere in the hierarchial file structure. The following benefits are thus gained.

- Better organized file structure, resulting in fewer files in application directories, and making it easier to find information on a disk.
- More free disk space because copies of programs and associated files are not needed in each application directory.

DPATH.DOC — Documentation for DPATH suitable for output on a printer. Includes information on the internal operations of DPATH and tells how to modify the program for use on different MS-DOS computers.

DPATH.ASM — Assembler source code for DPATH, using the MS-DOS assembler. Well documented, and written in a structured manner.

TABLE C
Product Rating

10 - Very Good
9 - Good
8 - Average

Rating values 8-10 are based on the ease of use, the programming technique used, and the efficiency of the product.

7 - Has hardware limitations (memory, disk storage, etc.)
6 - Requires special programming technique
5 - Requires additional or special hardware
4 - Requires a printer
3 - Uses the Special Function Keys (f1.f2.f3.etc.)
2 - Program runs in *Real Time
1 - Single-keystroke input
0 - Uses the H19 (H/Z89) escape codes (graphics, reverse video)

Real Time — a program that does not require interactivity with the user This term usually refers to games that continue to execute with or without the input of the player, e.g. p/n 885-1103 or 885-1211[-37] SEA BATTLE.

ORDERING INFORMATION

For Visa and MasterCard phone orders; telephone Heath Company Parts Department at (616) 982-3571 Have the part number(s), descriptions, and quantity ready for quick processing. By mail; send order, plus 10% postage and handling ($1.00 minimum charge, up to a maximum of $5.00. UPS is $1.75 minimum -- no maximum on UPS. UPS Blue Label is $4.00 minimum.), to Heath Company Parts Department, Hilltop Road, St. Joseph, MI 49085. Visa and MasterCard require minimum $10.00 order.

Any questions or problems regarding HUG software or REMark magazine should be directed to HUG at (616) 982-3463. REMEMBER-Heath Company Parts Department is NOT capable of answering questions regarding software or REMark.

NOTE

The [-37] means the product is available in hard-sector or soft-sector. Remember, when ordering the soft-sectored format, you must include the "-37" after the part number; e.g. 885-1223-37.

STRUCT.ASM — A set of macros which are required to assemble DPATH.ASM. These macros provide a set of coding structures for the MS-DOS assembler programmer. Includes WHILE-ENDW, REPEAT-UNTIL and IF-ELSE-ENDIF, which are completely nestable.

Comments: This program provides a facility which should have been built into DOS. A must for the hard disk user.

TABLE C Rating: (10)

HUG P/N 885-8040-37 MS-DOS HELP $20.00

Introduction: This package provides a comprehensive interactive HELP facility for Zenith Data Systems' implementation of both Version 2 and 3 of the Microsoft Disk Operating System (MS-DOS) for Zenith Z-100 series personal computers and IBM-PC compatible (Z-100 PC) personal computers.

The HELP programs and database files in this package allow quick and efficient access to complete MS-DOS command information which eliminates the need to have several manuals at the computer when command reference information is needed. The database files include all of the standard Zenith MS-DOS commands as documented in the Zenith MS-DOS manuals for both the Z-100 and Z-100 PC series computers, as well as the Zenith MS-DOS Programmer's Utility Package.

The database files are user extensible. The HELP programs and format of the database files were designed to be easily updated using a standard text editor. You can even create new HELP database files for your favorite word processor, modem communications program or programming language. Complete BASIC language source code is provided so that you can modify or enhance the programs as desired.

Requirements: An H/Z-100 series or H/Z-100 PC series personal computer and the MS-DOS operating system (either Version 2 or 3) are required. About 120k of disk space is required for the HELP program and database file, although the database file can be edited to reduce its size, if desired. The HELP programs require 128k bytes of memory. The HELP database can be modified using any standard text editor and the CNVT utility program provided. To modify the HELP programs, you must have either the Z-100 Z-BASIC compiler, or an IBM-PC compatible BASIC compiler.

The following files are included on this distribution disk:

README	.DOC	— Documentation file for the HELP program
HELP100	.BAS	— BASIC source code for Z-100 HELP program
HELP100	.EXE	— executable version of Z-100 HELP program
HELPPC	.BAS	— BASIC source code for the Z-100 PC HELP program
HELPPC	.EXE	— executable version of Z-100 PC HELP program
CNVT	.BAS	— BASIC source code for data base conversion program
CNVT100	.EXE	— executable version of Z-100 conversion program
CNVTPC	.EXE	— executable version of Z-100 PC conversion program
HELPS	.DAT	— sequential HELP data base file
HELP	.DAT	— random access HELP data base file
SD100	.COM	— Z-100 sorted directory utility program
SDPC	.COM	— Z-100 PC sorted directory utility program
SD	.DOC	— documentation for sorted directory program

The sorted directory utility programs are provided strictly as a convenience to the user and are not required to use the HELP programs.

Program Author: John F. Stetson

Program Content: The HELP programs are designed to be easy to use and efficient in operation. The HELP database is a BASIC random-access file for high speed access, even on floppy based systems. Over 100 entries are presented using a full-screen HELP menu and may be easily selected using the keyboard arrow keys. The commands in the HELP database are divided into the following functional categories:

Reference Commands — General reference information for MS-DOS concepts and capabilities.

Resident Commands — Information for commands which are resident in memory.

Resident Command Aliases — Information for command aliases for the resident commands.

Resident Batch Processing Commands — Information for commands which are used in batch processing.

Transient Commands — Information for transient commands which are loaded and executed from disk.

Transient Utilities — Information for commands which are part of the Zenith MS-DOS Programmer's Utility Pack.

Once a command is selected, the following information is displayed:

Command — General information about the command: alias names; Z-100 vs. Z-100 PC; MS-DOS V2 vs. MS-DOS V3; etc.

Function — Brief description of the function or purpose of the command.

Syntax — Complete, detailed command line syntax of the command including all file names, option switches, etc.

Examples — One or more examples which illustrate typical uses of the command.

Comments: As the MS-DOS operating system has evolved, it has become more complex. In addition, most end-users are overwhelmed by the amount of documentation which accompanies the operating system. This package organizes this information and makes it available to end-users in a friendly and easily accessible environment. In addition, the package is both comprehensive and user-extensible, which make it valuable for more sophisticated users.

TABLE C Rating: (10)

Continued on Page 83 ☛

MEDIA MASTER

FORMATS SUPPORTED	IBM PC	ZENITH Z-100	OSBORNE 1/EXEC	KAYPRO 2/4	KAYPRO 2X/4/10	DEC RAINBOW	SANYO 55/65
Actrix (DSDD)	•	•			•		•
Actrix (SSDD)	•	•	•	•	•	•	•
Avatar TC10 (DSDD)	•	•			•		•
Casio FP 1000 (DSDD)	•	•			•		•
Chameleon CP/M-80	•	•			•		•
Columbia MPC CP/M-80	•	•			•		•
Cromemco CDOS (SSDD)	•	•	•	•	•	•	•
Cromemco CDOS (SSSD)			•	•	•	•	
Cromemco w/Intl Term (DSDD)	•	•			•		•
Cromemco w/Intl Term (SSDD)	•	•	•	•	•	•	•
DEC Rainbow CP/M-86/80						•	
DEC Rainbow MS-DOS						•	
DEC Mate II CP/M (SSDD)						•	
DEC VT180 (SSDD)	•	•	•	•	•	•	•
Davidge (DSDD)	•	•			•		•
Digilog (DSDD)	•	•			•		•
Epson Multifont (DSDD)	•	•			•		•
Epson QX-10 (DSDD)	•	•			•		•
Facit DTC (SSDD)						•	
Fujitsu Micro 16S (DSDD)	•	•			•		•
Groupé III CP/M (DSDD)	•	•			•		•
Heath w/Magnolia CP/M	•	•	•	•	•	•	•
Heath/Zenith Z-DOS (1.xx DSDD)	•	•		•		•	•
Heath/Zenith Z-DOS (1.xx SSDD)	•	•	•	•	•	•	•
Heath/Zenith Z-DOS (2.xx DSDD)	•	•		•		•	•
Heath/Zenith Z-DOS (2.xx SSDD)	•	•	•	•	•	•	•
Heath/Zenith Z-100 CP/M (DSDD)	•	•					•
Heath/Zenith Z-100 CP/M (SSDD)	•	•	•	•	•	•	•
Heath/Zenith Z90-40 track (1k blocks)	•	•	•	•	•	•	•
Heath/Zenith Z90-40 track (2k blocks)	•	•	•		•	•	•
IBM PC CP/M-86 (DSDD)	•	•					•
IBM PC CP/M-86 (SSDD)	•	•	•	•	•	•	•
IBM PC DOS (1.xx DSDD)	•	•		•		•	•
IBM PC DOS (1.xx SSDD)	•	•	•	•	•	•	•
IBM PC DOS (2.xx DSDD)	•	•		•		•	•
IBM PC DOS (2.xx SSDD)	•	•	•	•	•	•	•
IDEA Ritelex (SSDD)	•	•	•		•	•	•
ISM CP/M (DSDD)	•	•			•		•
Ithaca (SSDD)						•	
Insight Dev 3Q 120 (SSDD)	•	•	•		•	•	•
Kaypro 10 (DSDD)	•	•			•		•
Kaypro 2 (SSDD)	•	•	•	•	•	•	•
Kaypro 4 (DSDD) *	•	•			•		•
LNW-80 (SSDD)	•	•	•	•	•	•	•
Lobo MAX-80 (SSDD)	•	•	•	•	•	•	•
Lobo MAX-80 512 (SSDD)	•	•	•		•	•	•
Micral 90/30 CP/M-80 (DSDD)	•	•			•		•
Morrow MD11 (DSDD)	•	•					•
Morrow MD2 (SSDD)	•	•	•	•	•	•	•
Morrow MD3 (DSDD)	•	•					•
Mai Sum 150 (SSDD)						•	
Monroe OC8820 (SSDD)						•	
NCR Decision Mate 5 (DSDD)	•	•			•		•
NEC PC-8001A (DSDD)	•	•			•		•
NEC PC-8001A (SSDD)	•	•	•	•	•	•	•
Olivetti ETX II (SSDD)	•	•	•		•	•	•
Olivetti EX100 (DSDD)	•	•			•		•
Osborne 1 (SSDD)	•	•	•	•	•	•	•
Osborne 1 (SSSD)			•	•	•	•	
Osborne 1 Vixen (DSDD)	•	•		•		•	•
Osborne Executive (SSDD)	•	•	•	•	•	•	•
Osborne Omnipac (SSDD)	•	•	•		•	•	•
Otrona (DSDD)	•	•			•		•
Phillips 3000 (SSDD)						•	
PMC Micromate (DSDD)	•	•			•		•
Reynolds & Reynolds (SSDD)	•	•	•		•	•	•
Sanyo CP/M (DSDD)	•	•			•		•
Superbrain (DSDD)	•	•			•		•
Superbrain Jr. (SSDD)	•	•	•		•	•	•
Systel II (SSDD)	•	•	•	•	•	•	•
Systel III (DSDD)	•	•			•		•
S-100 10 TRK (SSDD)			•		•	•	•
TI Professional CP/M-86 (DSDD)	•	•			•		•
TI Professional CP/M-86 (SSDD)	•	•	•	•	•	•	•
TRS-80 Model III FEC CP/M (SSDD)	•	•	•		•	•	•
TRS-80 Model III FEC D80 (SSDD)	•	•	•		•	•	•
TRS-80 Model III Burr-Labs (SSDD)	•	•	•		•	•	•
TRS-80 Model III Mem-Merch (SSDD)	•	•	•	•	•	•	•
TRS-80 Model I Omikron CP/M (SSSD)			•	•	•		
TRS-80 Model IV CP/M Plus (SSDD)	•	•	•	•	•	•	•
TRS-80 Model IV Montezuma Micro (SSDD)	•	•	•		•	•	•
Teletek 10 track (SSDD)	•	•	•		•	•	•
Teletek 80 track (SSDD)						•	
Toshiba T100 (DSDD)	•	•			•		•
Televideo TurboDOS (DSDD)	•	•			•		•
Televideo 802 CP/M					•		•
Wang Mawx CP/M (DSDD)	•	•			•		•
Xerox 820 II (SSDD)	•	•	•	•	•	•	•
Xerox 820 II (SSSD)			•	•	•		
Zorba 10 track (DSDD) *	•	•			•		•

** Requires access to "foreign" computer*

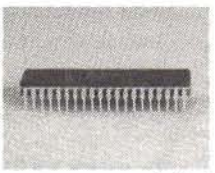
"MAKE YOUR Z-150 CP/M COMPATIBLE"

Intersecting Concepts Announces 3 Solutions To Solve Your Computer Incompatibility

But will it work on my computer? Yes! Finally, there are three easy ways to exchange information, transfer files, and run CP/M software on MS-DOS machines.

1. MEDIA MASTER is our direct disk-to-disk format conversion program. Already an accepted industry standard, this $39.95 program uses simple screen prompts that lets you read, write and format up to 70 different 5¼" diskettes from CP/M, MS-DOS and PC-DOS operating systems. So if you work on a IBM PCompatible at the office, but use a CP/M computer at home, now you can easily transfer files that would otherwise be "foreign" to your computer's operating system.

2. MEDIA MASTER PLUS goes one step further by converting 8-bit CP/M software to run on 16-bit MS-DOS and PC-DOS machines. This newly released $59.95 product combines our IBM-PC version of Media Master with ZP/EM, a powerful new emulation program. The results are amazing: CP/M programs using 8080 instructions *and* data can be transfered from popular computers like Osborne, Kaypro and Zenith to run on MS-DOS and PC-DOS machines!



3. ACCELER-8/16 is also new and dramatically improves the performance of Media Master Plus by tailoring the CP/M emulation around a NEC V20 microchip. This chip simply replaces the 8088 processor in your MS-DOS computer. Once installed, it'll run your CP/M and MS-DOS software much faster. (Speed improvements are roughly 15% faster in MS-DOS and 350% faster in CP/M!) With ACCELER-8/16 your MS-DOS computer is now CP/M compatible for only $99.95!

The Choice Is Yours

So if you just need to exchange files between different computers using different operating systems, there's Media Master at $39.95.

If you're upgrading to an MS-DOS or PC-DOS machine or already have one but don't want to spend money buying "new" software programs you already own, there's Media Master Plus for $59.95.

And if the execution speed of your CP/M programs is important, there's ACCELER-8/16 for $99.95, which provides a simple hardware and software solution to making your MS-DOS computer CP/M compatible. All three solutions save you money by eliminating expensive modems and communications software.

To Order

To order Media Master, Media Master Plus, or ACCELER-8/16, call 800-824-7888, and ask for operator 251. For additional product and upgrade information contact:

Intersecting Concepts, Inc.,
4573 Heatherglen Ct., Moorpark, CA 93021
or call 805-529-5073.

INTERSECTING CONCEPTS

Patch Page

Pat Swayne
HUG Software Engineer

This article contains patches for Dbase II version 2.41 to allow it to run properly under certain HUG background utilities, such as the Hug Background Print Spooler or FASTIO. It also contains a patch to fix a "bug" in the ANSI device driver for Z-100 PC series computers.

Dbase II Patch

If you run Dbase II with certain HUG background utilities (resident programs) loaded, you may get a message "Memory allocation error." "Cannot load COMMAND" when you attempt to QUIT Dbase operations. The problem occurs because Dbase II is a translated 8-bit program, and it uses a CP/M-like method for accessing the disk operating system that does not work properly when the HUG background utilities are loaded. Dbase II can be patched to access the DOS differently with DEBUG as shown below:

```
A>DEBUG B:DBASE.COM
-E352
1234:0352  E9.88  B0.CC  FC.CD  00.21  00.C3
-W
Writing 1234 bytes
-Q
```

This example assumes that you are logged on to drive A:, DEBUG.COM is on drive A:, and DBASE.COM is on drive B:.

ANSI Patch

The ANSI device driver supplied with MS-DOS for the Z-100 PC series of computers has the ability to map the computer's keys to produce user defined character sequences. Unfortunately, the driver has a "bug" that causes it to send an additional spurious character following the defined characters each time a mapped key is pressed. You can correct the problem using DEBUG as shown below:

```
A>DEBUG ANSI.SYS
-U402,403
1234:0402 BAAB07        MOV     DX,07AB
```

The above disassembly is done to check that your version of ANSI.SYS is the correct one for this patch. If your ANSI.SYS does not contain the instruction "MOV DX,07AB" at address 402, do not make the patch. Otherwise, continue with DEBUG as shown below:

```
-A402
1234:0402 MOV DX,7B1
1234:0405                (hit RETURN)
-A7DE
1234:07DE CALL 8AB
1234:07E1 NOP
1234:07E2
-A8AB
1234:08AB DEC CX
1234:08AC MOV [3B],DI
1234:08B0 RET
1234:08B1
-RCX
CX 07AB
:7B1
-W
Writing 07B1 bytes
-Q
```

This example assumes that both DEBUG and ANSI.SYS are on your currently logged disk.

If you have the Programmer's Utility Pack, the source code for the ANSI driver is contained in a file called Z150ANSI.ASM on the Z-100 PC BIOS Sources disk. If you wish, this file can be patched to correct the key mapping problem. First, load Z150ANSI.ASM into an editor, and locate the label CK4:. Change the next few lines below this label so that they look like this (Add the line with the comment "REMOVE EXTRA COUNT".):

```
CK4
        INC     DI                  , Update pointers
        DEC     CX
        DEC     CX                  ; REMOVE EXTRA COUNT
        MOV     STRING_PTR,DI       , Set up pointer to s
        MOV     STRING_COUNT.CL     , Count of chars  in
```

Now you can reassemble the corrected file by copying it to a disk along with Z150ROM.DEF, Z150BIOS.DEF, DRIVERS.DEF, and ASCII.DEF from the Z-100 PC BIOS Sources disk; MASM.EXE and LINK.EXE from the Utilities disk; and EXE2BIN.EXE from your MS-DOS Distribution disks. Log on to the disk containing these files, and enter these commands:

```
A>MASM Z150ANSI,
A>LINK Z150ANSI;
A>DEL Z150ANSI.OBJ
A>EXE2BIN Z150ANSI ANSI.SYS
A>DEL Z150ANSI.EXE
```

When you have completed these operations, you will have a new ANSI.SYS that you can copy to your MS-DOS system disk to replace the old one, instead of patching it with DEBUG. ✱

What Dreams Are Made Of!

Jim Buszkiewicz
HUG Software Developer

(HS–241 Hardware Review)

The HS–241 is Heath's kit version of the Zenith ZF–241 advanced personal computer. This computer is based on Intel's 80286 microprocessor and is fully IBM PC–AT compatible. It comes standard with a single parallel and RS–232 serial port. Also included is 512kb of RAM which can be optionally increased to 15mb! The 10 slot backplane circuit board should allow enough expansion room for even the most demanding computer user. This kit version presently sells for $2899, and takes about 2 hours to assemble. If you're not inclined toward kit building, the same wired version is available for $3999.

Due to the complex and sensitive nature of the electronics, each circuit board comes preassembled and tested. All that is required for construction is a screwdriver and pair of pliers. This kit requires NO soldering.

The first phase of construction includes the installation of the power supply, circuit board guides, and backplane circuit board. The system key–lock assembly is also installed at this time.

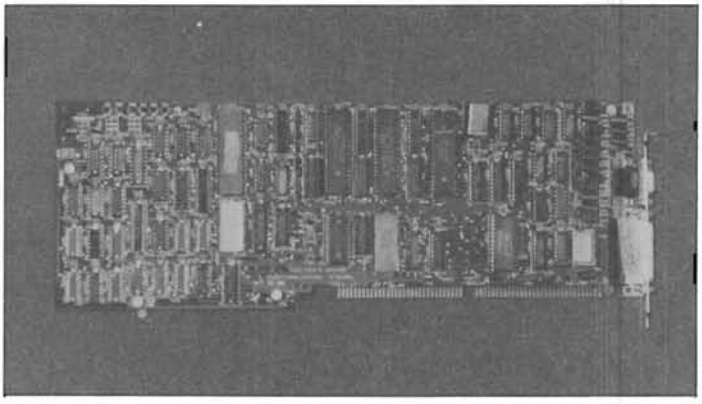
Next comes drive installation. Any combination of two 5" and two Winchesters can be installed. Each floppy/Winchester combination is covered in the assembly manual. One 1.2MB, 5" floppy drive comes standard with the system. I chose to install that drive, a standard 48tpi 360kb floppy drive, and two Winchesters, a Miniscribe 10MB, and a Control Data 40MB unit. Each "floppy–Winchester" combination is mounted in a disk drive chassis, and then each chassis is mounted inside the computer.

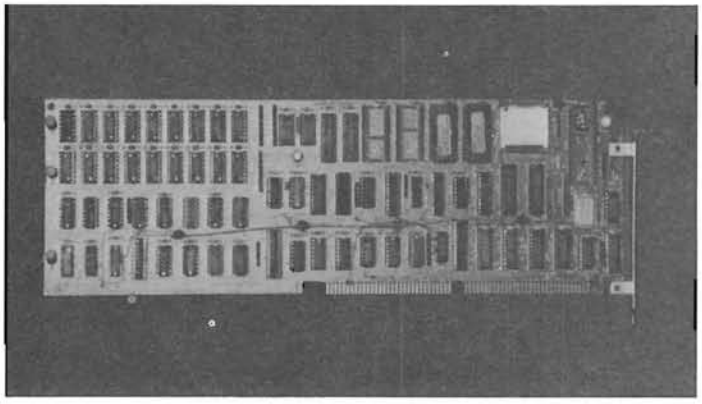
Once the drives are configured and installed, the individual circuit cards are then installed in the backplane. First the disk controller, the video board, CPU board, and finally I/O board. If you purchase an additional memory board, it is installed last. It should also be mentioned that a video board DOES NOT come standard with the computer. Heath has left it up to the user as to which video board to use including those from other vendors. If a different vendor's board is chosen, a circuit board sizing adaptor is provided since the Zenith video board is longer than normal.

After all the circuit boards are in place, A lithium battery (CMOS configuration RAM and real–time–clock backup power) is installed in the backplane board, and main AC power is applied. Several LEDs are then monitored on the backplane and I/O circuit boards for proper indication. If a problem does occur, these LEDs will help pinpoint its location. If everything has gone well to this point, the manual will take you into the MFM-200 monitor and through all of the monitor diagnostics. If those tests pass successfully, the next step is to perform a set of disk diagnostics. Yes, a complete set of disk–based diagnostics is even included with the system!

HS-241 Keyboard
PC users might note the more functional key layout changes.

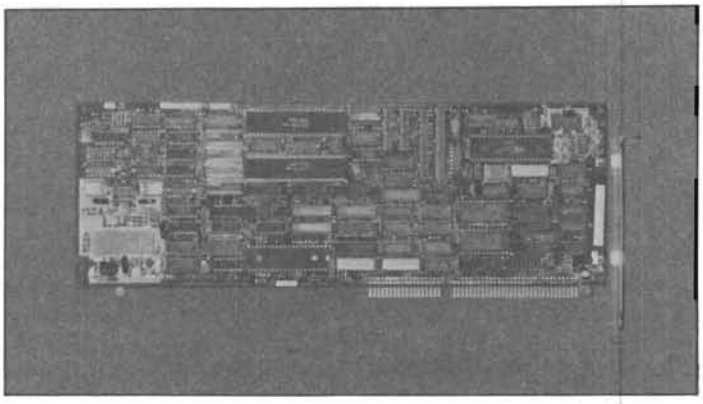
I/O Board
Note the non-standard 9-pin RS-232 connector. Perhaps a change for the better, since most of the pins on the standard 25 pin connector go to waste anyway.

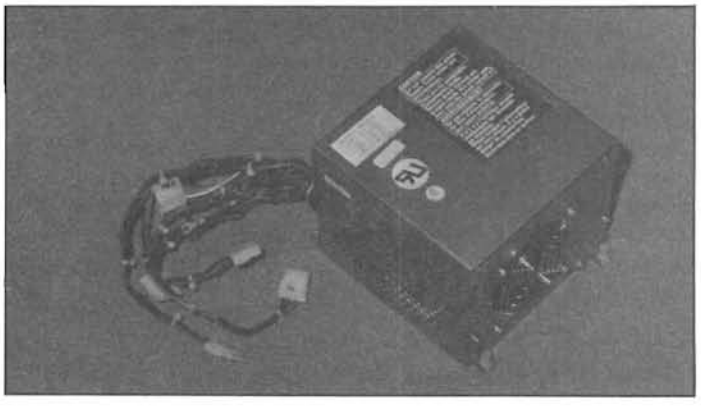




CPU Board
The processor chip is the square shaped device in the upper right-hand corner.
This guy gets really warm!

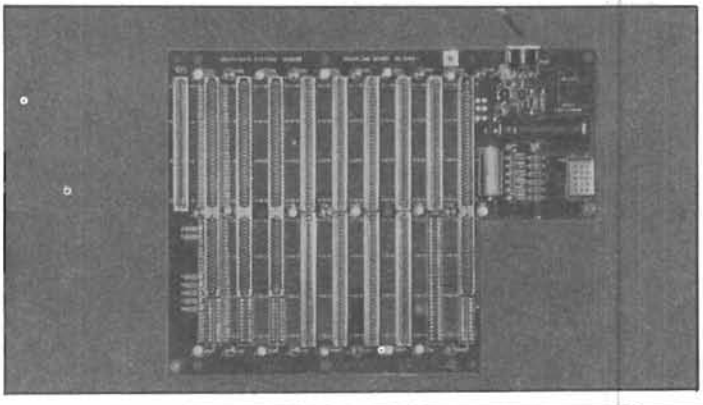
Disk Controller Board
This board is capable of controlling two floppies and three Winchesters.

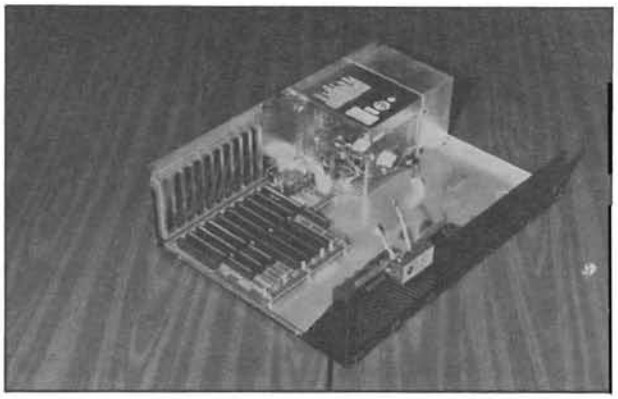




240 Watt Power Supply Module

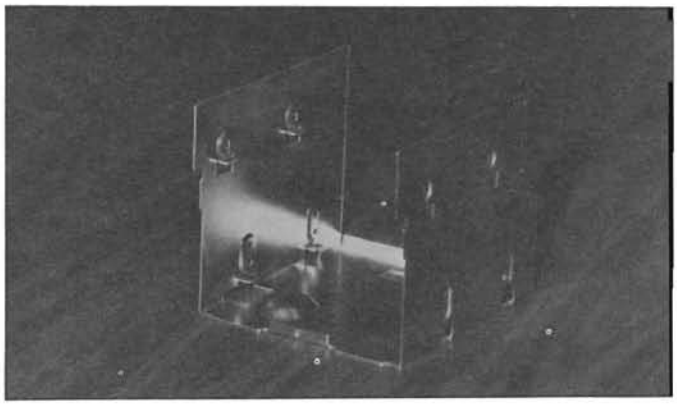
HS-241 Backplane Circuit Board
Note the lithium battery holder and keyboard connector in the upper right-hand corner.

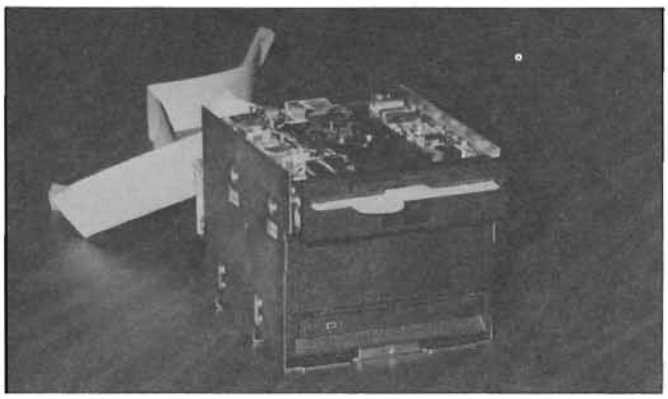




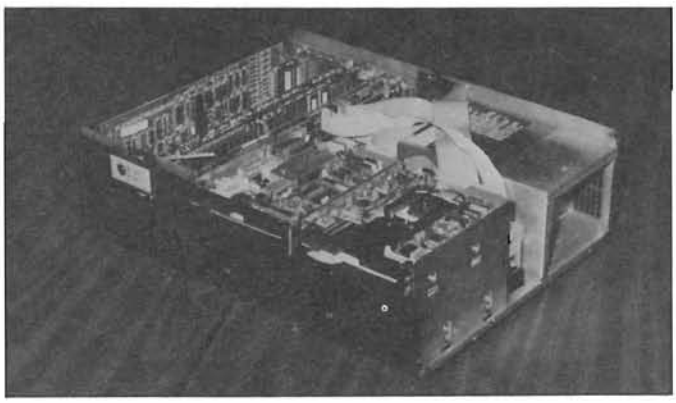
The first phase of assembly completed.
Power supply, backplane, circuit board guide,
and key lock assembly all installed.

Floppy/Winchester Drive Sub-Chassis





Floppy and Winchester drive installed
in one of the two sub-chassis.



Back panel layout of the HS-241

All hardware installed. Ready for action
with or without the cover!

A quick look at the keyboard layout will reveal that it is slightly different than that of the H/Z-100 PC. The backslash key, "\" is now located next to the right shift key (similar to that of the H/Z-100 system's keyboard). The keyboard now contains only one "ENTER/RETURN" key. The "ENTER" key that used to be on the far right, is a "+" key now. The "PRINT SCREEN" key has also been moved over to the far right. The "TILDE/GRAVE ACCENT" key is now located just to the left of the space bar. These changes, I feel are for the better. My only "want" would be for another ounce or so of spring tension in the keys themselves. It is also my understanding that the keyboard from the H/Z-100 PC computer is NOT electrically compatible with the H/Z-200 system keyboard, so the two CANNOT be swapped.

The MFM-200 monitor is very similar to the other Zenith "PC" monitors. Some of the monitor functions include:

1. Boot from disk
2. Display color bars
3. Display memory
4. Examine memory
5. Fill memory
6. Move memory block
7. Execute (Go)
8. Hex math
9. Input from a port
10. Output to a port
11. Examine registers
12. Search memory
13. Trace program
14. Unassemble program
15. Set video/scroll
16. Extended diagnostics
17. Setup mode

At this point, a note should be said about the "setup" mode. The Z-200 maintains various parameters about the system in battery backed-up CMOS RAM. These parameters are set using the MFM-200 monitor SETUP command. These parameters can be changed at any time, and need be only set once. The functions that can be set are:

1. Time
2. Daylight Savings Time switching ability
3. Date
4. Base memory size
5. Expansion memory size
6. Floppy drives 0 and 1 types
7. Boot drive or boot drive sequence
8. Video display: color, mono, enhanced, etc.
9. Video refresh rate 50/60 hz
10. Winchester drive types

The first thing "different" you'll notice about this system, is the blinding speed at which it operates. Simple functions like "directory displays" are almost instantaneous. Similar speed differences are realized in almost every program executed. Even though the CPU clock speed is 6 mhz, the no-wait-state memory, 80286 architecture, and 16 bit data buss, all contribute to its high efficiency and speed. I know you would all like to see some sort of speed comparison, so I used the "standard" Prime Number Seive program from Byte magazine. This is the program that does 10 iterations, and finds 1899 primes. This program was not I/O bound, and here are my results:

Standard H-161 — 14.3 seconds
8 mhz H-110 — 8.5 seconds
6 mhz HS-241 — 3.3 seconds

Even though there is a large speed difference between the HS-241 and H-161, the HS-241 is still fully PC compatible. Heath has provided a very sophisticated wait-state generator and the undocumented ability of slowing the system down to the speed of a standard PC (who knows, you might want to play some PC games on this brute). The following program listing will allow a user to change the speed of his H/Z-200 system to either full speed or "PC" speed. This program requires MASM, LINK, and EXE2BIN to assemble. Instructions for assembly and use can be found at the beginning of the listing.

```
   PAGE ,132
   TITLE CS

;*****************************************************;
;                                                     ;
;  Written by:  Jim Buszkiewicz                       ;
;  Original Idea:  D.W. Gohr                          ;
;                                                     ;
;  CS (change speed) puts the H/Z-200 computer into the ;
;  SLOW or FAST mode. In the SLOW mode it runs at the   ;
;  speed of the H/Z150  In the fast mode it runs at the ;
;  full speed of the 80286.                           ;
;                                                     ;
;  Use the following steps to create the program: CS.COM;
;                                                     ;
;  1  Assemble the source file  (MASM CS;)            ;
;                                                     ;
;  2. Link the object file. (LINK CS,CS;)             ;
;                                                     ;
;  3. Convert EXE file to COM. (EXE2BIN CS)           ;
;                                                     ;
;  4. Execute the program as follows:                 ;
;                                                     ;
;         A>CS S                <-- to run slow       ;
;         A>CS F                <-- to run fast       ;
;                                                     ;
;*****************************************************;

SCP_STAT        EQU     0064H
SCP_CMD         EQU     0064H

SCP_IBF         EQU     0002H
SCP_SLOW        EQU     00B1H
SCP_FAST        EQU     00B2H

DEF_FCB         EQU     005CH

CODE    SEGMENT
        ASSUME  CS:CODE
        ORG     00100H

start:
        mov     bx,DEF_FCB+1
        mov     al,byte ptr[bx]
        and     al,05fh         ;make it upper case
        cmp     al,'S'
        je      s1
        cmp     al,'F'
        je      s2
        mov     dx,offset usage ;show usage
        mov     ah,9            ;and
        int     21h
        int     20h             ;quit

s1:
        mov     dx,offset slowset
        mov     ah,9
        int     21h
        mov     bl,SCP_SLOW
        jmp     chg_speed
s2:
        mov     dx,offset fastset
        mov     ah,9
        int     21h
        mov     bl,SCP_FAST
chg_speed:
        mov     dx,SCP_STAT
```

```
csl:
        in      al,dx
        test    al,SCP_IBF
        jnz     csl
        mov     al,bl
        out     dx,al
        int     20h                     ;quit

slowset db      10,13,'Slow Speed Set',10,13,'$'
fastset db      10,13,'Fast Speed Set',10,13,'$'
usage   db      10,10,13,'Usage -->   A>CS S   or   A>CS F'
        db      10,13,'Where S=slow mode and F=fast
                       mode',10,13,'$'


CODE    ENDS
        END     START
```

While running the HS-241 in slow speed mode, my Prime Number Seive program took 14.7 seconds to execute; almost identical to that of the PC!

I hope no one tells my boss that this new computer runs three times faster than my old one, because Im sure he'll expect three times more work out of me, but that's his dream! *

Controlled Data Recording Systems Inc.

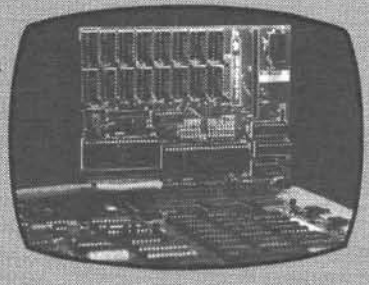
Quality Products and Support for the Heath/Zenith Community

New for the H/Z89-90 Computer Users:

The Super RAM 89 Package

Now get the speed and power of a high capacity Ram Drive System at a reasonable price.

Using the Ram 89 package, standard software shows an immense improvement in speed. Depending on the software being run, programs may execute 10 times faster than when run through standard floppies.



Now get RAM 89

HDOS DRIVER Software Only **$35**

RAM 89 DVD supports RAM SY0: and SY1:, added autoload features, plus more!

Ram 89 comes as a two card set that plugs into the left hand (16k expansion) side of the computer. No computer modifications required.

Board 1 has two banks of 256k chips possible for a total of 512k. Either or both banks are able to use 64k chips instead. Board 1 can be used by itself, with board 2 added at a later date.

Board 2 has an additional 512k, plus it has a real time clock capability, and a SASI interface hardware capability. Board 2 piggybacks onto board 1.

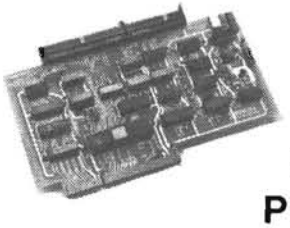
The Ram Drive Software (SRAM) allows one or two logical ram drives. The ram drive(s) can be located starting anywhere from logical A: to O: (standard drives get relocated). SRAM can be set to start at logical A: and warm boot with ram (no floppy disk accesses needed). Ram drive attatches to any of the versions of CP/M 2.2 bios used in the H/Z89-90.

GET A FULL MAGABYTE OF RAM FOR YOUR H/Z89-90

Board 1 includes hardware manual and ram drive software with no ram: $190.00. Each 256k bank, add $56.00. Board 2 (must have board 1). With no ram, no clock, no SASI $90.00. Each 256k bank, add $56.00. Ask about clock, SASI pricing.

FOR THE H/Z89-90 COMPUTERS

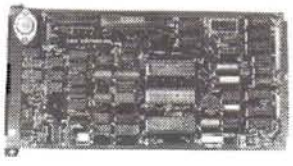
THE ONE CONTROLLER FOR 8" & 5.25" DRIVES THE FDC-880H PRICE: $395



Includes controller board CP/M boot prom, I/O decoder prom, hardware/software manuals BIOS source listing. HDOS driver now available for $50.00.

Now be able to run standard 8" Shugart compatible drives and 5.25" drives (including the H37 type) in double and single density, automatically with one controller. The FDC-880H operates with or without the Heath hard sectored controller.

THE FDC-H8 DOUBLE DENSITY 8" AND 5.25" CONTROLLER FOR THE H8 COMPUTER PRICE $495



Has all of the capabilities of our popular FDC-880H controller, with added features:

- Direct memory access (DMA) data transfer
- Hard sectored controller (H17) incorporated on the board.
- Runs with the standard 8080 CPU card and with Z80 CPU upgrades.

MODIFY 89

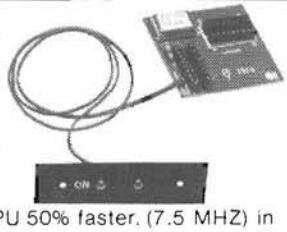
Allows owners of the FDC-880H or the FDC-H8 to read/write to the following disktypes:

Access S.S.D.D., Cromemco S.S.D.D., DEC VT180 S.S.D.D., IBM PC/Zenith 100, (CP/M) D.S.D.D.*, Kaypro II S.S.D.D., Morrow Micro Decisions S.S.D.D., NEC PC-8001 A S.S.D.D., Osborne S.S.D.D., Osborne S.S.D.D., Otrona D.S.D.D.*, Zorba D.S.D.D., TI Professional S.S.D.D., TRS-80 Model I (Omnikron CP/M), TRS-80 Model III (MM CP/M), Xerox 820 S.S.D.D., Zerox 820-II S.S.D.D. Plus more, Standard (Tests for H/Z37 and C.D.R. Disk types)

PRICE $49.95

FOR THE Z100 COMPUTERS

THE ORIGINAL Z100 SPEED MODULE. RUN YOUR Z100 PROGRAMS FASTER



The ZS100 runs the Z100 CPU 50% faster. (7.5 MHZ) in 8088 mode. The ZS100 installs easily with no soldering. The ZS100 is externally switchable between speed mode and normal. The ZS100 improves the time performance of applications packages with no software modifications needed. The ZS100 is for all Heath/Zenith 100, 110 and 120 series computers.

SUPER DRIVES 85

A powerful floppy drive package for the Z100 or any computer that can use 8" floppy drives.

Two extended technology 5.25" drives in a package with interface cables, case and power supply, that look exactly like standard 8" floppy drives to the computer. Now have **2.4 megabytes** of **floppy file storage capacity** in one 5.25" package.

ONLY **$695** LIST

Dealer inquiries invited.

For information about these products contact.
Controlled Data Recording Systems Inc.
7210 Clairemont Mesa Blvd.
San Diego, Ca 92111 Phone (619) 560-1272 or contact a Heath/Zenith dealer that carries the C.D.R. Systems Inc. line of products.
New - C.D.R. Systems Bulletin Board (619) 560-8929

PAINT.ASM — Part 1

Steven W. Vagts
9509 Gray Mouse Way
Columbia, MD 21046

Ever since I built my Heathkit H-88 in 1981, I had been somewhat disappointed with its graphics capability and documentation on use. It was only after a laborious effort and much midnight oil that I finally wrote my first program using its graphics set.

In 1983, I built an H-89 with the disk drive, and renewed my efforts to write better graphics routines and more sophisticated programs. Still, I wanted a better way.

Then in SEXTANT #9, March and April 1984, Siebert Ickler's article "Screen Dump!" came to my salvation. Here was the basis that I needed to write a program that could somewhat painlessly "paint" whatever Heath/Zenith (H/Z) graphics I desired on the screen, save the finished product to memory and disk, then reload it at will.

Limited to the Heath H-14 printer until recently, I have now reached the culmination of my efforts — finished printouts of a graphics-packed screen dump to a near letter quality, dot matrix printer.

To repay Mr. Ickler and all the other authors who unselfishly furnished excellent, informative articles to SEXTANT and REMark magazines, major contributors to my computer education, I have enclosed the fruits of my labors — "PAINT.ASM".

More than that, though, I hope this program may spark renewed interest on behalf of more individuals to become interested in assembly language.

I've noticed a dearth of assembly language programs in magazines and those articles seem to be submitted by only a certain core of individuals, not that I can throw any stones, since this is my first.

The second thing I've noticed was that those articles were very specialized, covering only certain small aspects of assembly language. Some cover disk operations, some address screen dump programs, and others give basic assembly language tutorials. Each encloses short programs to stress their points and leave it to the reader to merge these disjointed programs into a usable game, utility, or what have you.

By the time a person has acquired all the pieces, and filtered out the conflicting information on which way is best, he has either forgotten a few program tricks discussed last year, or given the whole process up as hopeless.

This article will hopefully provide two things. First, I want to develop interest from programmers, like myself, that have the basic knowledge of modifying a program and assembling it, but lack a basic all-around starting point.

Second, I needed a basic established program to start from for next month's article, which would then build on this first program. It provides an explanation of how a certain group of otherwise excellent printers, but lacking the necessary graphics programs, could be put to proper use. It also provides information on programmed computer control, adding to the learning experience.

This month, I will describe the inner workings of an abbreviated program. While it is abbreviated, it is nonetheless a whole program. It demonstrates the use of system calls, H/Z graphics, simple printer operation and disk file manipulation. All it lacks is some minor additional capability or modification, easily within the range of a novice, to make a very complete and useful utility.

Suggested areas of improvement: add two on-screen help lines to display the normal characters and have their graphics symbols represented immediately below. If that routine is assigned one of the remaining special function keys, then the other two could represent some other minor aid, such as deleting to the beginning of a line, or moving a block of screen text.

I've got ideas on moving blocks of text from one portion of the screen to another, but haven't tried them yet. Perhaps someone out there has some ideas on this and would care to share it?

This initial program will send the product to the Heath H-14 printer — less the graphics and reverse video, but valuable in its own right because it shows how the special characters are saved and used on the screen. Efforts to print the escape codes to this

printer had only resulted in torn paper as all the print head pins attempted to work at the same time and jammed – not a healthy act. So escape codes are filtered out for use with this printer.

Next month, I will add to "PAINT.ASM" to enable printing the graphics and reverse video to the Panasonic KX-P1092 dot matrix, near letter quality printer, enabling the development of personalized letterhead, cards, and graphic prints, to name a few suggested ideas. It will involve loading our own font into the printer — a rather tedious chore, but worth the effort.

System Hardware/Software

Getting off my soapbox, but before getting into the meat of this article, I need to address the hardware aspects of my system, presently consisting of an H/Z-89 with 64k of RAM (Random Access Memory).

I recently upgraded my operation with a CDR soft-sector controller board (FDC-880H), H&H parallel interface board (KC1222), and dual double-sided disk drives (Floppy Disk Service's 'TWOET' System) while still keeping my original hard-sector controller and the original drive in a separate enclosure. A slick operation.

An earlier version of "PAINT.ASM" had been working on the single drive with H/Z CP/M BIOS 2.2.04 prior to the upgrade. Incidently, I modified both the H/Z CP/M and the new CDR BIOS to utilize my H-14 printer on serial port 340Q and the new KX-P1092 off the parallel port of the H&H parallel board (a story in itself). During the cold boot, you choose which printer you will use.

The enclosed program, therefore, should have no problem operating under H/Z or CDR CP/M.

The Program – Finally!

I believe the program is fairly straightforward, so rather than an indepth analysis of each area, I will hit the highpoints and some of the trouble areas.

The program begins by displaying a page of instructions on the terminal screen. These explain the various codes, special function keys and Control (CTRL) keys. Upon using an ESC-E to clear the screen, the reader is left with his imagination to begin filling the screen as he sees fit.

This program makes maximum use of preprinted messages, stored toward the end of the program. The messages are printed until a delimiter is found, the '$' in this program, indicating the end of the message. Messages may merge with other messages.

MSG1 is an example of a merging message. It continues into MSG after printing the page of instructions. MSG prepares line 25 for use, prints brief key names to the top row of function keys, returns the cursor to "HOME", and shifts the numeric keypad. MSG is also printed following any use of line 25 for other purposes.

Shifting the keypad allows the arrows, Delete Character (DC), Delete Line (DL), Insert Character (IC) and Insert Line (IL) keys to operate as one might expect.

The function keys are used for toggling between graphics, reverse video and normal print. Key f5 is not used at this point, but will be put to good use next month.

MSG50 erases and enables line 25, displays the special function key names, and prints information on file I/O status without being printed as part of the screen. This line is also used to enter the file name for saving or loading the screen to or from the disk.

CTRL-D dumps the screen to memory, while CTRL-L loads the screen from memory. CTRL-P sends the screen to the printer (limited operation this month). CTRL-W writes the screen to the disk and CTRL-R reads the disk file to the screen. CTRL-E ends the program. These keys are all checked for in the INPUT and XCAPE routines.

This program uses the built-in functions of the CP/M operating system, the system calls, to perform the basic input/output (I/O) functions of the program. This simplifies the program considerably and gives the program some degree of interoperability with other CP/M computers. H/Z graphics peculiarities, however, make interoperability of little concern to us in this program. The system calls used here and others are described in detail in the CP/M 2.2 Interface Guide.

As described in Mr. Ickler's article, "Screen Dump!", the transmit page command is the key to the operation of this program. For such a simple command, ESC-#, causes the contents of the screen, including its imbedded escape sequences, to be sent to memory — at location BUFR at the end of our program.

The screen is transmitted from the upper left corner of the screen, 80 characters per line, including blanks. Line 25 is not sent and escape codes do not count as part of the 80 character line. At the end of line 24, the first and only carriage return is sent, marking the end of transmission.

Since this is the only carriage return, when the memory image is sent to the disk, screen or printer, it is necessary to insert additional carriage returns as needed. Therefore, an accurate count of all characters must be made during these evolutions and escape codes must be accounted for.

From the BUFR it is relatively easy to redisplay the screen, save it to disk, or send it to the printer. The screen is not cleared before redisplaying, so overlays of pictures are possible.

The File Control Block

The File Control Block (FCB) is another stumbling block in understanding this program, but an understanding of its function is essential to any programming effort involving disk files.

As addressed in the CP/M 2.2 Interface Guide, "The File Control Block (FCB) data area consists of a sequence of 33 bytes for sequential access and a series of 36 bytes in the case that the file is accessed randomly." We will concern ourselves with only the first 33 bytes.

The Default File Control Block (DFCB) begins at memory location 05CH and is defined at the beginning of the program with the system calls. Let's discuss the FCB in detail.

The first byte of the FCB (byte 0) contains the drive code, where 0 is the current drive (default) and numbers 1 thru 16 represent drives A thru P. Bytes 1 thru 11 of the FCB contain the file name and type. These bytes, we will be using extensively, especially next month.

Byte 12 contains the current extent number. To help explain its use, we will use the Read Sequential function, System Call #20.

During the sequential read, a 128 byte record is read from the file and placed into memory at the current DMA (Direct Memory Address), a temporary holding area for data being read from or written to a disk. Each time a 128 byte record is read, the extent

number is increased by 1. This byte works closely with bytes 15 and 32, which are reset to zero with each new extent number.

Byte 15 identifies each byte of the record that was read in the current extent number and ranges in value from 0 – 128.

Byte 32 represents the current record to read or write in a sequential file operation.

The record is read from the position defined by the present value of byte 32 and is incremented with each byte of new data. If it overflows (exceeds 128 bytes), then the next logical extent is automatically opened and byte 32 is reset to zero.

The remaining bytes are of little concern to us for this program.

Notice that before using the FCB, it is best to clear it. It is easiest to do this by inserting blanks in the first 31 bytes, then insert zeros in bytes 0, 12, 13, 14, 15 and 32. (See the CLRFCB routine.)

So what does the File Control Block do for us?

Well, whenever we specify a file name for disk operations, the FCB is used. Let's look at the routine, WDISK, in our program. We want to write our screen to a disk file.

After clearing the FCB, the program requests the file name. This is placed temporarily in the line buffer, LINEB, where it is checked to insure it is not too long. Then, it is loaded into the FCB in the FILIT routine. The first letter is loaded into byte 1 of the FCB.

To ensure this isn't the drive code, we also check if the next letter of the file name is a colon. If it is, the first letter we loaded into byte 1 of the FCB is moved to byte 0, to designate the disk drive to be used.

Subsequent letters are loaded into bytes 1 through 8 of the FCB, checking for a decimal point as we do. If a decimal point is found, we know that the file name is complete, and load the file type, bytes 9 thru 11 of the FCB, with any remaining letters.

We also must check for a blank, which would signify the end of the file name without a file type. The remaining FCB remains blank.

Armed with the file name installed in the FCB, we are then ready to attempt to open a file on the appropriate disk. If we cannot open the file, we next attempt to make a file using the MKFIL routine. If unsuccessful, the directory must be full, so the appropriate message is printed to line 25 of the screen. If successful, we proceed with writing to the file.

So the FCB is very important. Next month, we will proceed a little deeper, having the program check which disk our data file may be on.

Closing

By now I'm sure many of you have thought of several uses for this program. Let's compare notes. In addition to the obvious use of dumping the screen to a printer and all the advantages there, game programmers could have a ball with this.

How many game programmers have spent hours trying to design screen graphics for their programs? It's been driving me up the wall for quite some time. Here's a possible shortcut.

Design your initial screen of graphics using this program and save your finished product to disk. Take the portion of this program that reads the file from the disk and displays it on the screen, and place it in your game program. Now have your game program call the file, display the initial screen as a start point for the games use, then modify the graphics through your screen as you normally would. Depending upon disk space, numerous screens could be saved for your program's use.

For example, when I received the Heath Users' Group disk on "BASIC-E", it included an excellent program, "STARTREK". All it lacked was some screen graphics to give a visual display of the action. A subroutine could easily be added to the program to read the graphics file from the disk and display its contents to the screen at appropriate points. I've already created files showing the USS Enterprise, starbases, Klingon vessels, etc.

Once the initial screen is displayed, the action graphics takes over. While I'm only speculating, I've tested the theory on a background screen of stars, planets and spacecrud, and then made a planet explode quite easily.

Anyone have other ideas?

As I've already stated, next month we will add the routines necessary to turn the enclosed program into a real printing utility — sending our screen graphics to a printer.

To aid in troubleshooting — load the enclosed program before adding next month's additions. Next month's changes will have minimal effect on this program — as you can see, the link will be at the addition of the f5 key — and a change of PRNTR routine to use a bit image, programmable font printer such as the Panasonic KX-P1092 printer. So to save the hassle, ensure this program works properly before continuing.

I have pared "PAINT.ASM" considerably from my version for the purposes of this article, but it gives a firm base from which to customize your own. I would be happy to send the source code for everything except the data files (these consume a lot of paper and are easy to develop) for $3.00. I'll also send the finished product on disk, in the format discussed above for $7.00, if you include a preformatted disk. Include a phone number in case I have problems.

I'm interested in any improvements readers feel may be appropriate and I'd be happy to address questions if self-addressed, stamped envelopes are included.

See you next month.

Equipment and Information Sources Mentioned

In addition to REMark Magazine & Heath/Zenith Computers:

FDC-880H Double Density Disk Controller
C.D.R. Systems Incorporated
7210 Clairemont Mesa Blvd
San Diego, CA 92111

KC1222 Serial/Parallel Interface Board
H&H Computer Enterprises, Inc.
P.O. Drawer H
Blacksburg, VA 24060

Panasonic KX-P1092 Multi-Mode Printer
Panasonic Industrial Company
One Panasonic Way
Secaucus, NJ 07094

SEXTANT Magazine
Sextant Publishing Co.
716 E Street, S.E.
Washington, DC 20003

CP/M is a registered trademark of
Digital Research Corporation
P.O. Box 579
Pacific Grove, CA 93950

Listing

```
; VAGTS PAINT PROGRAM
;
; CP/M SYSTEM CALLS
CONIN   EQU     1       ;CONSOLE INPUT
CONOUT  EQU     2       ;CONSOLE OUTPUT
LSTOUT  EQU     5       ;LIST DEVICE (LPT) OUTPUT
CONSIO  EQU     6       ;DIRECT CONSOLE I/O
PRINT   EQU     9       ;PRINT STRING
LINPUT  EQU     10      ;LINE INPUT FUNCT
OPEN    EQU     15      ;OPEN FILE
CLOSE   EQU     16      ;CLOSE FILE
READ    EQU     20      ;READ FILE (SEQUENTIAL)
WRTFIL  EQU     21      ;WRITE FILE (SEQUENTIAL)
MAKFIL  EQU     22      ;MAKE FILE
BDOS    EQU     5       ;BDOS VECTOR
DFCB    EQU     05CH    ;DEFAULT FCB
DMA     EQU     080H    ;DMA AREA
ESCAPE  EQU     27      ;ESC CHAR

        ORG     100H
START   EQU     $
        LXI     H,0     ;ZERO HL REG
        DAD     SP      ;LOCATE STACK
        LXI     SP,STACK  ;SET OUR OWN
        PUSH    H       ;SAVE CP/M'S STACK
        PUSH    PSW     ;SAVE CP/M'S FLAGS
        LXI     D,MSG1
        CALL    PMSG

INPUT   MVI     C,CONIN
        CALL    BDOS    ;DIRECT INPUT
        CPI     01BH    ;ESC CODE?
        JZ      XCAPE
        CPI     008H    ;BACKSPACE?
        JZ      DELET
        CPI     07FH    ;DELETE?
        JZ      DELET
        CPI     00DH    ;CR?
        CZ      LF
        CPI     004H    ;CTRL-D?
        JZ      DUMP
        CPI     00CH    ;CTRL-L?
        JZ      DSPLY
        CPI     010H    ;CTRL-P?
        JZ      PRNTR
        CPI     017H    ;CTRL-W?
        JZ      WDISK
        CPI     012H    ;CTRL-R?
        JZ      RDISK
       CPI     005H    ;CTRL-E?
        JZ      EXIT1
        JMP     INPUT

DELET   MVI     E,01BH  ;ESCAPE
        MVI     C,CONOUT
        CALL    BDOS
        MVI     E,04EH  ;'N'
        MVI     C,CONOUT
        CALL    BDOS
        JMP     INPUT

XCAPE   MVI     E,0FFH  ;REQ INPUT CHAR
        MVI     C,CONSIO
        CALL    BDOS    ;A=CHAR
        ORA     A       ,CHECK IF 0
        JZ      XCAPE   ;A=0 FOR SOME REASON!
        CPI     053H    ;IS IT f1?
        JZ      RVID
        CPI     054H    ;IS IT f2?
        JZ      ERVID
        CPI     055H    ;IS IT f3?
        JZ      GRAF
        CPI     056H    ;IS IT f4?
        JZ      EGRAF
        CPI     057H    ;IS IT f5?
        JZ      CHFONT
TXSPCL  STA     SPCL
        MVI     E,ESCAPE
        MVI     C,CONOUT
        CALL    BDOS
        LDA     SPCL
        MOV     E,A
        MVI     C,CONOUT
        CALL    BDOS
        JMP     INPUT

RVID    MVI     A,070H  ;'p'
        JMP     TXSPCL
ERVID   MVI     A,071H  ;'q'
        JMP     TXSPCL
GRAF    MVI     A,046H  ;'F'
        JMP     TXSPCL
EGRAF   MVI     A,047H  ;'G'
        JMP     TXSPCL

MSG50   PUSH    D
        LXI     D,MSG2  ,ERASES & ENABLES LINE 25
        CALL    PMSG
        POP     D
        CALL    PMSG
        RET

DUMP    LXI     D,MSG3  ;DUMP TO MEMORY & TXMIT PAGE
        CALL    MSG50
       CALL    SAVE
        JMP     EMSG    ;END OF MSG

SAVE    LXI     H,BUFR
TXCH    PUSH    H
        MVI     E,0FFH  ;REQ INPUT CHAR
        MVI     C,CONSIO
        CALL    BDOS    ;A=CHAR
        ORA     A       ;CHECK IF 0
        POP     H
        JZ      TXCH
        MOV     M,A
        INX     H
        CPI     15Q     ;CR SENT AT END OF SCREEN
        JNZ     TXCH
        RET

DSPLY   LXI     D,MSG21 ;CLEAR SCREEN
        CALL    PMSG
        LXI     D,MSG4  ;RECALL SCREEN FROM MEMORY
        CALL    MSG50
        LXI     H,BUFR
DSPLN   MOV     A,M     ;GET CHARACTER
        CPI     15Q     ;CR? MEANS END OF SCREEN
        JZ      EMSG    ;END DISPLAY
        CPI     ESCAPE
        LDA     COUNT
        JZ      ADDONE  ;ALLOW FOR ESC CODE IN COUNT
        DCR     A
        STA     COUNT
        CZ      SCRLF   ;CRLF IF 80 CHARS PRINTED
        JMP     LIST    ;LIST TO SCREEN
ADDONE  INR     A
        STA     COUNT
LIST    PUSH    H
        MOV     E,M
        MVI     C,CONOUT
        CALL    BDOS
        POP     H
        INX     H
        JMP     DSPLN

CHFONT  JMP     TXSPCL  ;FUTURE ADDITION
PRNTR   LXI     D,MSG5  ;PRINTING SCREEN
        CALL    MSG50
        CALL    SAVE
```

```
        LXI     H,BUFR  ;GOTO START OF BUFR
PRLP    MOV     A,M     ;GET CHAR
        CPI     15Q     ;CR? MEANS END OF SCREEN
        JZ      LPDONE
        MOV     E,A     ;E=CHAR
        CPI     ESCAPE
        JZ      SKIP    ;SKIP ESCAPE CODES FOR H14
        LDA     COUNT
        DCR     A
       STA     COUNT
        CZ      SETCT   ;RESET THE COUNT IF Ø
        CALL    PRNTCH
        INX     H
        JMP     PRLP

PRNTCH  PUSH    PSW
        PUSH    H
        MVI     C,LSTOUT ;E HAS CHAR
        CALL    BDOS
        POP     H
        POP     PSW
        RET

SKIP    INX     H       ;SKIP ESC CODE &
        INX     H       ;FOLLOWING CHAR
        JMP     PRLP

CLRFCB  LXI     D,DFCB+1  ,BEGIN OF FCB
        MVI     B,31
        MVI     A,' '
CLEAR   STAX    D       ;CLEAR FCB
        INX     D
        DCR     B
        JNZ     CLEAR
        XRA     A
        STA     DFCB    ;ZERO DRIVE CODE
        STA     DFCB+12 ;ZERO EXTENT
        STA     DFCB+13
        STA     DFCB+14
        STA     DFCB+15 ,ZERO RECORD COUNT
        STA     DFCB+32 ;ZERO CURRENT RECORD
        RET


FILNM   CALL    CLRFCB
        LXI     D,MSG13 ;REQUEST FNAME
        CALL    MSG5Ø
        LXI     D,LINEB ;POINT TO LINE BUFR
        MVI     C,LINPUT
        CALL    BDOS    ;INPUT THE LINE
        LXI     H,LINEB+1  ;POINT TO # OF CHARS
        MOV     C,M     ;GET THE COUNT
        INX     H
        MVI     B,Ø
        DAD     B       ;GET END OF USED LINEB
        MVI     M,' '   ;STORE A BLANK AS MARKER
        MVI     B,12
CKL     DCR     C
        JZ      LNGTH   ;LENGTH OK
        DCR     B
        JZ      LNERR   ,LENGTH TOO LONG
        JMP     CKL
LNERR   LXI     D,MSG17 ;FNAME TOO LONG
        CALL    MSG5Ø
        CALL    DELAY
       MVI     A,Ø
        RET
LNGTH   MVI     B,8
        LXI     D,DFCB  ;POINT TO BEGIN OF FCB
        LXI     H,LINEB+1  ;POINT TO BEGIN OF LINEB
FILIT   INX     D
        INX     H
        MOV     A,M     ;GET LETTER OF NAME
        CPI     Ø3AH    ;CHECK FOR COLON
        JZ      COLON   ;DRIVE # GIVEN
        CPI     Ø2EH    ;CHECK FOR DECIMAL
        JZ      BLNK
        CPI     ' '     ;CHECK FOR BLANK
        RZ
        STAX    D       ;STORE IT IN FCB
        DCR     B       ;A CHAR WAS STORED
        JNZ     FILIT   ;LOAD ANOTHER CHAR
        INX     H
        MOV     A,M
        CPI     Ø2EH    ;LAST CHECK FOR DECIMAL
        JNZ     LNERR   ;LENGTH TOO LONG
FILTYP  MVI     B,3     ;3 CHARS FOR FILE TYPE
        LXI     D,DFCB+9  ;POINT TO FCB TYPE
TYPLP   INX     H
        MOV     A,M
        CPI     ' '     ;CHECK FOR BLANK
        RZ
        STAX    D
        INX     D
        DCR     B
        JNZ     TYPLP
        RET

COLON   LXI     D,DFCB+1  ,POINT TO FNAME BEGIN
        LDAX    D       ;GET LETTER OF DRIVE
        SUI     Ø4ØH    ;SUBTRACT Ø4ØH TO GET DRIVE #
        DCX     D       ;GO TO BEGIN OF DFCB
        STAX    D       ;STORE DRIVE #
        MVI     B,8     ;RESET B
        LXI     H,LINEB+3  ;POINT TO FNAME BEGIN
        INR     C
        INR     C
        JMP     FILIT

BLNK    MVI     A,' '
BLNK1   STAX    D       ;STORE BLANK
        INX     D
        DCR     B
        JNZ     BLNK1
        JMP     FILTYP

FILCHK  CALL    FILNM   ;GET FILE NAME
        ORA     A
        RZ
        LDA     DFCB+1
       CPI     ' '     ;CHK FOR BLANK - NO FILE
        RNZ
        LXI     D,MSG1Ø ;NO FILE NAME
        CALL    MSG5Ø
        CALL    DELAY
        MVI     A,Ø
        RET

OPNFIL  MVI     C,OPEN
        LXI     D,DFCB
        CALL    BDOS    ;TRY TO OPEN FILE AS IF EXISTS
        RET             ;A=255 IF ERROR

WDISK   CALL    FILCHK
        ORA     A
        JZ      EMSG
        CALL    OPNFIL
        CPI     255
        JNZ     CONT    ;OPEN OK
MKFIL   MVI     C,MAKFIL
        LXI     D,DFCB
        CALL    BDOS    ;CREATE FILE
        CPI     255     ;A=255 IF ERROR
        JNZ     CONT    ;MAKE OK
        LXI     D,MSG11 ;NO ROOM IN DIRECTORY
        CALL    MSG5Ø
        CALL    DELAY
        JMP     EMSG

CONT    LXI     D,MSG6  ;SAVING TO DISK
        CALL    MSG5Ø
        CALL    SAVE
        LXI     H,BUFR
        SHLD    BPTRL   ;SET NEXT LINE TO BUFR START
        LXI     H,DMA
        SHLD    DMAPTR  ;SET DMAPTR TO START OF BUFR
NEXTLN  LHLD    BPTRL
        SHLD    BPTRM   ;SET NEXT CHAR TO LINE START
        MVI     A,15Q   ;CR?
        CMP     M
        JZ      ENDLP   ;IF CR THEN WE ARE DONE
```

```
        MVI     A,80    ;# CHARS IN A LINE
        STA     COUNT
        STA     CHK
FNDLN   LHLD    BPTRL
        MVI     A,ESCAPE
        CMP     M
        JNZ     FNDLN2
        LDA     COUNT
        INR     A
        INR     A
        STA     COUNT   ;ADD 2 TO ALLOW FOR ESC SEQ
        INX     H
        MVI     A,'p'
        CMP     M
       JNZ     FNDLN4
        MVI     A,1
        STA     RVFLAG  ;ESC p FOUND - SET RVFLAG
        JMP     FNDLN3
FNDLN4  MVI     A,'q'
        CMP     M
        JNZ     FNDLN3
        XRA     A
        STA     RVFLAG  ;ESC q FOUND - RESET RVFLAG
        JMP     FNDLN3
FNDLN2  INX     H
        SHLD    BPTRL
        LDA     CHK
        DCR     A
        STA     CHK
        JZ      BACKUP  ;END OF LINE - BACKOFF BLANKS
        JMP     FNDLN
FNDLN3  INX     H
        SHLD    BPTRL
        JMP     FNDLN

BACKUP  LDA     RVFLAG
        ORA     A       ;SET ZERO FLAG
        JNZ     DISKIT  ;RV IN PROGRESS - BLANKS COUNT
        DCX     H
        MVI     A,' '
        CMP     M
        JNZ     DISKIT  ;LAST NON-BLANK FOUND
        LDA     COUNT
        DCR     A
        STA     COUNT
        JZ      DCRLF   ;NO NON-BLANKS - PUT OUT CRLF
        JMP     BACKUP

DISKIT  LHLD    BPTRM
        MOV     A,M
        INX     H
        SHLD    BPTRM
        CALL    DISKOUT
        LDA     COUNT
        DCR     A
        STA     COUNT
        JNZ     DISKIT
DCRLF   MVI     A,0DH   ;CR
        CALL    DISKOUT
        MVI     A,0AH   ;LF
        CALL    DISKOUT
        JMP     NEXTLN

LPDONE  MVI     E,0DH
        CALL    PRNTCH
        MVI     E,0AH
        CALL    PRNTCH
        MVI     A,81
        STA     COUNT
       JMP     EMSG

SCRLF   PUSH    H
        CALL    CRLF
        POP     H
SETCT   MVI     A,80    ;RESTORE COUNT
        STA     COUNT
        RET

CRLF    MVI     E,0DH
        MVI     C,CONOUT
        CALL    BDOS
LF      MVI     E,0AH
        MVI     C,CONOUT
        CALL    BDOS
        RET

DISKOUT LHLD    DMAPTR  ;A REG TO BUFR
        MOV     M,A
        INX     H
        SHLD    DMAPTR
        XRA     A
        CMP     L
        RNZ             ;NOT TO END OF BUFR YET
TODISK  LXI     H,DMA
        SHLD    DMAPTR  ;RESET DMAPTR TO BEGIN
        MVI     C,WRTFIL
        LXI     D,DFCB
        CALL    BDOS
        ORA     A       ;SET ZERO FLAG
        RZ              ;GOOD WRITE
        LXI     D,MSG12
        CALL    MSG50   ;NO MORE ROOM
        JMP     DKDONE

ENDLP   LHLD    DMAPTR  ;END OF DISKOUT
        DCX     H
        MVI     A,1AH   ;CTL-Z MEANING EOF
        MOV     M,A
        INX     H
        XRA     A
        CMP     L
        JZ      ENDFL
FILLZ   MOV     M,A     ;FILL BUFR W/ZEROS
        INX     H
        CMP     L
        JNZ     FILLZ
ENDFL   CALL    TODISK  ;WRITE OUT PADDED RECORD
        JMP     DKDONE

RDISK   CALL    FILCHK
        ORA     A
        JZ      EMSG
        CALL    OPNFIL
        CPI     255     ;A=255 IF ERROR
       JNZ     CONT1   ;OPEN OK
        LXI     D,MSG14 ;NO FILE IN DIRECTORY
        CALL    MSG50
        CALL    DELAY
        JMP     EMSG
CONT1   LXI     D,MSG7  ;RECALLING FROM DISK
        CALL    MSG50
RLOOP   LXI     D,DFCB  ;FCB ADDRESS
        MVI     C,READ
        CALL    BDOS    ;TRY TO READ A RECORD
        ORA     A       ;READ OK?
        JNZ     UNSUC2
        LXI     H,DMA   ;POINT TO DMA AREA
        MVI     B,128   ;PRINT 128 CHARS
PLOOP   MOV     A,M     ;GET A CHAR
        CPI     'Z'-40H ;CTRL-Z?
        JZ      DKDONE
        MOV     E,A     ;ELSE PUT CHAR IN E
        PUSH    H
        PUSH    B
        MVI     C,CONOUT
        CALL    BDOS    ;PUT CHAR ON SCREEN
        POP     B
        POP     H
        INX     H
        DCR     B
        JNZ     PLOOP   ;LOOP UNTIL 128 CHARS PRINTED
        JMP     RLOOP

DKDONE  CALL    CLOSDK
        JMP     EMSGD

NOREAD  LXI     D,MSG15 ;UNSUCCESSFUL READ
        CALL    MSG50
CLOSDK  MVI     C,CLOSE
        LXI     D,DFCB
```

```
        CALL    BDOS    ;CLOSE FILE
        RET

UNSUC1  CALL    NOREAD
        MVI     A,0
        RET
UNSUC2  CALL    NOREAD
        CALL    DELAY
EMSGD   MVI     A,81
        STA     COUNT   ;RESTORE COUNT TO 81
EMSG    LXI     D,MSG
        CALL    PMSG
        JMP     INPUT   ;RETURN TO PAINT MODE

PMSG    PUSH    PSW
        PUSH    H
        MVI     C,PRINT
        CALL    BDOS
        POP     H
        POP     PSW
        RET

DELAY   LXI     D,MSG16 ;TYPE ANY KEY
        CALL    PMSG
        MVI     C,CONIN
        CALL    BDOS    ;ABSORB CHAR
        RET

EXIT    CALL    DELAY
EXIT1   LXI     D,MSG8
        CALL    PMSG
        POP     PSW     ;GET CP/M FLAGS
        POP     H       ;GET CP/M STACK
        SPHL            ;SET IT
        RET             ;RETURN TO CP/M
```



```
MSG1    DB      01BH,045H,'Welcome to PAINT ',13,10
        DB  'Paint a picture you desire using the arrows to move the cursor ',13,10
        DB  'Special function keys are defined as:',13,10
        DB  '    f1    - sets  REVERSE VIDEO',13,10
        DB  '    f2    - exits REVERSE VIDEO',13,10
        DB  '    f3    - sets  GRAPHICS CHARACTERS',13,10
        DB  '    f4    - exits GRAPHICS CHARACTERS',13,10
        DB  '    f5    - changes PRINTER FONT (NOT H-14)',13,10
        DB  '    ERASE - ERASES or clears entire screen',13,10
        DB  '    ESC E - CLEARS THE SCREEN (EVERYTHING IS LOST!)',13,10,10
        DB  'When you are done painting, type:',13,10
        DB  '    CTRL-D  - DUMP  the painting.',13,10
        DB  '    CTRL-L  - LOAD  the painting.',13,10
        DB  '    CTRL-P  - PRINT to printer.',13,10
        DB  '    CTRL-W  - WRITE to disk ',13,10
        DB  '    CTRL-R  - READ  from disk ',13,10
        DB  '    CTRL-E  - ENDS  the program.',13,10
        DB  'When ready to start, hit ESC E to clear the screen '
MSG     DB  01BH,079H,031H,01BH,078H,031H,01BH,079H,035H,01BH,059H,038H,025H
        DB  01BH,'pRVID',01BH,'q    ',01BH,'pNVID',01BH,'q    ',01BH,'pGRAF'
        DB  01BH,'q    ',01BH,'pNORM',01BH,'q    ',01BH,'pFONT',01BH,'q    '
        DB  01BH,059H,020H,020H,01BH,078H,036H,'$'
MSG2    DB      01BH,079H,031H,01BH,078H,031H,01BH,059H,038H,02BH
        DB      01BH,078H,035H,01BH,04FH,01BH,071H,01BH,047H,'$'
MSG3    DB      'DUMPING SCREEN TO MEMORY ',01BH,023H,'$'
MSG4    DB      'RECALLING SCREEN FROM MEMORY ',01BH,059H,020H,020H,020H,'$'
MSG5    DB      'PRINTING SCREEN ',01BH,023H,'$'
MSG6    DB      'SAVING TO DISK.',01BH,023H,'$'
MSG7    DB      'RECALLING FROM DISK.',01BH,059H,020H,020H,020H,'$'
MSG8    DB      01BH,079H,036H
MSG9    DB      01BH,059H,020H,020H,01BH,079H,031H,01BH,079H,035H,01BH,045H,'$'
MSG10   DB      'NO FILE SPECIFIED ON COMMAND LINE!$'
MSG11   DB      'NO ROOM ON DIRECTORY TO CREATE FILE!$'
MSG12   DB      'NO MORE ROOM ON DISK!$'
MSG13   DB      01BH,079H,035H,'WHAT IS THE FILE NAME?$'
MSG14   DB      13,10,'NO FILE IN DIRECTORY!$'
MSG15   DB      'UNSUCCESSFUL READ OPERATION!$'
MSG16   DB      '   TYPE ANY KEY:$'
MSG17   DB      'FILE NAME TOO LONG!$'
MSG21   DB      01BH,045H,'$'

INT     DS      1
SPCL    DB      0
COUNT   DB      81      ;# CHARS IN A LINE
CHK     DB      80      ;# CHARS TO CHK FOR A PRNT LINE
LINEB   DS      20      ;LINE INPUT BUFFER
RVFLAG  DB      0       ;1 IF RV ESC IN PROG; 0 OTHERWISE
BPTRL   DS      2       ;BUFR NEXT LINE POINTER
BPTRM   DS      2       ;BUFR NEXT CHAR TO MOVE/WRITE
DMAPTR  DS      2       ;NEXT LOC IN DMA FOR MOVE
BUFR    DS      6000    :SPACE FOR STACK
STACK   DS      32      :PUT STACK HERE
START   EQU     $
        END
```

*

First Capitol Delivers Zenith. Your Way.

Have it your way.

If you're looking for a PC, you couldn't do better than Zenith. And you couldn't buy your Zenith at a better place than First Capitol. We'll put together a system the way YOU want it. And we'll do it for a price that competes with the guy who just sells boxes that fall on his loading dock.

Great Prices. And A Great Machine.

We're committed to Zenith. And to you. the standard Zenith models that come from the factory are a good buy. And we'll be glad to sell you one at a price that will match anyone. But if you want the full potential, we'll fix it up for you. Larger disk drives. More memory. Internal tape back up. Removable winchesters. All at a budget pleasing price.

We make Zenith Scream.

When a new model comes out, we're among the first to get it. Our Zenith fanatics take it apart, prod it and poke it until it screams for mercy. Then we put it back together (usually with some new parts) so that it REALLY screams. With power.

You'll be back.

Check with us. Check elsewhere. We'll put together a custom quotation for you we GUARANTEE you'll find interesting. We think you'll come back to us. There's a reason for the growing list of companies, government agencies, and end users that know us as the source for Zenith. Your Way.

First Capitol Computer

1106 First Capitol Dr.
St. Charles, MO 63301

1-800-TO-BUY-IT (800-862-8948) Orders and quotes

1-314-946-1968 Technical support and order status

The StockSystem Challenge.

We will meet or beat any dealer's price on new stock Zenith systems. And for enhanced systems, no one can touch our combination of value, performance, and custom service.

Before you buy . . . . . .

Call Us First. And Last.

ZENITH | data systems

AUTHORIZED SALES AND SERVICE

How To Put Together A HUGCON

Example: CHUGCON

John Roach and Molly Jewell
Capital Heath Users' Group, Inc.
P.O. Box 16406
Arlington, VA 22215-1406

The Capital Heath Users' Group, Inc. (CHUG) has sponsored CHUGCON, a free educational computer conference, each fall, for the past four years. The conference has grown in size each year. The first conference, CHUGCON 82, was held in a high school. The tutorials were in classrooms, the vendors were in the cafeteria, and the dinner banquet was held in a nearby restaurant. The last conference, CHUGCON 85, occupied the entire second concourse of the Crystal City Hyatt Regency. Conference attendees came from as far away as Germany. The 44 vendors had a 13,450 square foot ballroom to call their own, while 3 adjacent seminar rooms handled 32 tutorials during the 2 day conference.

This article will summarize the way in which CHUG puts together a CHUGCON. The information was gathered from the 1985 subcommittee Chairs, but it could have been drawn from any past conference because the key to the success of each conference has been the enthusiastic participation of a large number of people who were not listed on the committees.

Conference tasks were handled by 7 subcommittees and a Conference Chairman. The leader or Chair of each subcommittee and the Conference Chairman act as the conference steering committee. The subcommittees are: CHUG Booth, Logistics, Meals/Registration, Publicity, Sessions/Speakers, Swap Meet and Vendors. The subcommittees operated independently up to a year in advance of the conference, with most of the work occurring in the last 4 months. Each task is a story in itself.

Conference Chairman

The first task of the Conference Chairman is to get all available volunteers acquainted with one another. Schedule meetings at a mealtime and make sure the table has enough room for notepads, as well as food. As work progresses, meetings with the Chairs of each subcommittee may suffice, but the key factor is communication. Volunteers need to feel needed or they will disappear. Regularly scheduled meetings and a constant flow of information will attract more participation and minimize delays. The role of the Chairman in all of this is comparable to leading a buffalo stampede: you can't do it by staying at home.

Planning a conference is a matter of defining what has to be done, when it has to be done, and then making sure that it will be done. Defining what has to be done is a matter for everyone to discuss to make sure that nothing is overlooked and that everyone has a chance to get involved. Actual deadlines will be dictated by advertisers, suppliers and the hotel. However, once the deadlines are set, the Chairman has to make sure that the preparations stay on schedule. Thus, the Chairman's role is that of a facilitator. When something is behind schedule, he or she should help to locate the resources to get the job done.

Site selection is the first task in planning a conference. Large meetings require lead times of several years. A small meeting has the same status in a hotel appointment calendar as a wedding party. Your best weapons in bargaining for space in a hotel are the attendance statistics from your last conference or a similar conference. History suggests that the person who locates a conference site usually ends up as the Conference Chairman. Notice that the word was "locate" and not "selects." The entire steering committee should visit each possible site and consider accessibility, cost of accommodations, available dates and possible floor plans. Participation of the entire committee in the site selection insures their continued involvement in the conference.

CHUG Booth

These are the friendly people who answer your questions or accept your CHUG membership application, if your visit to the conference prompts you to join CHUG. The booth is one of the conference control points for the exchange of messages and the distribution of volunteers to areas where they are needed during the conference. Each CHUGCON has increased the membership of CHUG by about 10% with most of the new members joining at the CHUG Booth during a conference.

The booth is equipped with back issues of >CHUG, copies of The Best of >CHUG, membership forms, and a bulletin board for messages. Volunteers man the booth in 1 - 2 hour shifts during the time that the display and tutorial areas are open. Large signs identify the booth.

The CHUG Booth Chair starts filling volunteer time slots months in advance of the conference. As the list fills, it is necessary to recontact the volunteers to insure that they will show up and to emphasize the importance of fully staffing the booth for the entire conference.

Logistics

This is the group that handles on-site crises. There will always be a need for a solution to a problem that was not covered in the planning of the conference. A blue trunk serves as the war chest for the Logistics Chair. It contains duct tape, scissors, cord, needle and thread, tacks, staples, screwdrivers, extension cords, cable adapters, markers and anything else deemed handy by the prudently minded.

Depending on the number of volunteers for each committee, this group may also assist in directing vendors from the loading dock to the display area, ferrying speakers from the airport to the conference, crowd control, locating lost audiovisual equipment or performing other timely miracles to insure that the conference goes as planned. The Chair of this subcommittee should be second in command at the conference and should be completely familiar with the plans of each of the other subcommittees and the conference site.

Meals/Registration

Selection of luncheon and dinner speakers are the first priority of this committee. Early selection will permit adequate advertising and improve meal attendance. A list of meal choices should be based on inclusive prices (taxes and tip included). Selection of the menu can be made as late as a few weeks prior to the conference, but early selection will permit the menu to be used in the preconference publicity.

The Registration subcommittee can consist primarily of people who have no skills in the use of a database. On-site registration is a team effort just like the CHUG Booth. However, pre-registration can be handled by one person with database experience.

A database is required to track dinner reservations, payments, ticket mailing and conference registration. All of this activity will show exponential growth as the conference approaches. The deadline for meal tickets has to be far enough in advance to ensure that the mail will reach the committee before the caterer must be notified of the final count.

It is easy to become confused over reservation numbers at the last minute if reservation exchanges are made through the committee. Leave the exchanges to the public bulletin board and concentrate on keeping an accurate count of attendance and meals.

The number of tickets printed will not be equal to the number of attendees, so buy a stamp to validate the tickets that are actually distributed. Number the meal tickets and log the number into the database, because some attendees will lose or forget their tickets.

The registration statistics are important in planning future conferences, but a lengthy registration form will induce people to bypass the registration desk and attempt to enter the conference. The name and address of an attendee is the starting point for advertising your next conference, but requesting other information may or may not be helpful. For example, data showing that attendees spent an average of 6 hours at CHUGCON 84 was useful in negotiating with the hotel in 1985, but the 1984 registration questions on computer ownership were not used.

Publicity

Lead time for free advertisements in national magazines is six months or more. Select the markets to be addressed and then start advertising early and repeatedly. You will probably want to try to reach 3 potential markets: vendors, distant attendees and local attendees. This will require 3 different types of ads.

Potential vendors need to be contacted directly, as soon as possible. Heath/Zenith vendors read REMark, BUSS and SEXTANT. Conference announcements in these magazines will produce more vendor interest than a direct mailing to every Heath/Zenith vendor in the United States. Get volunteers to distribute handbills and vendor registration forms at other conferences. Emphasize the nature of the display arrangements at your conference in these handbills and solicit vendor participation as speakers at your conference. These initial contacts should be followed up with information updates as the conference plans take shape.

CHUGCON regularly attracts attendees from Florida to New York. You should consider trying to attract attendees from outside your local area because it is not difficult. Potential attendees also read REMark, BUSS and SEXTANT, so your conference announcements in these magazines should include information about the proposed program. HAM festivals are good advertising points and HAMs love to travel. Identify the HAMs in your organization and equip them with handbills. Local and national bulletin boards are another point of distribution for conference information. Use available contacts across the country to upload information to local boards everywhere. Repeat these bulletin board uploads each time a new information update is available. Run full page ads in your newsletter and supply photo ready ad copy to newsletters on your exchange list. Make sure the ads are eyecatching.

Local advertising is potentially expensive, so be selective and timely in placing your ads. Send material to local bulletin boards and newsletters frequently. Use all free local advertising available, including public service radio announcements and the events section of the newspaper. Paid newspaper advertising in the last two weeks before the conference is effective, but expensive. Purchase affordable ads with high visibility and wide distribution. If there is a local newsletter in your area written for the computer community, it is a good place for repeated advertising. For example, in the Washington area, PC-ads is a computer tabloid distributed by subscription or bookstores. They not only

ran our ads, but they came to the conference and wrote a really nice article about it.

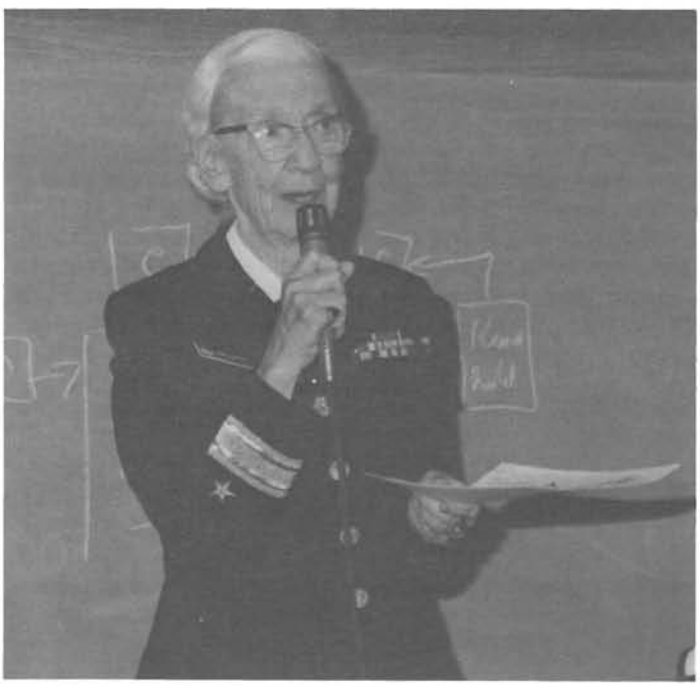
Sessions/Speakers

Speaker topics are a significant factor in conference attendance. But when you start planning the conference, you probably do not know who will be your speakers. Block out time slots for a tentative number of sessions, consider possible topics and then start looking for speakers. Start in your own group. Offer a complimentary banquet ticket or a membership renewal as an inducement to speak at your conference. Some vendors may want to describe their products. Sign them up to speak and locate a babysitter for their booth while they are speaking. Be helpful; these people are your program.

Once you have part of the program ready, advertise it. This will get more speaker participation. Try to organize the sessions into cohesive themes when you have enough speakers. This will also attract more speakers. Be flexible in your planning. The original speaker plans for CHUGCON 85 called for 15–18 speakers and we ended up with 32.

The Sessions/Speakers Chair tracks session arrangements with a database. The data include speaker, title, topic, location, time, equipment needs and name of volunteer assigned to introduce the speaker.

The assisting volunteer is as important as the speaker. First of all, speakers appreciate being introduced rather than left to fend for themselves. Secondly, computer talks may require special equipment. The volunteer can be given the task of taking care of the equipment and speaker for just one talk. This will insure that the talk goes all right without tying up the volunteer for the entire conference.



Commodore Grace Hopper passed out nanoseconds and a memorable string of observations to make the dinner banquet the high point of CHUGCON 85.

A speaker ready room or "Green Room" should be available in the sessions area. It is the second conference control point and serves as the base of operations for the Sessions/Speakers and Logistics Chairs. A telephone is available for emergency calls. The speakers are provided with this telephone number and the location of the Green Room. Light refreshments are on hand for speakers and overworked volunteers. Special equipment and backup equipment are stored here between talks. Volunteer speaker escorts are introduced to speakers at the Green Room. Large copies of the program are posted in this room and on the door of each seminar room to aid in tracking the talks. Late program changes are transferred from the Green Room copy to the other large program copies.

Copies of all speaker correspondence are held by the Sessions/Speakers Chair. The easiest way to do this is to use a looseleaf binder with dividers for each speaker. A published program for the conference should include a synopsis of each talk and background information on each speaker along with the title of each talk and the name of each speaker. All of this information can be obtained with a speaker questionaire and placed in the binder.

The initial contact, a follow-up questionaire and information updates are usually the only written correspondence with the speakers. Generous use of the telephone will usually convince hesitant speakers to participate and will straighten out last minute problems better than a letter. The Chair of this subcommittee can expect a phone bill that is close to the size of the Vendor Chair's phone bill.

Swap Meet

Be sure to mention the swap meet in all ads about the conference. Advise attendees to bring a shopping list to the conference. A telephone number for swap meet information appears to be useful because it usually turns up more volunteers than equipment for sale.

There are 2 swap meet areas, CHUG and independent. The CHUG area is managed by volunteers as a service to members. Attendees who are not CHUG members may reserve a table in the swap meet area, but they will have to supervise their own items.

Managing the CHUG swap meet area is another task that requires a team effort during the conference. The Chair draws up a list of volunteers to take turns managing the CHUG area. CHUG members who wish to offer items at the swap meet should volunteer to help. If volunteers are willing to help, but cannot man the swap meet area, they help set up the area at the start of the conference. There is plenty to carry and they may change their minds about helping to manage the area.

An identification tag is attached to each item brought to the swap meet. A card identifying the item, its owner, a desired price and the condition of the item is left with the manager on duty at the swap meet. An electrical outlet is available for equipment trials.

Vendors

Vendors incurred no fee for tables or floor space at CHUGCON 85. The hotel did charge for power, phones, and security. CHUGCON asked for a refundable participation deposit to ensure vendor interest.

A standard size booth was 80 square feet. Space was allocated on a first come, first serve basis. Vendors could obtain more space by requesting it in their application.

Most vendors donated door prizes at the conference. Receipts identifying the recipient of each door prize were returned to the vendors for their records.

The Vendor Chair is responsible for: vendor relations, vendor area setup, and door prizes. The subcommittee manages this information using two special forms and a system of database files and programs. The two forms are the Vendor Participation Reply form and the Door Prize Receipt form. These are shown in Figures 1 and 2.

```
YES! I INTEND TO PARTICIPATE IN CHUGCON 85 ON BOTH 2 AND 3
NOVEMBER, 1985! AND.

____I WISH TO GIVE A PRESENTATION ENTITLED_____________
_____________________________________________________

____I WISH TO USE THE CONFERENCE THEATRE
____I WOULD LIKE MORE INFORMATION ON HOTEL ACCOMMODATIONS
____I WOULD LIKE TO PARTICIPATE IN THE SWAP MEET
____I WOULD LIKE TO DISPLAY IN THE VENDOR AREA
    (Space allocated by date of postmark on request)
NAME_____________________________________________________
TITLE____________________________________________________
COMPANY__________________________________________________
ADDRESS__________________________________________________
_____________________________________ZIP_____________
Description of products__________________________________
Electrical requirements (# outlets and wattages)_________
_____________________________________________________
```

Figure 1.
Vendor participation reply card distributed in conference advertising.

```
                    RECEIPT
_____________________________________________________
DESCRIPTION.
_____________________________________________________
_____________________________________________________
DONATING VENDOR__________________________________________
Address._________________________________________________
City/State/Zip___________________________________________

CHUGCON PICK^UP OFFICER

_____________(print name)_____________(Signature)_____
WINNER'S NUMBER:_____________________
WINNER'S NAME:___________________________________________
WINNER'S ADDRESS:________________________________________
_____________________________________________________
OTHER REMARKS AS NEEDED:_________________________________
```

Figure 2
Door prize receipt form.
Completed copy is returned to donor.

Information from the two forms is placed in database files. The 3 types of reports needed from the vendor database are mailing labels, a complete file dump and vendor status on particulars, such as receipt of electrical, security and refundable deposit checks.

Vendor publicity, registration and notification are the primary components of vendor relations. The three vendor mailings correspond to these functions. Two and one half months prior to CHUGCON, the subcommittee mails a general notification to potential vendors. Based on replies to the first mailing and other sources, the second mailing to interested firms provides vendors with specific information. After the registration deadline, 2 weeks prior to the conference, the third mailing confirms vendor status including tentative location in the vendor area.

Complete and accurate area layout is another important part of a smooth-functioning vendor sales operation. The Vendor Chair prepares and refines a floor plan for the vendor area which includes the space, electrical, security and other special requirements of each vendor. The vendor area layout must contain sufficient detail to allow the hotel to position partitions, furniture and electrical hookups. This floor plan must be approved by the Fire Marshal, so it should also indicate the width of all aisles and the location of every exit. A computer aided design program produces the vendor area layout for CHUGCON. This delights the hotel and simplifies last minute changes to the plans.

The Chair and subcommittee oversee the hotel conference set-up and supervise the entry of the vendors and their equipment into the sales area. Access to the vendor area must be limited to vendors until they are ready for the entry of the general public.

The Chair is responsible for the acquisition, distribution, accounting and reporting of door prizes. He encourages all vendors to donate prizes, and ensures that each prize has a receipt slip. At the appropriate time, he arranges for prize distribution, including drawing winner's names and completion of receipt forms. After the conference, he notifies each donor of the identity of the winner of their prize. The list of donors prizes and recipients is then passed to the >CHUG editor for publication.

Early publicity and vendor commitment are the keys to the success of a vendor exhibit. They free the Vendor Chair to attend to the inevitable last minute details without worrying about late registrations and related problems. Additionally, the database system devised to support the vendor chairman is of inestimable value in ensuring the accuracy of a multiplicity of details.

Conclusion

Conferences start as an idea and grow to reality because of the efforts of a large number of people. The subcommittee Chairs must understand their assignments and be able to work independently on their tasks, because they are the focal points for a great deal of activity. For this, they get their name on the program. However, the success of the conference ultimately depends on the volunteer workers who didn't want their names on the program.

Acknowledgment

Special thanks to the subcommittee chairs for their input in the preparation of this summary: CHUG Booth, John Fuller; Logistics, Bob Shon; Meals/Registration, Carolyn Johnson; Publicity, Douglas Campbell; Sessions/Speakers, Julia Broetzman; Swap Meet, Bruce Gewirz; Vendors, Gary Jewell.

About The Authors

***John Roach**, veteran of CHUGCON 83, 84 and 85; 1985 CHUG President; current editor of >CHUG, the monthly publication of the Capital Heath Users' Group, Inc.*

***Molly Jewell**, veteran of CHUGCON 82 and 85.*

*

Search For A Language Part II

Leo R. Hansen
6771 East 4th Street
Tucson, AZ 85710

Honeymoon Is Over

In the first article (October 1985), I presented my reasons for choosing IBM Pascal over other candidate languages, primarily it seemed like the best balance between what I wanted and what was available at the time. However, as always, there are things that you do not find out about until you actually get to try the system yourself.

The first thing that hits you right between the eyes, since you can see it when you list the directory, is the size of the compiler, linker and library files. The compiler is 180,608 bytes long, however, it is in two files. PAS1 and PAS2, which are 81,408 bytes and 99,200 bytes, respectively. The linker is 41,856 bytes long and the Pascal library is 83,456 bytes long. There are also four other files that are needed to support Pascal which total another 17,996 bytes. These are ENTX6S.ASM, FILKQQ.INC., FILUQQ.INC, and PASKEY. The total number of bytes then required to support the Pascal language is 323,916. Then, if you would like an editor of some sort and print utility, you are out of luck. You are out of luck just with the Pascal files. The remaining disk space is insufficient to do anything practical. So, the first thing you do is try keeping your source and all temporary files on another disk. Storage requirements like this make a single disk system very difficult to use, if not impossible.

After fiddling around and running out of disk space a few times, especially if you let the compiler generate the object listing (the assembly language version of your program), you finally quit fighting it and decide to set up some system that will have plenty of room and will make some sense, so you can follow it when you are half asleep.

Development Disk Configurations

I finally settled on a four disk system, this was dictated by the storage available on my H-120, i.e., two 320KB disks (I'm still with ZDOS version 1.25). On Disk 1, I have the editor, some utilities (CHKDSK, DSKCOPY), a print program, a date and time update program, and the source file of the program I am working on. The editor is fairly large weighing in at 55,040 bytes, however, it is worth every byte. I have used numerous program development editors and word processing editors and this beats them all going and coming. It is "Extended Screen Editor" by Cherry Engineering and I recommend it without reservation. It is a what-you-see-is-what-you-get editor, which can handle files of any size limited only by the disk storage available. It is designed specifically for the Z-100 computer family and is very easy to configure with the various function keys and keypad. By keeping the source file on the "editor" disk, I have a back-up in case I get carried away cleaning up the source file disk.

On Disk 2, I have the Pascal compiler, both passes, the support files ENTX6S, FILKQQ, FIKUQQ, PASKEY, and some utilities again. On Disk 3, I have the Pascal linker and library files also with some utilities. On Disk 4, I keep the source files and all the temporary files that are generated by the compiler. The run time program is also sent to this disk. All of the above disks are used in a system which utilizes single letter commands to perform editing, file transfer, compiling, linking and execution. These single letter commands are obviously batch files. The first "BAT" is the edit bat which is: ESE WORK.PAS. This will call up an existing file called WORK.PAS or if not existing, create one at your option. After completing preparation of the source file, I then call on T.BAT to transfer the WORK.PAS file to device B which contains Disk 4, the work disk. T.BAT looks like this:

```
DEL B:WORK.PAS
COPY WORK.PAS,B:WORK.PAS
DIR B:/W
```

After the first line has been executed, you are slightly vunerable since you have wiped out the source file on the working disk and only have the source on the editor disk. Something could happen to the editor disk and then you would have lost your source file. A safer way would be to rename WORK.PAS to WORK.NEW before copying to B: and then delete WORK.PAS from B: and rename WORK.NEW to WORK.PAS. I call the DIR command just to see if everything that I expect is there on the work disk. At this point, I remove Disk 1, the editor disk, and insert Disk 2 in drive A. Disk 2 is the compiler disk and is invoked by typing C (this calls C.BAT) which looks like this:

```
B:
A:PAS1 WORK,WORK,CON:WORK;
A:PAS2
A.
CHKDSK B:
```

I change the default disk to B so that all the working files are sent to B which has plenty of room on it, and also I don't like writing to my system disk when I don't know how big the files are going to be. This is especially true if you are creating source listings, object listings and object code files. If you are just using an editor to create a file then run out of disk space, it just tells you to put in another disk and will continue where it left off. I also call CHKDSK after all this is finished, but not just to see how much disk space there is. More on this when I discuss the surprises below.

I now remove Disk 2 from drive A and insert Disk 3 into drive A. The linker is invoked by typing L calling up the L.BAT file which is:

```
B:
A:LINK @A:LINKWORK
A
```

Again, I change the default disk to B: so that all files go to drive B which is where the space is. I now type in R for R.BAT which looks like this:

```
B:WORK
```

This executes the WORK.EXE file that has been generated.

So, by using the BAT file commands you can edit, compile, link and execute a file by typing four single letter commands. If you write error free programs like most of us Zenith/Heath programmers, you can combine these into one BAT file that will do the last three in a single command by merely combining all the commands into one file.

Surprises

I mentioned that I included a call to CHKDSK at the end of the compile, but not just to see the memory or disk space utilization. Before I achieved the ability to write error free code, I would generate numerous errors during a compile. This was no big deal because I would merely switch disks and correct the error and run again. However, one day this message appeared that said I was out of disk space. This was impossible, I was only working with little files and after all I had 320KB storage on the disk. I ran my directory program which lists the free space, and lo and behold it said free space was 0. No problem, I'll just delete a few unwanted files and be on my way. So, I deleted a few unwanted files and I'm on my way. On my way until I need to write to the B drive again. POW!! No free space. Hay! What the @#%& happened, I just made room and there still is no room at the inn. Out of desperation, I run CHKDSK to see if my disk is okay or what. Walla, all the disk space I knew was there still is. What apparently happens is, since the files do not have to be in one block there are pointers to each succeeding block. The more fragmented your disk becomes, the more pointers there are until finally you have insufficient space and the free space goes to 0. CHKDSK combines all your program fragments and reduces the number of pointers required, thereby freeing up all that space that you knew was there all the time anyway. Hence, that is the real reason I have a call to CHKDSK at the end of the compile.

The next problem that I had was with the object code listing. I couldn't shut it off. If you want to see something eat up disk space, compile a 500 line program and send the object listing to a disk. The IBM Pascal book indicated that one of the listing options was NUL.CODE, but this didn't seem to work. After much digging, I discovered that the compiler directive that controls the object code generation defaults to on, so I turned it off with (*$OCODE*). Now all I get is an empty file called WORK .CODE.

The next surprise was more of a disappointment than a surprise. This was the relatively poor treatment of separate compilation using UNITS. The examples are more like templates to follow blindly without a lot of explanation as to what you are doing or why. After considerable blood, sweat and tears I have finally managed to handle separate compilation, either with Assembly language or Pascal modules. This leads directly to the next topic: Recommended Books.

Recommended Books

The IBM Pascal book is called: "Pascal Compiler". This is a trimmed down combination of Microsoft's Pascal Reference Manual and User Guide books. In some areas, the IBM version is very good, definitions of the language, getting started and how to do basic things. The advanced items are not covered very well or at all in some cases, such as using the system functions and procedures. The index is terrible, things are not where you expect them to be, the categories they are grouped in are hard to follow until after you have been through it a few times. Then, when you are cold on the subject, you can't remember where it was, which makes it worse because you know it's there, you've seen it before. However, since you get the book when you buy the language, keep it for reference or a paper weight. Two other books that compliment the IBM book are: 1) "Pascal For The IBM PC" by Kevin W. Bowyer and Sherryl J. Tomboulian, published by Robert J. Brady Company, and 2) "Programming The IBM Personal Computer: Pascal" by Neill Graham, published by Holt, Rinehart, and Winston. Both books are very good as far as they go, which is not quite as far as the IBM book. However, the coverage up to that point is better than the IBM book because of the numerous examples that they both use. In that regard, the use of examples, they are excellent. As companion volumes they are very complete. The problem is, the advanced capabilities of the language, i.e., separate compilation is touched very lightly. To date, I have not found a really good book that covers the advanced features of the language. Another Pascal related book that is very good is "Software Tools In Pascal" by Kernighan and Plauger, published by Addison-Wesley Publishing Company. A lot of the utilities in MS-DOS 2.0+ are detailed in this book, so if you want to customize those kinds of functions, this book is an excellent reference.

The Assembly language side of the house is similar to the Pascal side, i.e., only a few really good books. The best, bar none, is "IBM PC Assembly Language" by Leo Scanlon, published by Robert J. Brady Company. This is a well written book with good examples and explanations of what is happening and why. I recommend this book without reservation. The only other Assembly language book that I find useful is "The 8086/8088 Primer" by Stephen P. Morse, published by Hayden Book Company, Inc. This book fills in the very few areas that Scanlon's book is light on. A general book on the 8086/8088 that is very good even though it is not very big, and is not a language reference manual, is the INTEL book, "An Introduction To ASM86." This book tells a lot about the why of the compilers and the general memory organization of the processors. Again, an unconditional recommendation.

I have yet to find a book that does a really good (deep) job on MS-DOS. The Zenith documentation is very comprehensive,

but very short on examples on the advanced topics, but is way above anything on the market right now. I can't even recommend a poor book on MS-DOS.

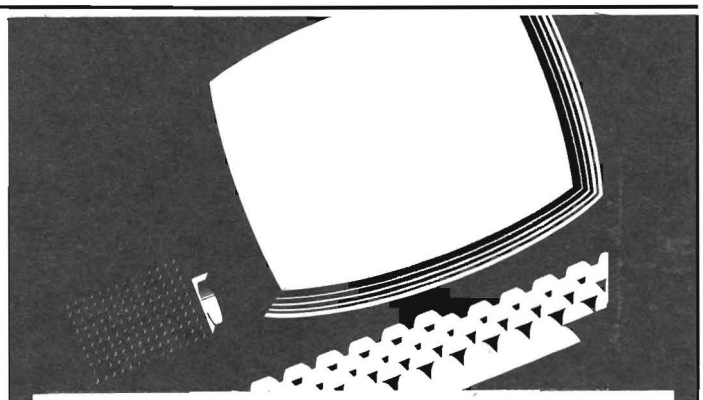
Do It Again?

Would I select IBM Pascal again? At this point, I probably would. It fulfills my hardcore requirements and has some nice capabilities that I wanted. Since I started this article, I have switched from the 48 tpi drives that are standard on the H-120/Z-120 to 96 tpi drives. A lot of the inconvenience of switching disks is now gone. I now have all the Pascal files on a single disk and keep the source and work files on a single disk. The 8 MHz speed-up kit and the high capacity drives make the H-120/Z-120 a really nice machine. There are two other languages (one really) that I would consider the next time around. These are Microsoft Pascal (IBM Pascal is a subset of this one) and Modula-2. There are some additional capabilities that Microsoft Pascal has that were taken out of IBM Pascal that would be nice to have. Four of these are: 1) double precision integer and real numbers, and decimal arithmetic with precision to 14 decimal digits, 2) direct control of interrupt service programs, 3) concurrent programming, and 4) program overlays. Modula-2 also supports some of these same capabilities and would, therefore, be considered for the same reasons as Microsoft Pascal, the next time around. The only drawback that I am aware of at this time is that some implementations of Modula-2 are in P-code which does not interest me.

At any rate, these were some of the trials and tribulations that went into my selection of a language. I hope some of it is useful to anyone still considering which language to use.

Happy Languaging.

*



FIRST ANNUAL

MAY 2nd, 3rd, and 4th

at HENRY VIII INN & LODGE

Bridgeton, St. Louis County,
Missouri 63044

OR MORE INFORMATION CONTACT

MIDWEST HUGCON
HEATH/ZENITH Computer Center
3794 McKelvey Rd.
Bridgton, MO 63044
(314) 291 1850
(VENDORS Contact Steve Hart)

Sponsored by the
St. Louis Heath
Users' Group

SLHUG BBS
(314) 291 1854
(MODEM)

HERO 2000

Jim Blake
Eastern Regional Manager
Educational Sales
Veritechnology Electronics Corp.

At 33000 feet aboard United 221, the captain comes on the phone and gives his normal speech, but adds "oh, and welcome aboard to first class passenger, HERO 2000." (He didn't mention MY name!)

Yes, Heath Company's new ROBOT was about to make his first public appearance in New York. Actually, he isn't a robot . . . well he is, but robot is easier to say than Intelligent Machine. Intelligent, because he can accept a variety of inputs, evaluate them, make an appropriate decision and react accordingly.

Thousands are familiar with HERO I which was introduced three years ago as, 1) an educational tool to teach electronics and robotics and, 2) an entertaining toy for the person that needs to be the first on the block to have a robot that can sing and dance. HERO I out sold all other robots in the world combined. It generated several million dollars in free advertising while appearing on the Today Show, Johnny Carson and almost every talk show in existence. He even testified before a joint session of congress! And sales are still strong.

Here then, is HERO 2000, and what makes it unique.

HERO 2000 is an Intelligent Machine Learning System on wheels with some remarkable capabilities. (I can editorialize, can't I?) It is controlled by an 8088 16 bit microprocessor at 5 Mhz, but relieved of its I/O duties by 11, count 'em, 11-8042 slave microprocessors. Robot movements are powered by 8 closed loop DC servomotors. The fully articulated arm has 5 axes of motion and can lift over a pound in any orientation. The gripper rotates 360 degrees and bends 270 degrees at the wrist. A patented 'sense-of-touch' system adjusts the gripper force. (HERO, please don't squeeze the Charmin.) A 64k monitor ROM contains a specially enhanced version of X-Y BASIC, a direct text-to-speech conversion utility, 6 demonstration programs, diagnostic routines and a debugger. 24k RAM is standard, expandable to 576k. User interface is either through the head mounted hexidecimal keypad and seven segment LED readout or through the full ASCII keyboard on the Remote Console from as far away as 100 feet.

When the machine is powered up, HERO checks his vital signs and if everything seems healthy, you are asked to input the TIME, DATE, DAY OF WEEK and TEMPERATURE. Until powered down, HERO remembers all this and automatically adjusts for leap year and, optionally, daylight savings. The temperature will be reported to within one degree relative to what you told him. If you do nothing else, HERO will sit there and periodically say "hello" to remind you that he is waiting for a command and that his battery will expire in about 6 hours. Pressing the 'sleep' button will prolong battery life to 6 days. Optionally, the battery could jump start your Toyota!

CONTROL OF THE MACHINE: Let Me Count The Ways

Most of the keys on the head are dual function. In the MEMory/DEBUG mode, you basically have control at the machine language level, a familiar environment for HERO I users. In addition to examining/changing memory locations, you can change the baud rate of either of the two RS-232 serial ports, run selected programs, INput and OUTput bytes to a port and check memory. Exiting this mode, the user can do one of five things:

1. Move the Robot forward, backward, turn left and right in any of seven speeds from a snails pace of 3.8 inches per second to a brisk 16 inches per second.
2. Boot from an optional built-in disk drive. (MS-DOS)
3. Load a previously saved program from the cassette port.
4. Enter the TERMINAL mode, in which case, HERO expects to find either a terminal or any computer with a communications package such as CPS, ZTERM, etc. at his RS-232 serial port.
5. Enter the REMOTE mode, where HERO starts sending signals over his 74 Mhz RF link looking for his companion Remote Console.

In all modes, HERO 2000 BASIC automatically signs on and announces the amount of free bytes followed by the familiar 'Ok' prompt.

Let's examine the REMOTE mode first, since the Remote Console is standard. The hand held console has a standard ASCII keyboard, a two line, 40 character LCD display and some special function keys that allow you to hide behind posts and drive him around airports and make smart remarks to pretty girls. In fact, I

drove him in the hotel lobby, credit card in hand and they actually checked him in and gave him a key. This can be dangerous, however. Riding down the elevator one sleepy morning, the elevator stopped, door opened, I pushed the forward button, HERO got off with a lady, the door slammed shut, and 30 seconds later I was on the 41st floor! Problem is, I didn't notice what floor he had got off on! Well, I found him on the 10th floor yelling, "I have lost contact with my Remote Console" . . . and trying to make a date with the lady.

At the console you can control the machine in four ways.

Command Mode

Here, you just type in the command and get an immediate response. Here are a few examples.

PRINT TEMP	The current temperature is displayed on the screen.
BASE 20	The machine travels 20 inches forward.
LFT 5.1	The left drive motor travels 5.1 inches so that he turns to the right.
ARM 45.3	The arm extends outward to 45.3 degrees.
FORCE 0	The gripper closes to a tender touch.
FORCE 8	The gripper closes with about 3 pounds of pressure.
SAY "hello"	HERO speaks.
SLEEP 0,10,0	Rest 10 minutes (up to 45 days)
ELBOW 90	The elbow extends straight out.

Program Mode

BASIC users know that you can write a program using line numbers, save it to cassette or disk and RUN it. For instance:

```
10 SAY"Here I go".' say something smart
20 BASE 30:' go forward 30 inches
30 ELBOW 90,7:' extend elbow to 90 degrees at speed 7
40 ARM 45.5:' extend arm to 45 degrees at speed 5
50 IF SONARB<12 THEN BASE -10:
   ' see if something is in front of me
60 IF SOUND>10 THEN SAY" I MUST HAVE BUMPED INTO SOMETHING"
70 TORSO 100:' turn the torso right to 100 degrees
80 LPRINT "I'm getting tired of this foolishness":
   ' Print the statement on virtually any printer
```

If this simple program were RUN, all functions would occur simultaneously. More on this later.

Teach Mode

There are two ways. If you type TEACH on the keyboard, you can drive through airports, but he doesn't remember being there. (I've had that problem myself.) Airport security guard to a colleague: "How do you frisk a robot". Colleague: "I don't know, I'll ask Eddy."

However, if you type TEACH 10, you can teach HERO to perform a task, and he'll remember it. For instance, you want to teach the machine to repeatedly pick up parts from a shute and, depending on their size, place the parts in separate stacks. Using the special function keys, you guide the machine through the necessary movements to perform the task satisfactorily. In the process, however, you will probably make some mistakes. But, listen to this. The machine not only generates the BASIC code to perform the task (beginning at line 10, because we said TEACH 10) but it FORGETS your mistakes! Since it wrote the program for you, it is not only a great time saver, but educational as well. How about a machine that teaches you how to program in BASIC?! Of course, you can modify, enhance or add speech and conditional statements to the program simply by typing them in.

Terminal Mode

In the terminal mode, HERO can be connected to a terminal through the RS-232 serial port at baud rates from 45 to 38400. This is the most desirable method of programming HERO, since you can see 24 lines of your program at a time and at a nice crisp baud rate, typically 9600. The Remote Console can act as terminal too, but it is restricted to 2 lines at 40 characters and 600 baud is maximum.

Link Mode

The Remote Console can be used as an RF link between HERO and your terminal or computer as far away as 100 feet. For instance, you could be in another room working on an Apple connected to the Remote Console talking to HERO.

Environment Reports

HERO 2000 is very aware of his surroundings.

Sonar

There are two ultrasonic transducers that detect objects within a radius of 127.5 inches with an accuracy of .5. One is mounted in the front of the base. The other mounted on the head that scans a full 360 degrees. It takes readings at 15 degree intervals and is software controllable by the user. Normally, it continually scans 360 degrees, but you can cause it to stop at any of the 24 available bearings. A report from this port is obtained by the keyword SONAR(n) where n is the bearing, 0 being right in front of the Robot.

As an example, the following program is handy for changing the color of hotel maids.

```
10 IF SONAR(0) > 60 THEN 10
20 SAY" WOULD YOU PLEASE HELP ME FIND MY PANTS"
30 GOTO 10
```

The keyword SONARB returns the distance to an object from the base sonar.

Light

The same rotating reflector that is used for the head mounted sonar also detects and quantifies ambient light levels over the visible spectrum. It has a resolution of 255 levels. The keyword used is LIGHT(n) where n is the bearing.

Sound

A sound sensor detects ambient sound within the range of normal human hearing with a resolution of 255. The keyword here is SOUND.

Temperature

The specs say that this sensor measures temperatures between 60 and 90 degrees F. It's actually much greater than that. The keyword TEMP will return the temperature within 1 degree.

Battery

The keyword BATTERY will return a relative battery condition with 100% being full charge.

Motor Position

Here is another thing that makes HERO an Intelligent Machine. As mentioned, there are 8 CLOSED LOOP DC servomotors, each with its own 8042 microprocessor that control the Robot.

The Robot travels about on four wheels. One spring loaded free swiveling front wheel, so that he can negotiate small changes in

the terrain and a similar trailing wheel, though not spring loaded. In the center, one left and one on the right side are two motor driven rubber wheels that give HERO a pulling force of 26 pounds. (Trailer hitch not provided.)

If you command the Robot to go forward 20 inches, the 8088 loads the two slave microprocessors (one dedicated to each motor) with a count, and power is applied to the left and right base motors. Infrared opto-isolators mounted on the motors send back pulses that decrement the count. When the count reaches zero, HERO has arrived at his destination. But what if something gets in the way? Well, if it's a brick wall, and a slick floor, he'll spin his wheels until the count is zero. However, the programmer can interrogate each port to see if there is a discrepancy in the count between the drive motors or, he can use the command MPOS(BASE) which should return to 20, if the Robot traveled 20 inches.

This command, MPOS(xxxx) is valid for each of the 8 motors. You want to know where the ARM is? MPOS(ARM) will tell you. Another example: IF MPOS(GRIP)=0, then HERO's hand is empty! Neat.

This is a good place to amplify on something mentioned earlier. Consider the following few lines of code.

```
10 BASE 10 :'            go forward 10 inches
20 TORSO 45 :'           turn torso right to 45 degrees
30 ARM 90
40 ELBOW 90 :'           extend arm and elbow straight out
50 PITCH 90 : '          gripper straight out too
60 ROLL 180  '           turn his gripper
70 GRIP 9 .'             open gripper
80 PRINT TEMP
90 RHT 20:LFT -20:'    do a 360 in his own diameter
100 SAY"I DID ALL THESE THINGS SIMULTANEOUSLY"
```

When this program is RUN, each command is serially 'squirted' out to the respective slave microprocessor so that all the commands are processed at the same time.

But suppose two robots are working side by side simulating an automated assembly line? There may be an occasion where they could get in each others way. Using the sonar or some other means, you may not want a command to be executed until some condition is satisfied. There are a couple ways to do that and one is to modify one or more of the above statements like so.

In the program above, if line 20 were modified to read:

```
TORSO 45,5,$
```

then the TORSO would turn to 45 degrees at speed 5, BUT because of the dollar sign, the rest of the program would not be executed until the TORSO had completed the turn.

HERO 2000 BASIC is rich with all the commands you would expect to find in the most popular BASICs, plus some special ones needed to teach or learn about automated intelligent machines. In addition to those already used or mentioned:

- There are 9 different ways to dissect the keyword TIME(n) so that you can get day of week, date, milliseconds and so on.
- CALL is provided so you may call assembly language routines.
- CON# and ASSIGN allow you to change control of the print and console device on the fly.
- DELAY minutes, seconds, milliseconds
- ENABLE line number, port, condition, mask. A way to interrupt normal program flow.
- EXEC address. Allows BASIC to access another program.
- IOBYTE, PEEK, and POKE allows the user to get himself in trouble.
- SCALL performs a FAR call to a user subroutine and pass parameters.
- SENSE allows you to test a bit on a port.
- TLOAD and TSAVE. To load and dump to cassette.
- WRIST pitch, roll, speed moves the gripper in two axes in one command.

Speak To Me

HERO 2000 can talk, sing and do sound effects. And HERO I users will love this. To make the Robot talk, you just type SAY "HELLO I AM HERO 2000" and the built-in text-to-speech converter will look up the proper phonemes, put them together, and in the normal robotish monotone voice, he will say "HELLO, I AM HERO 2000." However, you can change the attributes to make the speech more natural. Indeed, there are 4096 different combinations. You can change the Inflection, Duration, Rate, Amplitude, Articulation Speed, Frequency, Note and Mode. You can imbed these attributes right in the sentence, ie: SAY"[2001] HELLO I AM HERO 2000" would cause HERO to speak in a high pitched voice. Or, you can use the actual phonemes by using the SPEAK statement. For instance, SPEAK" hf e r o w t u u1 th ah1 w z ae d pa" would cause HERO to say 'HERO 2000'. Again, attributes can be imbedded between the phonemes to make him sing or whatever.

Another tool is the PRINT TEXT$ command. If you don't know what phonemes to use, you can type PRINT TEXT$("HERO TWO THOUSAND") and the actual phonemes are printed out for you to use as is, or edit.

HERO 2000 and the Remote Console come with their respective battery chargers and later this year, an Auto Docking accessory will be available. When HERO detects that his battery is getting low, he will be able to look around the room and locate his battery charger, navigate to it, turn around and back up and automatically connect. From full dead, it will take about five hours.

Inside the Robot, the electronics are arranged on plug-in cards. There are a total of 12 slots on the backplane. In the basic configuration, 4 of these slots are in use. An accessory available now, is an experimenters board with fully buffered access to the buss and a removable solderless bread-board for user developed circuits. Add-on boards from other vendors are sure to be plentiful, since several of the folks that developed special applications for HERO I were given a sneak preview along with all the technical data back in November.

On the front of the head is a series of status lights and 8 user definable LEDs that give HERO that robot look. Such things as Transmit and Receive, Cassette, Power, Low Battery, Float and Voice are reported. You can type DISPLAY 7,TEMP for instance, and a binary representation of the temperature would be written to the 8 user LEDs. Oh, and you can write to the seven segment LEDs, too. Type DISPLAY "LIGHT" and the string LIGHT is automatically converted so that the proper segments are lit to display the word.

Courseware

The machine comes with a comprehensive users manual, plus the BASIC manual and a detailed technical manual with circuit descriptions (a course in itself) and schematics.

Intelligent Machines is the first of three optional courses supporting HERO 2000 that examines the latest hardware and software used with computer-controlled machines. The material stresses application, as well as operation. Fifteen experiments using the Robot reinforce the concepts presented. It is available in both classroom and self-study formats. The classroom edition includes the textbook with examinations, student workbook, instructor's guide, experiment parts and a cassette tape of robot experiment programs. The self-study format includes a comprehensive text with examinations, plus experiment parts and a robot program tape.

Subjects covered include the definition of intelligent machines, artificial intelligence, decision making, communicating with machines, various data input techniques sensory input devices, output, and the major components of robots as used in industry.

Electronics For Automation covers 7 major areas and has 13 reenforcing experiments. Subjects covered are: industrial controllers, passive sensory devices and circuits, active sensory devices and circuits, signal conditioning circuits and machine control circuits and applications.

Programming And Interfacing gets right down to the nitty gritty of the robot and guides the user through the internal workings. Detailed technical discussions are supported by 17 hands-on experiments.

HERO 2000 with battery chargers and Remote Console is $4500. Educators can call their Educational Representative to see a video tape of HERO in action or an on-site demo.

This is the first in a series of articles about HERO 2000 written by educators for use in the education of Electronics and Robotics.

The Education Representatives are:

Technical Training Aids
Curt Baggett
P.O. Box 20042
Birmingham, AL 35216
(205) 822-1326

Pete Peterson Associates
Pete Peterson
P.O. Box 2047
8614 E. Columbus Avenue
Scottsdale, AZ 85251
(602) 945-6605

George L. Oliver Company
George Oliver
P.O. Box 1842
44834 Grimmer Boulevard
Fremont, CA 94538
(415) 651-6720

RANESCO
(R.A. Nieminen Educational Sales)
Robert Nieminen
P.O. Box 698
San Juan Capistrano, CA 92693
(714) 493-8040

Barron Farquhar Ltd.
Barron Farquhar
709 W. Littleton Boulevard, Suite 25
Littleton, CO 80160
(303) 795-1450

S & H Associates
S. Shimabukuro
458 Poipu Drive
Honolulu, HI 96825
(808) 524-2453

Midwest Edusystems, Inc.
Henry Stone
2701 Grand Avenue
Galesburg, IL 61401
(309) 343-6136

Ron Bales, Inc.
Steve Pearson
P.O. Box 546
907 Commercial Street
Emporia, KS 66801
(316) 342-5236

Peter Mackesy & Associates
Peter Mackesy
6 S. Washington Street
Norton, MA 02766
(617) 285-9028

Howard & Smith, Inc.
Jerry Dwyer
P.O. Box 457
Royal Oak, MI 48068
(313) 585-2300

EDUTEC, Inc.
Vince Pecoraro
7070 Brooklyn Boulevard
Minneapolis, MN 55428
(612) 533-0000

HAAS Associates
Hy Haas
P.O. Box 1032
St. Charles, MO 63301
(314) 724-9584

Wired Kits Corporation
Morton Sugarman
P.O. Box 278
16 Sagamore Road
East Norwich, NY 11732
(516) 922-6590

Carolina Training Associates
Bill Johnstone
P.O. Box 816
Sanford, NC 27330
(919) 776-8161

SIMC Training Systems
Ed Simms
3749 Dewlawn Drive
Toledo, OH 43614
(419) 382-2724
1-800-421-7462 (Ohio Only)

Schoolfutures, Inc.
Gene Small
2092 East Main Street
Hillsboro, OR 97123
(503) 640-1729

Arthur B. Williams & Associates
Bill Nettles
P.O. Box 257
Hershey, PA 17033
(717) 566-6767

Teaching Systems, Inc.
Owen Savage
5529 Dyer
Dallas, TX 75206
(214) 361-8665

Ronald A. Williams, Ltd.
Ronald A. Williams
P.O. Box K-34
Richmond, VA 23288
(804) 282-8239

*

Sum Of Squared Deviations

Willard A. Nielsen, Jr.
F-29 Koshland Way
Santa Cruz, CA 95064

When you get your Microsoft Word update announcement for Version 2.0 and if you decide to take advantage of the offer, expect to spend $50.00 and to find Zenith Data System (ZDS) suspending technical support for Word users. I don't know if ZDS will continue to provide support for 1.0 users who decide not to upgrade. Support for 2.0 users will be available through Microsoft.

Microsoft came out with Version 2.0 long before ZDS announced its availability. This delay was explained to me by a software consultant at Heath: They experienced problems with their printer, and not because HUGgies are lax in returning their registration card. What occurred between the date Microsoft released Version 2.0 and the date ZDS announced the availability of the update?

If you are the charitable type, you might have thought that ZDS was rewriting the manual for the H/Z-150/160, as they did for the manual that came bundled with the kits. Wrong. You will receive the Microsoft manual. Not a bad manual, of course, but recall the changes in Microsoft's manual for Version 1.0 and the manual provided by ZDS. Now those changes can prove troublesome. But first . . .

What are you spending those 50 bucks for? First, you will receive five disks instead of ZDS' one. There is the Program disk, and a Backup copy disk for day-to-day use. The Utilities disk contains all the printer description files and a file called SETUP that is used to transfer the appropriate printer description and the MS-DOS operating system file to the backup program disk. The fourth disk is the Spell disk that contains the files for the various tools that are described below. Last, Microsoft provides a Learning Word disk that enables you to almost bypass the manual. For experienced Word 1.0 users, the explanation of how to use the Spell tools need only be read.

Second, you get a keyboard template. On one side, the template has aids for mouse users, on the other for keyboard users. But it doesn't really fit the 151 keyboard. Its value is dubious, but it looks cute.

Third, a lot more printers are supported in Version 2.0. That's great if you have a lot more printers or if you have one of those new laser printers or 24-pin head printers (e.g., Epson LQ-1500). More seriously though, the support of additional printers becomes important when you consider upgrading your printer system.

Fourth, you can get between one and three more lines of text in your window with Version 2.0. How? You have the option to turn the menu off. This extends the window length by one double-spaced line, or three single-spaced lines. This command is accessed through the O(ptions) command. There does remain, however, one status line informing you of the page/division number, { } (the scrape symbol), ? (the help symbol for mouse users), Microsoft Word: (document name). In other words, the last line of the command area, if you have the menu showing, still appears.

The menu remains off until you press <escape>. Then, the cursor moves to the command area, the window shrinks to 'normal' size, the command menu reappears. The moment you carry out whatever command you wanted (i.e., hitting <return>) the window expands and the menu disappears. To return to the expanded window without carrying out a new command, strike A(lpha).

Fifth, you should be able to access the IBM characters by using the character code number entered while depressing the ALT key. Thus, to get an o with an umlaut (^T), you need to press ALT-148. Neat! But you must use the numeric keypad, with or without activating the NUM LCK key. The quick reference guide provided by Microsoft, however, does not have the codes listed.

Other than these minor changes, and a few others even more minor (e.g., the ability to confirm and change page breaks), there is one significant feature of Version 2.0 that may make it worth the expense. Recall Version 1.0. Recall the command area. There was a command called Library that regardless of how it was accessed would do nothing. Now it does something.

The Library Program

Hit <escape>, L(ibrary). And what do you see? A sub-menu! Spell, Hyphenate, Run. This is deceiving, however, because there are more "tools" than these three. But to take advantage of the Library, you will need 256K RAM instead of the 192K recommended by Microsoft to use Word or the 128K recommended by ZDS. Of course, the more the merrier.

It is at this point where the difference between ZDS' manual and Microsoft's manual becomes telling. Recall how ZDS' manual suggested the 'autoexec.bat' file should look? Something like this:

```
DATE

TIME

B:

A:WORD

<CTRL>-Z
```

This puts your Word program (MW.PGM) and system (WORD .COM) files in drive A, and your document (or, data) in drive B. This suggestion was made because, by making your default drive the same drive in which you have your document, all you need do when you want to T(ransfer)/S(ave) is type the filename — there is no drive destination designation, no path names. Saving was facilitated.

To use the L(ibrary), however, you must have the Word program/system files in your default drive. For most of us, that would be drive A. Word automatically searches drive B for your document/data file, which you wish to load into the window, and automatically searches your default drive for any of the "tools" available through the L(ibrary). If you continue to follow ZDS' suggestion of making your default drive your document/data drive, expect not to be able to use some of the L(ibaray) commands. If you try, all you'll get is a Bad command or filename message! You can run a few of the commands by pretending they are normal DOS commands, however (this will be explained below).

S(pell) will check your spelling according to three different dictionaries. There is Microsoft's resident dictionary (a general dictionary for everyone and for every file), a special dictionary you can create that is attached only to the document in which you create the dictionary, and another dictionary you create, but available to all documents. Thus, you have the option of creating two special dictionaries and of adding words to the resident dictionary.

The L(ibrary) command, once evoked, defaults to S(pell), so all you need do is hit <return> or S. (All messages from the Word program will be in italics, commands in all caps.) You see: Saving work file . . ., then Enter Y when Spell disk is ready. Upon receiving this message, remove the Word program/system disk from drive A and replace it with the Spell disk. Once this is done, the screen goes blank and then reforms with three horizontal windows. The command area consists of DICTIONARY HELP OPTIONS PROOF QUIT, the default is PROOF.

If you wish to create a dictionary, press D and the command area asks you for the dictionary name. The help provided by evoking H does nothing to detract from Microsoft's reputation for providing excellent on-line help. But unlike the help provided by the main Word program, it is not context sensitive. Two OPTIONs determine the speed in which the S(pell) program searches for unknown words. The command menu changes to LOOK-UP: QUICK COMPLETE once O is evoked. The default is QUICK. As the name suggests, Q(uick) searches for misspelled words more quickly than C(omplete). C(omplete) compares every character in every string to every character and every string in the various dictionaries. QUICK, on the other hand, assumes that the first two characters of each string of characters are correct and starts its comparison with the third character. PROOF, however, forms the backbone of the S(pell) program.

Press P, and what do you see? First, a new command menu: ADD CORRECT HELP IGNORE MARK OPTIONS QUIT RESUME NEXT PREVIOUS, the new default is IGNORE. Then the following messages: Reading document; Checking Main Dictionary; Checking User Dictionary; then Checking Document Dictionary. Expect to twiddle your thumbs for a few seconds while the S(pell) program goes through the document. After it has finished, it shows the unknown word in the third window and that word in context in the first window. You then make a choice as to ignore, mark, correct, or add it to one of the dictionaries. If you decide to correct it, the command area changes to CORRECT: and you see the message Enter correct spelling or press direction key for (possible) alternative spellings. The alternatives are listed in the second window. You can either use the direction keys to highlight the correct spelling or type it yourself. The command is carried out by hitting <return>. If you decide to MARK the word, an asterisk (but you can define the mark to be anything else if you wish) will be inserted in front of the word and at every occurrence in the document. This facilitates searching once you exit the S(pell) program. The rest of the command menu is self-explanatory.

After all the unknown words have been displayed, S(pell) asks you to Enter Y to process, N to discard changes, or Esc for previous. Incidentally, this program also gives you a word count of the total number of words in the document, the number of words appearing only once, and the number of unknown words. Once you've entered Y or N, the screen goes blank and you are asked to replace the Spell disk with the Word Program disk. Be forewarned: you are placed at the beginning of the document and not where you were when you evoked S(pell).

H(yphenate) allows you to control the hyphenation option. If activated, Word will automatically hyphenate words that normally would be brought to the next line via wordwrap. But for this you will need the HYPH.DAT file. Since I never want hyphenation to be evoked, I've erased this file from my back-up copy. So I cannot report on how it functions. But if it works like the rest of the programs, I suspect that those who wish for hyphenation will be very happy.

It is from the L(ibrary)/R(un) command where you access other tools that were heretofore unavailable in Version 1.0 and some of the DOS commands. Hit <escape> L R and you are asked to enter a DOS command. You may enter the command to evoke one of the new Spell tools (explained below) or a DOS command. One point of confusion may arise when L(ibrary)/R(un) is evoked. Once you've entered a specific command, you will be asked to enter Y when the program disk is ready. If you've entered a resident DOS command and if you have COMMAND.COM on your Word program disk, then you do not need to change disks. Just enter Y. If you've entered a transient DOS command, then replace the Word program disk with your MS-DOS disk. If you've evoked one of the Spell tools, then replace the Word disk with the Spell disk. All Spelling tools reside

on this disk. In short, the message Enter Y when the program disk is ready may be asking you for the COMMAND.COM file, or else the MS-DOS disk, or else the Spelling disk. So be sure you know what you're asking for so that you will know what program disk is being asked for. Be that as it may, what is of interest for me is the additional tools that are available.

Other Spell Tools

There is a Word Count tool (evoked by the command WC FILENAME.EXT <return>) that quickly tells you the number of words in your document. (And I do mean quick, I was told that one of my documents had 12,021 words in a matter of seconds!) Of course, the Spell program also tells you that, but by running the Word Count tool you can get the information a lot quicker. The spelling, of course, is not checked.

There is a Word Frequency program (WORDFREQ FILENAME .EXT <return>) that allows you to print and study the number of times you use each word in your document. On the screen, it provides a summary consisting only of the total word count, number of unique words, and the number of unknown words. Then it will make a file (with the same filename as your document but with the .FRQ extension) and list it in order of frequency — from the words most frequently used to those used least frequently. This list is treated as any other Word document file.

Three other tools exist as part of the Spell disk. There is a tool called Lookup (LOOKUP <return>) that finds the spelling of a specific word. It asks you to input a string of characters (word) and it will compare that string with its dictionary and then give you possible spellings. This saves you from having to have a dictionary close at hand. Another tool is called Anagram (ANAGRAM <return>) that can either unscramble words or create a list of words that can be formed from a specified string of characters. The last tool is called Word Find (WORDFIND <return>) that searches for words matching a specified string of characters containing wild characters (* or ?).

Unlike the S(pell) program, all tools return you to the place from where you evoked the commands. Remember the S(pell) program places you at the top of the document, regardless of where you evoked the program.

Conclusion

The essence of Version 2.0 lies in the Library. Is it worth $50.00? As a writer, I absolutely need my words to be correctly spelled. I wish Microsoft had a better description of the main dictionary (e.g., number of words) and the capacity of the user and document dictionary (e.g., the maximum number of words possible). Indeed, I would like a way to review the dictionaries in order to remove words appearing in them that I never or only very occasionally use. The possibility of creating various dictionaries that are document specific is nice if you write for diverse audiences, each requiring a specific lingo.

If your system does not have 256K RAM, the cost of memory has so decreased that you should anticipate spending no more than $25.00 to upgrade a 128K system to 256K. This will buy you two banks of 64K RAM. But with only the minimum memory, Word is slooooooooow. Remember that the price of a memory upgrade board is also falling. You should be able to get a fully occupied 2MB board for around $300.00. Considering that you built the 151 in the first place, the task of installation should not be intimidating.

In conclusion, I would say that the sum of squared deviations between Version 1.0 and 2.0 to be more than the 50 bucks you have to pay, but the absolute value depends on your memory configuration and whether a very good and quick spell program is worth that absolute value. *

Updated HUG Club Information

New Club

Winter Park, Florida
Martin-Marietta Computer Club
1730 Shiloh Lane
Winter Park, FL 32789-5847
(305) 356-3782 Group Size: 300
Contact Person: George McClure
Meeting time and place varies.
Newsletter; BB being established.
Evening Phone: (305) 647-5092

Updated Information

Alaska HUG, Eagle River, meets at Horizon Recovery Center at 1650 S. Bragaw. Phone: (907) 279-7754.

Sun Cities, Arizona HUG now meets the third Sunday of each month at 2:00 pm at the Heath/Zenith Computers and Electronics Center, Tucson, AZ.

Denver HUG has a new address: c/o Rob Chapman, P.O. Box 20023, Denver, CO 80220-0023.

New Contact Person for **Indianapolis, Indiana HUG** is Mark Siminski. New address and phone: 7610 Home Drive, Noblesville, IN 46060, (317) 841-0576.

The **Memphis Tennessee HUG** has disbanded. However, they hope to reorganize in the future. Those interested may send inquiries to P.O. Box 3221, Memphis, TN 38173.

With regret, notification has come to use that the **Southwest Michigan HUG, Kalamazoo, Michigan** has also disbanded.

The **Anderson, South Carolina HUG** meeting time and place varies from month to month.

Toledo, Ohio HUG's new president is Tim Tribble. He can be reached Monday thru Friday after 6:00 pm at (419) 862-2417. THUG mail should be sent c/o H/Z Computers and Electronics Center, 48 S. Byrne Road, Toledo, OH 43615. The club now meets the second Wednesday of each month at the store.

Keith Davis, R#1, Kendallville, IN 46755 would like to organize a HUG club in the **Ft. Wayne, Indiana** area. Those interested may contact him at (219) 347-0242. *

Performance.

Introducing EasyPC™ from UCI. The most advanced IBM PC emulator you can buy for your Z-100. First to provide the compatibility, speed and audio support to make your Z-100 run like a PC. Since the EasyPC provides complete PC circuitry, you actually run PC programs on PC hardware. Quite simply, it's like installing a PC right inside your Z-100. In the PC mode, your Z-100 can run virtually the entire library of PC programs, including LOTUS 1-2-3, dBase III, FLIGHT SIMULATOR, and most other copy-protected software. In the Zenith mode, your Z-100 is a Z-100, with no change in performance.

EasyPC is fast. In fact, it is noticeably faster than both the Z-150 and the IBM PC. And EasyPC can run at either 5 or 8 MHz. It supports both floppy and Winchester drives, provides crisp color video graphics, supports PC-compatible sound generation, and even comes with its own self mounting speaker.

Quality.

You already know and appreciate the kind of quality and reliability that is built into your Z-100. UCI matches that commitment to quality from first step to last. Sophisticated engineering and quality components combined with precision automatic assembly, component burn-in, and both in-circuit and full functional testing, mean superior performance and long term reliability.

EasyPC's advanced circuitry utilizes two, high quality, S-100 multilayer boards and a piggyback board. Yet its ingenious design requires only *one extra slot*. So you get all the advantages of PC compatibility, with room to spare.

Availability.

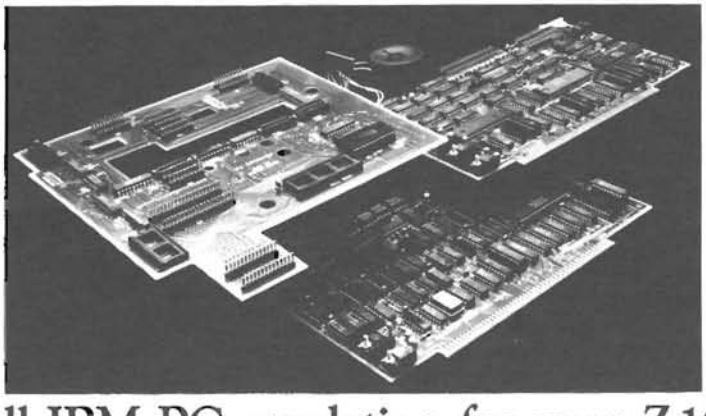
EasyPC is ready when you are. If you've been waiting too long for PC compatibility, your wait is over. EasyPC is here now. Complete documentation is included for quick and easy installation, so you'll be up and running fast. Your only problem will be which PC programs to get first.

EasyPC is available through your independent Zenith dealer. Ask him for a demonstration.

Price.

At $699, you'll pay a little more. But you'll get the best. Total compatibility. High speed performance. Unsurpassed quality and serviceability. In a word, value. You also get UCI's one year Solid Service Guarantee and full customer support. If you have any questions or problems with any UCI product, or need the name of your nearest dealer, just call 800-UCI-COMPUTER.

EasyPC™



Full IBM PC emulation for your Z-100.

UCI CORPORATION
948 Cherry Street, Kent, Ohio 44240
(216) 673-5155 / 800-UCI-COMPUTER

A Closer Look At The Juki 6100 Printer

D. C. Shoemaker
HQ USEUCOM Box 897
APO NY, NY 09128

Several recent articles in the computer magazines have presented a useful and timely look at the Juki 6100 daisywheel letter-quality printer. It uses friction feed to permit use of letterhead paper, prints at an honest 18 characters per second, uses readily available print wheels from Triumph Adler and ribbons for the IBM Selectric, isn't all that noisy, and is built like a Sherman tank. The price is very attractive, too, generally in the $400 – 450 range, depending on where you look. I was in the market for such a printer (it was my wife's idea, and believe me, you don't waste any time acting on that kind of suggestion!) and the reviews I'd read convinced me that the Juki was the printer I wanted. I have some special factors in my evaluation process, in part due to my overseas location and in part due to my dependence on the mails, so you might very well choose another. The Juki also had the advantage for me of being a known quantity, since an acquaintance already had one in professional use with his Zenith 120 and was thoroughly pleased with it.

I bought mine by mail order from a highly reputable dealer in the U.S. (CCS Data Station, 8 Roanoke Street, Christiansburg, VA 24073). When you're overseas, you can't be too careful. A quality dealer is the only way to do business, especially overseas; if anything goes wrong, he or she is your only hope. I received it in good order through the Army Postal System, which says volumes about the way Juki packs their equipment; I've seen printers arrive in oddly reshaped boxes that rattled ominously. CCS's owner, Ted Fleshman, had tested the printer before shipping it in its original shipping carton, heavily protected by form-fitting plastic blocks, so I knew it worked before it left Virginia. It didn't take long to find out that it worked right out of the box when I ran the self-test at home.

There are a few things in the published reviews that I've seen that require some clarification, and that's one purpose in my writing this. The other purpose is to explain how to do the parallel port interfacing. It turned out to be more difficult than I'd expected, and I'd like to save you the same trouble.

First, with respect to the printer itself. One of the first things that I looked at before I bought my printer was the specification sheet. It said the Juki 6100 would operate on either 110 or 220 volts, 50 or 60 Hertz. That was true. What the spec sheet didn't say was that the power supply board installed in the printer determines what voltage to use. There's no voltage selecting switch as we Heath/Zenith owners have come to know and love. If you order your printer from the U.S. as I did, it comes with a 110 volt power supply installed. If you buy it in Europe, it comes with the 220 volt supply. I have no idea what the power supply board costs, and I'm operating my printer on a 220-to-110 volt transformer with no difficulty at all, so it really doesn't matter. There's just the irritation that I have to use a transformer at all, and I personally don't like to have multiple voltages on my system. Naturally, the frequency of the alternating current makes no difference; the power supply rectifies everything to direct current.

Next, the printer does not come with a connecting cable. Don't laugh; some of us old-timers who've used nothing but Heath equipment have been spoiled. All my Heath printers came with the required cables. Most other equipment makers do not provide this support. That's normally no problem; if you forgot to buy one when you bought your printer, you just go back to the store and get it. In my case, that's a 5,000 mile trip. I had to make my own from locally available parts. More on that later. Forewarned is forearmed.

The published performance appraisals are generally accurate. The only two areas where elaboration is required deal with specialty print quality and documentation. Print quality first. Like any good daisywheel printer, the Juki produces formed-character print that is even, dark and crisp. It's also fast and accurate in head positioning, due in part to its unique linear induction motor. This precision is a two-edged sword. The Juki doesn't offset boldface print sufficiently to make it standout very well. If you look closely, you can tell that a character or a word is in boldface, but the casual reader would never notice. This is not a problem for me, as I never use the feature in my writing; it may be more important to others. Juki may have improved its performance in this area, as some reviewers have commented on a lack of accuracy in this area. On the plus side, every character lines up exactly right, with no unevenness at all. The print output is very professional-appearing and pleasing to read.

My other observation also reflects improvements on Juki's part in another area. The manual that came with my printer is 163 pages long, and a vast improvement over the 44-page original. I only know of one printer manufacturer who does a better job with manuals, and that's Heath. Given the fact that Juki doesn't expect you to do any circuit-level maintenance, their new manual provides literally all the information you need to interface the printer to almost any computer. They provide highly detailed step-by-step instructions for what they consider five of the most common computers, the Apple II, the IBM PC (and hence probably the Zenith 150?), the KayPro II, the Osborne I (alive and doing well in Europe) and the TRS-80 Model III (and by extension the Model IV). In the case of the IBM, Juki even goes into detail on how to patch the Basic I/O System (BIOS) to make the IBM work as it should (there may be a moral there).

Having said how much I like the Juki and its manual, I'd like to illustrate computer/printer interfacing with a discussion of how to connect said printer to a Heath 110 or 120 computer. Since serial (RS-232) interfacing is generally the most difficult and offers the greatest variety of problems hooking up (there seems to be no real standard, as you'll see when you look at the manual), the manual devotes the most attention to that. However, to use the Juki in the serial mode requires an extra-cost optional serial interface board. I chose not to pay extra for that, in part because I already have uses for both my H/Z-120 serial ports, so I was more interested in the parallel interfacing requirements.

The standard form of parallel interface originated years ago with the Centronics line of printers. Like the S-100 buss, the Centronics interface became a default standard, and a great many manufacturers adopted it. Juki is one of those. Fortunately, so is Heath. Not so fortunate is Heath's omission of the parallel port pin-out for the H-110 and H-120 computers. Since Heath, like some others, uses DB-25 connectors for all I/O ports, whether serial or parallel, it wasn't just a question of buying a standard parallel cable with a male plug on each end. While the Juki required the male Centronics plug, the H-120 required a male DB-25 plug. At first glance, they're not compatible, and the H-120 manual gave no pin-for-pin correspondence. I'm sorry to say that I had to borrow a page from the Z-150 manual to see how to connect a 38-pin male parallel plug to a 25-pin DB-25 connector. Figure 1 shows those connections, so you won't have to go hunting about for them.

Centronics Plug		Signal		DB-25 Plug
1	----------	STROBE	-------	1
2	----------	DATA Ø	-------	2
3	----------	DATA 1	-------	3
4	----------	DATA 2	-------	4
5	----------	DATA 3	-------	5
6	----------	DATA 4	-------	6
7	----------	DATA 5	-------	7
8	----------	DATA 6	-------	8
9	----------	DATA 7	-------	9
1Ø	----------	ACK	-------	1Ø
11	----------	BUSY	-------	11
12	----------	PAPER ERROR	---	12
13	----------	SELECT	-------	13
15	----------	ERROR	-------	32
16	----------	INITIALIZE	----	31
25	----------	GROUND	-------	33

Figure 1
This chart shows the pin-for-pin correspondence for the Centronics parallel connector to the DB-25 connector for the H/Z-110, 120, 150 and 160 computers when connecting to a Juki parallel interface printer.

I made my cable from a spare length of 25 wire cable from some now-forgotten Heathkit. You could also use a flat ribbon-type cable. Note that you need only 16 wires, and all that's necessary is that they be color-coded to ease your assembly. The connections seem "natural" so it may be possible to buy commercially made cables that meet the Juki-Heath requirements, but nothing of the sort is available in my part of Germany.

Once the cable is made, you can configure the computer's operating system to drive it. The Juki looks just like a Diablo 630 to the computer, so where that printer is offered as a choice (for instance, PeachText 5000) just select that. Remember that when you configure a text processor like PeachText 5000 or WordStar, you must also configure the operating system. For CP/M-85 and Z-DOS/MS-DOS, just select the first option, the parallel Centronics or MX-80 option. From then on, everything will work just fine.

When you use the parallel interface, it's not necessary to open the printer case to get to the configuration switches on the circuit card in the back. There's no baud rate or parity to worry about. The only switches you'll need to set, occasionally, are the ones inside the front cover. These are much easier to get to, and control line spacing and the like. You can also change these characteristics by writing a small program that transmits the proper escape codes to the printer in response to a console command. For instance, the public domain programs like TP.COM that allow you to type on the printer directly from the keyboard can be used to send the control codes.

Speaking of TP.COM, programs like that can be used to easily address envelopes on the Juki 6100. Just insert the envelope and type as if you were using a typewriter. Much easier than printing one label for that occasional letter to Aunt May. Looks very professional, too. Naturally, when you use single-sheet printers it's easier to write programs in Pascal or BASIC that would address a whole stack of envelopes for you, reading the addresses from a data file. Some word processing programs will do similar things for you.

Controlling the Juki with software running from your computer is a relatively easy programming task. Even if you're not a programmer in the full sense of the word, you can use BASIC or even assembly language to set most of the Juki's printing options. For instance, while there's a front-panel switch on the printer to control the number of characters per inch of type (characters per inch, or CPI) the corresponding switch for the number of lines per inch (LPI) is located inside the front of the case. This means removing the top cover whenever you want to shift from six lines per inch to eight. As this is one of the print parameters I change frequently, I immediately began to look for a software solution. My first attack on the problem was with the BASIC program shown in Listing 1.

```
1Ø CLS ' Clear the H/Z-1ØØ screen
2Ø PRINT"Line Space Index (LSI) Test Program":PRINT
3Ø INPUT"LSI (7 or 8)";N ' Get the desired line spacing
4Ø LPRINT CHR$(27);CHR$(3Ø);CHR$(N) ' Send it to the Juki
5Ø LPRINT"The LSI is now set for";N ' Report what we've done
6Ø END ' the program
```

Listing 1
A ZBASIC program to set the lines per inch, or Line Space Index, for the Juki 6100 daisywheel printer.

Now your first observation probably is that this isn't a very elegant way to set the line spacing, involving as it does calling ZBASIC, loading the program and running it, then exiting ZBASIC and going on about your word processing tasks. And you're right. However, it's also quick and easy, and a good way to see what the Juki manual means with some of its examples. For instance, it may not be obvious that a seven will get you eight lines per inch, and an eight will get you six. But closer examination of the manual (page 117 in my copy) shows that this is how it's done.

You say you'd like something more practical, something that you could execute directly from the operating system without loading BASIC? The program in Listing 2 may help. While the BASIC approach will work with any disk operating system on any com-

puter, this program is written in 8080 assembly language for use with CP/M 2.2. It could easily form the basis of a similar program in another assembly language for another microprocessor, but that, as they say, will be left as an exercise for the student.

If you're familiar with assembly language, there are no mysteries in the program above. If you're not an assembly language programmer but want to try your hand, just type what you see above, and call the CP/M assembler by typing the ASM SET61. If you've typed everything exactly as shown, you will get some largely meaningless numbers and the CP/M prompt. When that happens, type LOAD SET61. When that process is complete, you'll get another set of numbers. Of greater interest, you will have an executable file called SET61.COM on your disk. Run it to set the Juki 6100 parameters.

Naturally, both of these programs could be easily changed to set any other parameter you might want to change from the power-up settings. The manual tells you what they are and how to set them.

By way of general information, for those of you who may have missed it in the March, 1985 issue of Byte (page 304), there's a ready source of inexpensive daisywheels for the Juki. Write to Frank Millis or Paul Dearman of Gentry Associates, Inc., 7665 Currency Drive, Orlando, FL 32809 (phone 305 859-7450.) Robert Levine indicates in his note to Jerry Pournelle that the print wheels sell for $7.95 each. That seems well worth looking into.

I should also point out, in all fairness, that in the same column (page 303), John Williams registered a strong complaint against the Juki tractor feed. I haven't had any personal experience with the tractor feed option, and due to the relatively slow speed of the printer itself can't really recommend it, but apparently there are problems both with the installation and the operation. Mr. Williams raised a potentially more serious problem when he said that thus far Juki hasn't been willing or able to resolve the trouble. Sometimes a little publicity can work wonders for a recalcitrant service department, so we'll have to wait and see. With a readership like the Good Doctor's, results may be forthcoming and the problem resolved. I hope so, because in all other respects the Juki is an excellent piece of equipment. And, as I mentioned already, I've had no trouble at all.

If you're looking for an inexpensive letter-quality daisywheel printer for under $500.00, consider the Juki 6100. It's reasonably quick, not too loud, easy to support with ribbons and print wheels, easy to interface, and a durable, well-built machine that should give years of use. Highly recommended.

```
;SET61.ASM for the Juki 6100 daisywheel
;by D  C  Shoemaker, 4/28/85
;
;Assembly constants
;
BASE    EQU     0
BDOS    EQU     BASE + 005H
TPA     EQU     BASE + 100H
;
;CP/M I/O functions
;
CRT     EQU     9       ;Print string
INPUT   EQU     1       ;Move char from kybd to A reg
LIST    EQU     5       ;Move char from E reg to LST:
;
;Miscellaneous EQUates
;
CR      EQU     0DH     ;Carriage return
LF      EQU     0AH     ;Line feed
BELL    EQU     07H     ;Bell
;
        ORG     TPA     ;Start program at 100H
;
;Print prompt and get input
;
START   LXI     SP,STACK        ;Set up stack
        LXI     D,PROMPT        ;Point to prompt string
        MVI     C,CRT           ;Call console print function
        CALL    BDOS            ;Print on screen
        CALL    SCIN            ;Move char from kybd to A reg
        CPI     '6'             ;A 6?
        JZ      LPI6            ;Then LPI=6
        CPI     '8'             ;Is it an 8?
        JZ      LPI8            ;Then LPI=8
        CPI     ' '             ;None of the above?
        JMP     ERROR           ;Sound console bell and
                                 return to menu
;
SCIN    PUSH    H               ;Single Character INput
;
        PUSH    D
        PUSH    B
        MVI     C,INPUT
        CALL    BDOS
        POP     B
        POP     D
        POP     H
        RET
;
ERROR   MVI     E,BELL          ;Sound console bell
        MVI     C,CRT
        CALL    BDOS
        JMP     START
;
;Printer style selection routines
;
LPI6    MVI     E,1BH           ;Reset to startup
        MVI     C,LIST          ;configuration of
        CALL    BDOS            ;6 LPI
        MVI     E,1AH
        MVI     C,LIST
        CALL    BDOS
        MVI     C,73H
        MVI     C,LIST
        CALL    BDOS
        RZ
;
LPI8    MVI     E,1BH
        MVI     C,LIST
        CALL    BDOS
        MVI     E,1EH
        MVI     C,LIST
        CALL    BDOS
        MVI     E,06H
        MVI     C,LIST
        CALL    BDOS
        RZ
;
;Prompt strings
;
PROMPT  DB      'This program sets lines per inch on the
                 6100 printer ',CR,LF
        DB      '    Your choices are:',CR,LF,CR,LF
        DB      '    (6)  Vertical pitch of 6 LPI',CR,LF
        DB      '    (8)  Vertical pitch of 8 LPI',CR,LF
        DB      ' ',CR,LF
        DB      '     Your choice: ','$'
;
        DS      10h             ;8 levels for stack
                                 (16 bytes)
STACK.
        END
```

Listing 2

An 8080 assembly language program to set the Juki 6100 printer's lines per inch parameter. For use with CP/M 2.2.

DISCOVER WHAT 10 SOFTWARE INSIDERS ALREADY KNOW!

1. "...a little company that has succeeded over the last five years in giving its customers a lot of value for their money..." **- Richard O'Reilly, syndicated computer columnist**

2. "All the firm's programs are graced with features that the entire industry should be required to provide. Their manuals are the most honest in the industry." **- T. R. Reid, synicated computer columnist**

3. "...the two most important attributes of Software Toolworks' programs: they're priced low, and they work well." **- The Washington Post**

4. "Unlike the software of many publishers these days, none of The Software Toolworks' programs are hobbled by copy protection, nor does the company try to tie up its customers with mumbo-jumbo licensing agreements..." **- The Los Angeles Times "Computer File"**

5. "I don't know how The Software Toolworks stays in business charging such low prices for such nifty products..." **- David Gerrold computer columnist and science fiction author**

6. The Software Toolworks C/80 compiler is "...the best software buy in America..." **- Microsystems**

7. "It would be the best of all possible worlds if some magic diskette could be made that would transform any hardware in such a way as to receive The Software Toolworks' numinous software." **- Wm. F. Buckley, Jr.**

8. "The Software Toolworks has a whole raft of programs that work, and (the company) sells them for reasonable prices. I strongly recommend that you get their catalog." **- Dr. Jerry Pournelle computer columnist "Byte" and science fiction author**

9. "One of the original pioneers in low cost, high quality software, The Software Toolworks publishes over 50 solid products for MS-DOS and CP/M under $60." **- Walt Bilofsky, chairman of The Software Toolworks**

10. "All products are shipped within 24 hours, come with a money back guarantee, and free technical support is available (though you'll hardly need it)." **- Joesph Abrams, president of The Software Toolworks**

SEE US AT THE WEST COAST COMPUTER FAIRE!

MEDIA MASTER PLUS™ - Now you can run most CP/M programs on your IBM PC or compatible *without* additional hardware. *Media Master™* lets you read, write and format over 140 different double density CP/M formats. *ZP/EM™* (included) emulates an 8080 processor to execute CP/M software at machine language level giving you the equivalent of a 1 MHz 8-bit computer. Includes terminal emulation for Osborne, Kaypro, Heath/Zenith and DEC VT52 displays. Requires 192K RAM. For IBM PC and compatibles. – **$59.95**

TOOLWORKS C™ - Ideal for learning this popular programming language, the full featured *TOOLWORKS C* compiler is powerful enough for professional program development. 8-bit CP/M version proven in over 5 years of use to be among most efficient and reliable. MS-DOS version produces relocatable object files (.OBJ), includes complete library source code, supports all version of DOS. An outstanding value. – **$49.95**

TOOLWORKS C MATHPAK™ - Add true 32-bit floating point and signed integer data types to the *TOOLWORKS C* compiler. Speed programming tasks with a complete library of mathmatical functions (source) included. MS-DOS version supports the 8087 for true number-crunching power. – **$29.95**

AIRPORT™ - You are an air traffic controller with radar screen filled with aircraft under your guidance. See how long you can last without cracking. Tougher than the real thing! CP/M and MS-DOS. – **$19.95.**

LISP/80™ - Experiment with artifical intelligence programming! Over 75 LISP functions. Includes source for an editor written in LISP and several demonstration programs. CP/M and MS-DOS. – **$39.95**

READING PROFESSOR™ - Boost your reading speed while increasing comprehension and retention. Self paced lessons show quick improvement with immediate feedback. Includes 48 practice reading passages and other exercises. Developed by skilled educators, this program tracks your progress at reading rates up to 2,500 wpm. Additional libraries available for Junior High, College/Adult, and Professional levels at *$15.95* each. For IBM PC, most CP/M-80 computers. – **$59.95**

COMPUTER CHEF® - Original computerized cookbook and recipe file. Data base of over 70 recipes with space to add your own. Computer selects recipes by ingredient, keyword, or title then displays or prints your selection. Scale recipes by portions or ingredient quantities. CP/M and MS-DOS. – **$29.95**
WOK TALK™ (100 Chinese recipes) – **$29.95**
CHOCOLATE BYTES™ (100 chocolate recipes) – **$29.95**
WHAT'S FOR DINNER™ (200 recipes for use with above programs)– **$19.95**

THE ORIGINAL ADVENTURE™ - Explore the dangerous Colossal Cave for treasures. Endure mystical spells, overcome adversaries. The only official version, uncut, endorsed by its creators, Crowther and Woods. CP/M and MS-DOS – **$19.95**

MYCHESS® - Award-winning world class chess program, featuring nine skill levels and over 850 opening moves. Prints moves and saves games in progress. Outstanding graphics display board on IBM PC and most CP/M computers. – **$34.95**

ELIZA - The pioneering artificial intelligence program, *Eliza* is a "psychiatrist" that carries on a conversation in plain English. Use authentic script, expand it or develop your own. CP/M and MS-DOS. – **$24.95**

The Software Toolworks®

14478 Glorietta Drive Sherman Oaks, CA 91423

MasterCard VISA AMERICAN EXPRESS

TO ORDER CALL:

1-800-223-8665; in CA 1-800-228-8665

U.S. and Canada add $2.00 per order s/h. Overseas orders add $5.00 per order. CA residents add 6.5% sales tax. *Dealer inquiries invited.*

Giving New Life To An Old Friend

Jim Buszkiewicz
HUG Software Developer

Hardware Review Of The H/Z-25 Super Chip Set

The H/Z-25 printer has always been the workhorse of printers in the Heath/Zenith product line. Even though the printer is no longer being produced, thousands are still in use, and because of people like Larry Fina of FINA SOFTWARE, will probably be so for a long time to come. Mr. Fina has created a new super firmware chip set for the H/Z-25 printer. This new firmware consists of three new ROM ICs and a 2k static RAM chip, and is presently selling for $84.95.

Installation of the part is quite simple and would probably be best accomplished by removing the cover completely. Once the cover is off, the chassis cover can then be carefully lifted by removing the four 6-32 X 1/4" phillips head screws holding it in place. Removing the chassis cover should expose the printer logic circuit board. U231, U233, and U234 are then removed and replaced with the three new firmware ROMs. Socket U235, which is now empty, receives the new 2k X 8 static RAM chip. Five jumper changes are made, and the installation is complete, except, of course for reassembly! To help with parts placement on the logic board, FINA includes a detailed pictorial with their instructions, showing where the new parts and jumpers go. I'm confident that anyone can successfully make this modification, even if you originally did not construct your printer.

This new firmware now includes running a test printout on 8" wide paper instead of the mandatory 14" stuff. The new character set can now be thought of as two banks of 128 characters. The first bank of characters is the one that originally came with the machine. The second bank consists of symbols, Greek characters, fractions, arrows, heiroglyphics, etc., and even a smiley face. By special order, this second bank of characters could be made up according to your own specifications. This entire 256 byte character set can now be printed in the following different ways:

1. normal
2. italics
3. as superscipts
4. as subscripts, as italic
5. as italic superscripts
6. as italic subscripts
7. single or double strike
8. interline underlining

All these different print styles and underlining occurs with a single pass of the printhead!! In addition, you still have double-sized normal, double-sized italics, double-sized double-strike, double-sized underlining, and double-sized underlined italic double strike.

Whew! For a minute there, I felt like I was doing a chewing gum commercial! All of the new functions are accessed via escape codes and where possible, Mr. Fina made them Epson compatible. Some of the other non-printing features are, print head automatically clears left hand perforations during a form feed, and linefeed access in 1/48" increments.

Fina Software is located at 16144 Sunset Blvd. #3 in Pacific Palisades, CA 90272, and their telephone number is (213) 454-6393. Before shipping, each ROM set is tested and a functional test printout is included with each order. Mr. Fina has spent literally, thousands of hours in creating and perfecting this firmware. Time, I feel, that was well spent.

INTERNATIONAL
HEATH/ZENITH USERS' GROUP CONFERENCE

Official Conference Registration Form

O'Hare Hyatt Regency
Rosemont, Illinois
August 15, 16, 17

Name(s): _____

_____

_____

Company: _____

Address: _____

City: _____ **State:** _____ **Zip:** _____

Enclosed is $25.00 for each of the individuals listed above to attend the International HUG Conference being held the weekend of August 15, 16, and 17, 1986. Please send tickets along with information regarding hotel reservations and transportation.

Amt. Enclosed: _____ No. Attending: _____

For Our Information:

Which Heath/Zenith computer do you now operate? _____

Are you a Non-User-Attendee?	Yes	No
Are you a computer related manufacturer?	Yes	No
If yes, would you like exhibit information?	Yes	No
Are you, or anyone in your party, interested in activities in or around the Chicago area other than the Conference?	Yes	No

If yes, please indicate any suggestions you may have:

_____

Special Notice To Exhibitors:

Exhibitor Information Packages are available on request from the Heath/Zenith Users' Group. Those of you interested in exhibiting your products should contact us as early as possible to ensure a position at this year's event.

For Your Information:

The $25.00 you are paying for your reservation to the International HUG Conference entitles you to all functions of the Conference. Visitor tickets, for those of you simply attending the seminars and exhibits, are available for $10.00. Visitor tickets do not include eligibility for prizes or food while attending the Conference.

Please send your completed registration form or suitable copy to:

Heath/Zenith Users' Group
Attention: International HUG Conference Registration
Hilltop Road
St. Joseph, Michigan 49085

Registration(s) must be post marked no later than July 31, 1986. Cancellation will not be accepted after this date.
Sorry, We cannot accept purchase orders

Continued from Page 41

DATA BASE MANAGEMENT SYSTEMS

Part Number	Description of Product	Selling Price	Issue
HDOS			
885-1107-[37]	HDOS Data Base System H8/H89	30.00	23
885-1108-[37]	HDOS MBASIC Data Base Sys.	30.00	23
885-1109-[37]	HDOS Retriever ASM (3 Disks)	40.00	23
885-1110	HDOS Autofile (2 Disks)	30.00	23
885-1115-[37]	HDOS Navigational Program	20.00	25
885-8008	Farm Accounting System	45.00	30
CP/M			
885-1219-[37]	CP/M Navigational Program	20.00	31
MSDOS			
885-6008-37	MSDOS NAVPROG	20.00	73
885-8034-37	DBZ-A Database For The Z100	25.00	69

AMATEUR RADIO

Part Number	Description of Product	Selling Price	Issue
HDOS			
885-8016	Morse Code Transceiver Ver 2.0	20.00	42
CP/M			
885-1214-[37]	CP/M MBASIC Log Book (64k)	30.00	23
885-1234-[37]	CP/M Ham Help	20.00	49
885-1238-[37]	CP/M Ascinity	20.00	57
885-8020-[37]	CP/M RF Comp. Aided Design	30.00	44
885-8031-[37]	CP/M Morse Code Transceiver	20.00	57
MSDOS			
885-8038-37	MSDOS RFCAD Ver. 3.50	30.00	73

COMMUNICATION

Part Number	Description of Product	Selling Price	Issue
HDOS			
885-1122-[37]	HDOS MicroNET Connection	16.00	37
885-8005	MAPLE (Modem Appl. Effector)	35.00	29
CP/M			
885-1207-[37]	CP/M TERM & HTOC	20.00	26
885-1224-[37]	CP/M MicroNET Connection	16.00	37
885-3003-[37]	CP/M ZTERM (Z100 Modem Pkg)	20.00	34
885-5004-37	CP/M-86 TERM86 and DSKED	20.00	56
885-5005-37	CP/M-86 16 Bit MicroNET Conn.	16.00	61
885-5006-37	CP/M-86 HUGPBBS	40.00	62
885-5007-37	CP/M-86 HUGPBBS Source List.	60.00	62
885-8012-[37]	CP/M MAPLE (Modem Program)	35.00	34
885-8023-37	CP/M-85 MAPLE	35.00	45
MSDOS H/Z100 - H/Z150 PC			
885-3019-37	ZDOS 16 Bit MicroNET Connect.	16.00	61
885-3027-37	MSDOS HUG PBBS	40.00	66
885-3028-37	MSDOS HUG PBBS Source Listing	60.00	66
885-3033-37	MSDOS HUG MCP	40.00	71

MISCELLANEOUS

Part Number	Description of Product	Selling Price	Issue
885-0004	HUG Binder	5.75	
885-1221-[37]	Watzman ROM Source Code/Doc	30.00	33
885-4001	REMark Vol. I Issues 1-13	20.00	
885-4002	REMark Vol. II Issues 14-23	20.00	
885-4003	REMark Vol. III Issues 24-35	20.00	
885-4004	REMark Vol. IV Issues 36-47	20.00	
885-4005	REMark Vol. V Issues 48-59	25.00	
885-4006	REMark Vol. VI Issues 60-71	25.00	
885-4500	HUG Software Catalog	9.75	
885-4501	HUG Software Catalog Update #1	9.75	
885-4600	Watzman/HUG ROM	45.00	41
885-4700	HUG Bulletin Board Handbook	5.00	50
885-3015-37	ZDOS Skyviews	20.00	55

NOTE: The [-37] means the product is available in hard sector or soft sector. Remember, when ordering the soft sectored format, you must include the "-37" after the part number; e.g. 885-1223-37.

Continued from Page 10

```
#endasm
}

cout(port)
{
#asm
    POP  D      ret adr
    POP  H      byte
    POP  B      port in c
    DB   355Q   prefix
    DB   151Q   out (c),L
    PUSH B
    PUSH H
    PUSH D      ret adr
#endasm
}
```

Sincerely,

Lansing E. Tryon
29 Fairhaven Road
Rochester, NY 14610

P.S. Thanks to Mr. Gilchrist for the port setup routine. I've got to try it.

Continued from Page 43

NAVPROGseven Update

HUG P/N 885-6008-37, NAVPROGseven for the H/Z-150, has been updated, by the author, to NAVPROGeight. This latest version has a new feature which performs the weight and balance calculations for the aircraft. Only this newest version will be shipped when ordering.

Heath/Zenith Users' Group

Changing your address? Be sure and let us know since the software catalog and REMark are mailed bulk rate and it is not forwarded or returned.

CUT ALONG THIS LINE

HUG MEMBERSHIP RENEWAL FORM

HUG ID Number: _____

Check your ID card for your expiration date.

IS THE INFORMATION ON THE REVERSE SIDE CORRECT? IF NOT, FILL IN BELOW.

Name _____

Address _____

City-State _____

Zip _____

REMEMBER - ENCLOSE CHECK OR MONEY ORDER

CHECK THE APPROPRIATE BOX AND RETURN TO HUG

	NEW MEMBERSHIP RATES	RENEWAL RATES
U.S. DOMESTIC	$20 ☐	$17 ☐
FPO/APO & ALL OTHERS*	$35 ☐	$30 ☐ U.S. FUNDS

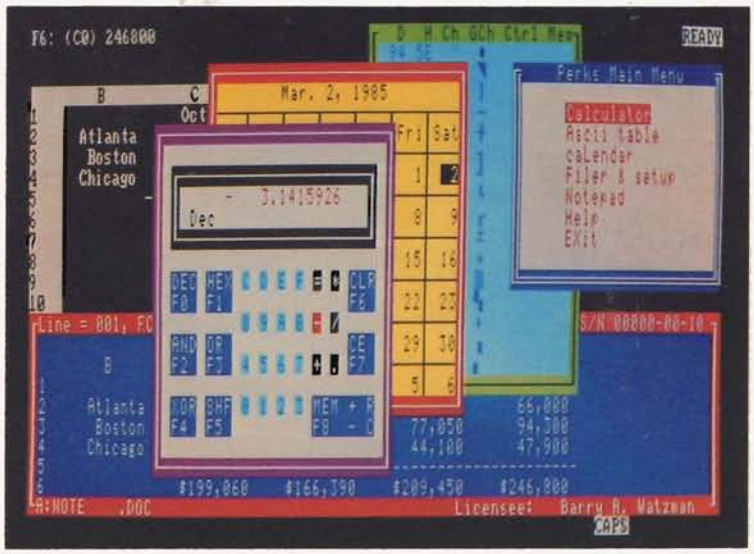
* Membership in France and Belgium is acquired through the local distributor at the prevailing rate.

Announcing

PERKS™

Version 2!!!

New! Cut/Paste Module!
New! Typewriter Module!
New! Variable Size Notepad Buffer (up to 64K)!
New! Modem Module!
New! Dialer Capability!
New! Card File Module!
New! Removable under *both* MS-DOS *and* Z-DOS!
New! Additional Features in Existing Modules!
Old! Still works with *both* MS-DOS *and* Z-DOS!
Old! Still small in size (79K)!
Old! Still only $69.95!



Shown above is an actual screen photo of Perks in operation. The Notepad window contains data "imported" from the Lotus 1-2-3 worksheet being prepared when Perks was activated. Shown is Perks Version 1, photographs of Perks Version 2 were not available at press time.

Now the leading Z-100 Desktop Utility is even better! Version 2 of Perks adds the most asked for additional features while retaining Perks' superior user friendliness, better documentation and compatability, ease of use, small size and ability to run under both Z-DOS and MS-DOS. If you have a Z-100, this is one program you can't afford to be without. And at it's low price of $69.95*, you don't have to!

Perks is available at all Heath/Zenith Computers & Electronics Centers, many independent Zenith Data Systems dealers or directly* from:

BARRY A. WATZMAN Microcomputer Systems & Consulting

560 Sunset Road • Benton Harbor, Michigan 49022-7142 • (616) 925-3136

Perks is a trademark of Barry Watzman.

*Plus $4.00 for S&H, Michigan Residents add 4% sales tax also.

Heath/Zenith Users' Group

Hilltop Road
Saint Joseph, Michigan 49085

BULK RATE
U.S. Postage
PAID
Heath Users' Group

POSTMASTER: If undeliverable, please do not return.

P/N 885-2074