Issue #7 THE STAUNCH 8/89'er Apr-May-Jun 1988

PORT TO PORTAL -- Editorial

I think you'll find extraordinary value in this issue. In fact, my only regret is I ran out of room! So something had to be cut: my own column, "The 8-Bit Iowan," but on good grounds, as I think you'll agree.

You've no doubt noticed the green insert by now, too. This is **not** a paid advertising supplement. I thought a list of the software distributed by **Staunch** would be worth including and didn't want to bite into valuable editorial space. So I've coupled the list with a flyer on my keyboard templates. You'll see some items on the list which ANAPRO is now releasing to the public domain.

But the major reason for the space limitation is the first appearance of a new section, called VENDOR.UPDATE. This is **the** place you can turn to for information on the vendors serving us and our systems.

There, you will read Peter Shkabara's letter suggesting the formation of a "demand group." I'd be happy to serve as coordinator for such, but I need to know how many of you might be interested in participating and what products (hard-to-find or discontinued) you're looking for. So write me at the address at the end of this issue after you've read Peter's letter. I'm also expanding it beyond **Staunch**'s still somewhat limited circulation.

But understand, for the moment I want to discover just how much interest there might be and the products desired. So don't send any money! If you and others respond enough, I'll query the relevant vendors to see if they might participate. I'll let you know what transpires by postcard and on these pages. And since you'll be writing anyway, Hank and I would like to know what kind of articles you'd favor seeing in **Staunch**.

Further, speaking of writers, if you plan to include source code in any article submissions, send for **Staunch**'s updated author's guide. I'm placing some limits on the size of your code I'll publish. Please include an SASE with that query.

Now to the reader poll I requested last time. My thanks to all of you who responded. I even received a couple of phone calls! Though many of you have had your machines almost as long as I have, 30% of you bought your hardware **after** mid-82, when H/Z introduced its first 16-bit machine (the '100), a date I'm using as a breakpoint. Twenty percent also have **REMark** collections that don't predate the same period. Further, 13% have H-8's and 6% have had your equipment less than three years.

So what do these numbers mean? Well, you will see more novice articles on these pages. But there will **still** be advanced material for you older users. Though it doesn't appear in this issue, "The 8-Bit Iowan" will continue to discuss the latter. I would also like to see some material specific to the somewhat neglected H-8. So let's hear it from you aspiring writers!

And one more thing, I received surprisingly little criticism from going to 88 lines per page (1pp) in the last issue. Still, it **is** more difficult to read even though I can sandwich more substance into **Staunch**'s pages. But when we go bimonthly, I'll revert to the 77 lpp that Hank used.

Finally, I must apologize to both Hank and you. To Hank because I misspoke when I claimed he used 66 lpp in the issues he edited; it was actually 77. And to you for the numerous typos in the last issue. You'll find this one more cleanly edited! And before I forget, the editorial deadline for the next issue is the 15th of July.

--Kirk L Thompson

THE EIGHT-BIT R/W -- Letters

Restarting the beginner

[From Hank Lotz, Pittsburgh, PA] I'm writing in reply to the letters from readers Kiessel and Allen in **STAUNCH** #6. First, regarding William H. Allen's request for beginner-type articles in **STAUNCH**, I have to say I'm behind him all the way. (Understand, of course, I wouldn't devote the entire issue to neophyte material.) I recognize that few (or no) radically elementary articles have appeared in **STAUNCH**, and I hope to help remedy that situation in the future. Previously, as editor, I was apprehensive about how "old-timers" might react at seeing too many simple rudiments, and I encouraged the purchase of REMark's back issues to fulfill the educational need, rather than our (pardon the expression) reinventing the wheel. I still recommend the **REMark** collection. But I think seasoned users should unselfishly bear with us if we devote some space to beginners. On the other hand, I don't really feel I have overly neglected the novice, witness the Q/A column for example. Also, I sometimes sneak "well-known" facts into my articles for the benefit of all -- novices included. That's why a beginner should not bypass any titles just because they may sound inapplicable or out-of-bounds to him. Read carefully, and you'll be surprised what you'll find -- tucked away!

Now, regarding the letter from Richard J. Kiessel -- I am very happy he expressed his enjoyment of my "zero-length file" article. (And, say! That article starts off: "This is beginner-level..."!) He is quite right about the ability of PIP to create the files. Actually, there are several other ways; I like SAVE 0 because it's much faster. Richard is also correct about my reference to using /.COM to rerun programs. Yes, "RESTART.COM" is a nice descriptive name, it's just that I make use of my /.COM **tens** if not **hundreds** of times each day at the computer, and I'm not about to be typing RESTART when a simple / will do! (Besides, technically, the term "restart" means to resume in the **middle** of an interrupted program, so maybe "RERUN.COM" would be more accurate.) His statement about it not working on WORDSTAR or SUPERCALC is understandably true. It's no good for MAGIC WAND either; results are unpredictable at best! -- I know because that's exactly how I dramatically trashed a disk once (with machine-gun-like sound effects)! And in fact, the inability of /.COM to salvage MAGIC WAND files finally prompted me to write a program called SALVAGE.BAS, which aids recovery of WAND files after, say, an inadvertent QUIT, or a "Bdos Err On B: Select". But, in general, there is no reason /.COM cannot work on programs which initialize data upon start-up. It's great for quick reruns of PIP, STAT.COM (and Derby's fantastic SD.COM), and other programs which **do** start from the beginning. I also use it for SUPER ZAP, with success so far. Those programs that do NOT actively initialize critical variables are the dangerous ones with /.COM. [See the **MISCELLANY** section later in this issue and the green insert for further information about Hank's SALVAGE and other CP/M utilities. -Ed.]

In the HAM shack

[From Dick Shotwell, 546 Grandview Drive N., Twin Falls, ID 83301] ...I am ... a Ham, W7GDS, and I want to get H-89 software for my packet radio system. I would prefer hard sector media but if that is not available, I will locate a soft sector board and change drives. I want to dedicate my older 89 to amateur radio use. Can anyone out there help? [Dick, instead of going to the expense of the soft-sector

upgrade (not that it isn't worth it; see "The 8-Bit Iowan" in issue #'s 2 and 3) you might check around for someone to do a media conversion for you. Most software doesn't care what type of drive it's stored on. -Ed.]

[From Jack Bankson (WA6JXG), 1482 Zion Way, Venture, CA 93003] I need HAM radio software for [a] hard-sectored CP/M H89.

HELP!
[From Terry Hall, Wheaton, IL; compiled from numerous letters and postcards] It seems you and OMAHUG have a good start on collecting HDOS software ... If you don't want to [become a national repository for it], maybe you should put out a plea for a volunteer. I might even be willing to consider it if no one else comes forth... [There are still a number of groups besides OMAHUG who handle HDOS materials, Terry. I'm trying to identify them for you and other readers; you'll find valuable information on that score in the new vendor section. -Ed.]

WANTED: Patch code to change Heath's standard H47.DVD driver to recognize port 174 (instead of port 170 currently on distribution copy). I want to run my H47 dual-8" controller card along with my 5 1/4" soft-sector controller. Would also be glad to buy a disk with modified software if anyone has this.

WANTED: I love my new SUPERSET 19/89 from TMSI, but Henry Fale's Quikstor HDOS software won't allow me to boot from my winchester with Superset installed. Quikstor looks for a standard Heath Terminal ROM. Has anyone successfully patched the software or found any other workaround to use Superset with a Quikstor winnie under HDOS? Please write or call collect if you can help me! ... I have nothing but praise for Lee [Hart], who's been most helpful on the phone and by mail. I think he'll keep at it till a solution is found. But in the interim perhaps buyers should be forewarned of the conflict...

[Finally, d]o you know of an HDOS or CP/M program for making intermediate-sized type on an Epson (something between regular and banner, like 18-to-24 point size)? Terry Hall, 516 East Wakeman, Wheaton, IL 60187 (312-665-4594).

[From T.J. Skollingsberg, 5448 S. China Clay Dr., Salt Lake City, UT 84118] --Has anyone hooked up the SK-203 printer buffer between the H8 and the H14?
--Has anyone converted their WH8-64 dynamic memory card to 256K using the Trionyx bank select card successfully?
--Isn't there an easy "plug-in" method of speeding up the H8's Z80 CPU to 4 or 6 MHz? Something with a few jumpers to change without needing to change all one's software?

Turboing SuperCalc
[From G.R. Stradley, 1304 Witthollow Rd., Sevierville, TN 37862] ...Last year I was able to get a used H-89A which came with dBaseII, SuperCalcII, & Turbo Pascal but no instruction books. It has 2 [half-height] DSDD disk drives & CP/M 2.2.04. It runs @ 2 or 4 MHz & has auto repeat keying. Also Write-Hand-Man. I am very pleased with it & especially the 380K disk space. Does anyone know of a way to make SuperCalcII run @ 4MHz? Also Turbo Pascal? dBase runs fine @ 4 but not the others. [I don't know a thing about SuperCalc, but Turbo will run at the higher speed if you configure it with TINST.COM. Perhaps someone can clue us in on SuperCalc? -Ed.]

HDOS 3.0 trouble on a classic H-8
[From Winslow Palmer, 120 Carol Woods, Chapel Hill, NC 27514] ...I have an H-8/H-19/H-17/H-14 Computer with the 8080A CPU, 48K of RAM and two SSSD drives assembled from kit in 1981; to which was later added a third drive, a Trionyx gold motherboard and another 8K of RAM ... A disk with HDOS-3 was obtained from Mr. Parrott but so far have been unable to get it to run. It starts OK up to BOOT, then the system goes dead, with the interrupt light off from which it responds only to a system reset. I presume there is an interrupt call at that point for which there is no servicing routine.

[Answer from Dan Jerome, Burnsville, MN] When you boot HDOS Version 3, it goes first to ... 8k above low memory, and then drops down to the lowest address available, just like CP/M.

In order for HDOS Version 3 to work on the H8, the computer must be equipped with the ORG-0 card and associated ROM. [Anyone wanting to run HDOS 3.0 on an '8] should be strongly urged to purchase [these items]. After he installs this new hardware, HDOS 3 will operate satisfactorily, as 4 of my friends will testify. Also, it would be better for him to purchase and install the Z80 card while it is still available...

I believe one can still purchase this hardware from the following vendor:

TRIONYX ELECTRONICS CO. / Box 5131 / Santa Anna, CA 92704 / (714) 830-2092

[Win replied,] I want to thank you for your courtesy in forwarding Mr. Jerome's reply to my call for help ... His discussion of the problem was quite clear and bore out what I more or less suspected...

As to changing to an ORG-0 board and the Z80 CPU, et al, as a retiree ... I don't see myself making a significant upgrading to the set I have ... If HDOS-3 had run OK, it would have been useful, but I probably won't change the whole system just to have -3 over -2 ... Anyone want an SSSD HDOS-3 Disk??

Eight-inch drives
[From Thomas Walls, 6360 Montgomery Ave., Philadelphia, PA 19151 (215) 878-4032] **WANTED** HZ 207-42 Dual 1/2 Height 8-in drives.

=====

QUESTIONS and ANSWERS

Q — [On an H-19 or H-89] is there a way to erase the twenty-fifth line other than disabling it? Could you then write to it again without reenabling it?

A — You correctly imply that "disabling" line 25 also erases it, but yes, there is another way. To illustrate, if you go off-line ("OFF LINE" key) and press these 4 keys: "ESCAPE Y 8 SPACE" the cursor will go to column 1 of the 25th line (line must be enabled). You can then erase it without disabling it, by hitting SHIFT-ERASE. An unshifted ERASE can expunge the whole line too, **if** you are in column 1, because it erases from the cursor to the right, inclusive. And yes, you can then still write to the line, even if you move the cursor out of it and go back in later. By the way, don't omit the SPACE bar in the above 4-key sequence. It won't appear to do anything **if** the cursor was **already** in col. 1, but it is needed to satisfy the 4-byte sequence.

Q — I can name a file "TAB1" when copying a file "TABULATE.DAT" with PIP. If I try it with HUG's CP/M File Manager, HFM, believe it or not it comes out as "TAB1LATE.DAT"! Is this a bug?

A — Not really. HFM is designed to default as closely as possible to the original filename. When copying "TABULATE.DAT" to a new filename that is shorter and/or drops the filetype, you must tell HFM

you want (in this case) "TAB1." -- **with** the period -- not "TAB1" or it will keep the rest of the characters as defaults. "TAB1.DAT" (if you wanted that name) would've worked too, because it has a period. Moreover, "TAB1.D" would also work because "defaulting" does not occur **within** an extension. [Unfortunately, while investigating this question I found **real** bugs in HFM! But I also learned some **nice** things about this useful program. See the MISCELLANY column for more on HFM. --HL]

=====

A RARE BUG IN MAGIC WAND EDIT

by Hank Lotz

Finding this little "bug" in MAGIC WAND is remarkable to me because that word processor is so wonderfully engineered and, in general, amazingly bug-free. For five years I ran MAGIC WAND, under CP/M, before this bug in the EDIT portion surfaced; conditions have to be **just right** (or just wrong) for it to occur. So, while the bug is interesting, it is not all that threatening. (There's even a workaround for it, as you'll see.)

It's tough to describe in one sentence, so let's ease into it, shall we? First, before you get too alarmed, please remember this important fact: The bug can happen **only** when a "Disk is full!" message appears in MAGIC WAND EDIT. That does **not** mean you get it **every** time you run out of disk space. But if you **don't** run out of space, you're safe from it.

Okay, what **harm** does the bug cause when it strikes: At **worst**, you can lose the final 128 bytes of a text file when you exit the EDIT program. The **least** damage is the loss of only **one** byte (the last one). But none of this happens unless the disk space is **almost but not quite** enough for your text file.

Now, exactly when and how does this bug occur? As you know, if you type END when you're short on disk space, EDIT says "Disk is full!". One option, then, is to type F (list filenames), and then use EDIT's K (Kill) command to erase one of the files. This frees up room, and you just type END again. Normally -- repeat, **normally** -- that's all it takes, and your file is safely on disk. But **if** disk space is just **slightly** short **at the time a "Disk is full!" appears** our quirk pops up.

To be precise, there must be **128 or fewer** text characters left in memory **after** you see the "Disk is full!" message, for you to qualify for the bug. Anything more than 128 and you're all right. But let's put you in the vulnerable situation. You type your first END and the disk write begins. After a moment it stops and you see "Disk is full!". You glance at the Command Screen for how many characters are left to be written, and you see "Characters: ... 0-In Use"! What!? Zero? If nothing's left to output, why complain that the disk is full? So you go to the Text Screen to check for yourself, and lo! **nothing** is on the Text Screen! Well, what can you do but play the game, so you look at the files, Kill one, and do another END. This time EDIT seems to terminate okay and you're back at a system prompt, like B>. But you TYPE your file to the screen, and the last few (up to 128) characters are missing! (Would it have helped if you "Killed" a bigger file? No.)

Could you have prevented the loss of these characters? Yes, very easily. (Here's that workaround.) Whenever you see "Disk is full!" **and at the same time** you see "0-In Use" the bug is waiting to strike. At that point, **do** go ahead Kill a file to make disk room. But then, **type the W command** (not END). Finally, type END. No characters will be lost.

Let's look inside. At an END, EDIT apparently writes 128 characters to whatever output buffer it uses, and then puts that memory buffer to disk. It refills the buffer, writes to disk, and so on. When it goes to put the **very last** full (or partially full) buffer to disk, and finds the disk full, the **next** END forgets there are still characters in the buffer. The W command forces the buffer to flush.

At the time of the "Disk is full!" message, if **more** than 128 chars remain, no bug will occur. And if there are 129 for example, only 1 char will show on the **Text** Screen. After you clear disk space your next END command **will** write the entire file anytime there is at least one visible character on the **Text** Screen. Also, in that example, only the 129th character would be reported by the **Command** Screen ("1-In Use"). This shows that the previous 128 chars are already in the buffer, and it thus betrays the size of the buffer.

If you'd like to see it with your own eyes, here are "demo" instructions to duplicate the bug using a 90K hard-sectored floppy. Start with an **empty** formatted disk. This assumes you're logged onto drive B: with EDIT.COM on drive A:. First we do some SAVE's to nearly fill the disk.

```
B>SAVE 176 DUMMY1
B>SAVE 176 DUMMY2
B>SAVE 4 DUMMY1K  (to be used later for Kill)
```

This creates two 44K files and one 1k file, and leaves 1k of free space. Now call EDIT and answer Y to the question.

```
B>A:EDIT BUGFILE
BUGFILE is a new file (Y/N)? Y
```

Type MBN after the * prompt, in case you inadvertently use spaces. I do **not** use spaces in this test; EDIT could replace some of them with TAB bytes at exit and foul up our byte count. Also type MCY, to make CR's visible, in case any sneak in. CR's are bad for this test because EDIT adds a line feed, upon exit, for each CR encountered -- you guessed it -- fouling up our byte count. Hit RETURN to go to the Text Screen. Hold down a character, say X, and the REPEAT key for 14 solid, continuous, 80-char rows of X's. Carefully add 32 more X's on line 15. After you type that 32nd X, hit the DL key twice. Do not move the cursor. Now go right to the Command Screen.

There you should read "1152-In Use". It **must** also say "Workspace contains **0 Lines...**" so we're sure no CR's are in this file. With "1152-In Use" and "0 Lines" we have 1024 chars plus 128 extras. Substitute ABC for your last 3 X's. This is arbitrary, but allows for later checking. But be sure the Command Screen stays at 1152 and 0. Give the END command. You should get **"Disk is full!"**, and **"0-In Use"**. Kill the DUMMY1K file and do another END. The B> prompt should appear. Enable display wraparound on your terminal (press OFF-LINE, ESC, v (lowercase), and release OFF-LINE). Now view your file -- B>TYPE BUGFILE. Notice the ABC is not at the end. (If you don't know what "wraparound" is, do the TYPE with and without that step and see the difference. The DIP switch on my TLB is always set for wraparound.)

Now run STAT *.*; you'll see BUGFILE as 1K with 8 sectors, instead 2K with 9. You produced the bug. But if you repeat the exercise with a W command (before the last END command), it will work **properly**. 1025 characters would also have sufficed for this demo, instead of 1152. In contrast, 1153 **won't** give the bug, and that's in keeping with what I've said.

Although W writes the last buffer where END won't, an END is still needed after the W, for housekeeping. It renames the original file to a BAK file, and names the temporary $$$ file to the original name. W writes buffer characters to disk, even if none were on the Text Screen. END **will** function properly if not interrupted by a "Disk is full!" condition.

The moral of the story is use W if your workspace (the Text Screen) is empty after a "Disk is full!". (But turn on the C/R Display with MCY;

a lone CR could make the screen **look** empty.) Then after doing the W, do an END. This won't occur often; the odds against it are high.

=====

VENDOR.UPDATE

CP/M again! I received information from Tom Wolfe and William Derby that CP/M CONNECTION, one of the vendors I mentioned in the last issue, has already bitten the dust. But as a courtesy to customers, the company sent a list of others supporting CP/M. This list included Central Computer Products, of course, but also two others. These were:

Spite Software / 4004 SW Barbur Blvd. / Portland, OR 97201
Workman Associates / 1925 E. Mountain St. / Pasadena, CA 91104

I wrote, but have received nothing, as yet, from the former. However, Workman sent a catalog.

Most of its products are utilities such as a sort program, an MBASIC subset compiler, spelling checker and indexer, and modem program. However, its lineup also includes adventure games (even an adventure construction kit!) and various languages (C, Modula-2, and Ada). Prices are also reasonable for everything but the latter two languages. I think you'd find its catalog worth exploring.

Further, Walt Janowski, in his "The Eight-Bit World" column in **Sextant** #34 (Late Spring, 1988), mentions another one. This is:

Xpert Software / 8865 Pollard Ave. / San Diego, CA 92123 / (619) 268-0112

I also wrote to this outfit, but don't expect to receive anything before this issue goes to the printer.

GENERIC LIQUIDATION. Shortly after the last issue was mailed, Generic Computer Products, Inc. (Box 790, Marquette, MI 49855) sent me a flyer to announce a close-out of its CP/M software for our machines. For more information, write. However, only soft-sector disks are supplied and some of the packages require a Z80.

Micronics. I noticed in an ad in the April issue of **REMark** that Micronics Technology has moved. The new address is: 410 Bellehurst, Montgomery, AL 36109.

MPI Printer Ribbons. Late last fall, Allie Lingo wrote with the address of a supplier for MPI printer parts and service. I didn't have room last issue, but here it is, a little late, perhaps, but hardly never!

Micro Products International / Suite A / 11011 S. Wilcrest / Houston, TX 77099 / (713) 568-6160

Viking Software. Prompted by a note of ANAPRO's Peter Shkabara in **H-SCOOP** #97, I wrote to Grant Gustafson of Viking C Systems. He sent a sizeable package of material along with his two-page catalog. My own interest was in his library for Software Toolworks C/80 compiler under CP/M. But he **also** continues to support HDOS with a number of products.

He carries printer device drivers for the latter for Anadex, C. Itoh, Comet, Epson, Gemini, IDS, MPI, and Okidata models. He also has utilities for moving files between HDOS, CP/M, and MSDOS; an inexpensive database package; sorting source code for MBASIC; an EPROM burner; yet **another** speedup kit; and other good stuff. Quite a bit of the software is available for **both** CP/M and HDOS. The address is:

Viking C Systems / Grant Gustafson / 2243 Belaire Drive / Salt Lake City, UT 84109 / (801) 466-6820 weekdays, (801) 484-9573 evenings and weekends

SEBHC Journal. [From Leonard Geisler] The SEBHC JOURNAL regrets that increased production costs, plus the recent postage-rate hike makes it necessary for us to raise our one-year subscription price from U.S. $15 to $17.50 as of 1 August, 1988. All future issues of the JOURNAL are to be enclosed in polyethylene mailers as soon as possible because so many subscribers have complained of not receiving copies or getting them in mutilated condition. The mailers should eliminate most of these problems. There is no change in the single-edition price ... ($2.50/copy).

You might also like to inform your readers that the JOURNAL has recently revised its' HDOS 2.0 "Programmer's CARE Package" Disc #0, and has introduced a new HDOS 2.0 GAME DISC #1. These 5-1/4" discs are available at U.S. $3.00 each in ssdd H/Z soft sector, and at U.S. $3.66 each in sssd H/Z hard sector versions.... [Thanks for the update, Lenny. The postage hike is also forcing me to double the shipping charge of my keyboard overlays (to $2.00) as of 1 June. But the cost of a subscription to this newsletter remains the same $8.00 through the end of the year. Hank and I anticipated the hike and included it in our price rise last December. And for those of you who aren't familiar with the friendly competition, the **Journal**'s address is: **SEBHC Journal**, 895 Starwick Drive, Ann Arbor, MI 48105. -Ed.]

ANAPRO. [From Peter Shkabara] ...[B]est wishes for your work as editor of the newsletter. In your capacity as such, would you be interested in distributing some of my software which is no longer commercially viable. I am willing to release it for noncommercial distribution (public domain). You are welcome to charge a distribution fee to cover costs. The following items are available:

WSPATCH - patches to WS3 and 3.3 to implement H19 keys.
WAND - patches to Magic Wand to display functions on 25th line.
KRESMOD - patches to Heath BIOS 2.2.03 and 04 to allow use of a single BIOS with the Kres, Najay Systems and the Analytical Products speed mods.
PIEPATCH - adds improvements to the Toolworks PIE editor.
CLOCK - software package to support the CDR or Analytical Products real time clock.
EPSON - a menu utility to set the MX80 functions.

...You may wish to update your readers that the TIM2 and REP3 are no longer available. The demand was too low to justify a new circuit board run. A limited quantity of REP3s is still available from Quikdata. Some parts and instructions for the TIM2 are still available.

...My recommendation to the H8/H89 community is to form a "demand group" for products. For example, CDR had discontinued the RAM[drive] board, but when sufficient demand had built up, they made another run. It was only an additional 30 boards, but the point has been made. It is possible for the Staunch 8/89'er to be a clearing house for such demand. Readers [could] submit their requests or orders to the editor. These would be collected till the quantity justified an order. The manufacturer could ship a bulk order to the Staunch 8/89'er at volume pricing as appropriate. The price to the end user would be adjusted to cover costs.

[Thanks for the software donation, Peter, including the bonus HDOS version of the Epson printer menu utility; readers will find ordering details on the green insert. Your recommendation about the "demand group" is one I'm following up, as readers have

ANAPRO Corp. Products
(Courtesy of Peter Shkabara)

CP/M software

CLOCK	Support software for C.D.R. or Analytical Products real time clock.
EPSON	Menu utility for MX80 functions.
KRESMOD	Patches to Zenith BIOS's 2.2.03 and 2.2.04 for most 4 MHz speed mods.
PIEPATCH	Patches to Software Toolworks's PIE 1.5(d) editor to select tabbing and BAK files from within program; uses DDT.
WAND	Patches to Magic Wand 1.11 and 1.12 to display function keys on 25th line and reset terminal to power-up state when exiting; uses DDT and SUBMIT file.
WSPATCH	Patches to WordStar 3.0 and 3.3 to use function keys and shifted keypad or Epson MX80 or FX80 printer; uses DDT and SUBMIT file.

(Available as a package on either two (2) standard hard-sector disks or one single-sided soft-sector disk.)

HDOS software

EPSON	Menu utility for MX80 functions; ASM source included.

(Available on either standard hard-sector or single-sided soft-sector disk.)

Frazer Users' Group Library
(Courtesy of Terry Hall)
HDOS only

(Of the 30 hard-sector disks prepared, **Staunch** presently has 11.)

Vol 000	Catalog disk.
Vol 101a	MBASIC games.
Vol 102a	MBASIC utilities and demos.
Vol 103a	MBASIC games and utilities.
Vol 104a	More MBASIC games.
Vol 105a	Further MBASIC games.
Vol 106a	MBASIC games and utilities.
Vol 107a	Assorted MBASIC programs.
Vol 140a	ABS games.
Vol 141a	ABS utilities.
Vol 142a	More ABS utilities.

(Available on standard hard-sector or soft-sector disk; please specify format desired in the latter since more than one hard-sector can be squeezed onto one soft-.)

Hank Lotz's Utilities
(Written by Hank Lotz)
CP/M only

LP	Command utility for H-14 printer.
MAKFIL	Experimental sequential file creation utility.
SALVAGE	File recovery utility for Magic Wand.

(Available on standard hard-sector and soft-sector disk; these programs are more fully described in this issue's MISCELLANY column.)

QDELETE	Utility for QUERY!3 to change all records in a file to "deleted."
TXT2Q3	Compiled version of Kirk Thompson's utility to convert ASCII files to QUERY!3 format.

(Available on standard hard-sector and soft-sector disk; these programs are more fully described in issue #5, p. 3.)

OMDOS/SMALLDOS
(Written by Skip Chambers)

Clone of HDOS 2.0 using less memory and without the system overlay structure; requires MTR-89 or -90; now includes INIT.ABS to format disks. NOTE: This package is **not** in the public domain; **Staunch** serves as a distributor.

(Available only on two hard-sector disks for $12 postpaid; this package is described in issue #6, p. 8.)

TXT2Q3
(Written by Kirk L Thompson)

MBASIC utility to convert ASCII files to QUERY!3 dbms format; program has a bug which crashes it when destination .DTB file exceeds 32K, but a workaround is available. A complete rewrite is in progress and will be supplied free to all "registered" users when finished.

(Available for HDOS and CP/M on standard hard- and soft-sector disk or in MSDOS XT-format [program conversion required in the latter case!]; this package is described in issue #3, p. 1.)

Utilities for HDOS
(From OMAHUG library)

BASE5	Convert or calculate numbers in decimal, octal, split octal, hexidecimal, and binary.
CRASH	Disk file recovery utility similar to DUMP.
DD	Brief directory lister with file sizes.
DK.DVD	H37 (soft-sector) device driver.
RESTORE	File undeleter.
Z19/89 DIAGNOSTICS	Complete suite of programs to test and troubleshoot TLB and CPU boards.

(Available on either standard hard- or soft-sector disk.)

How to Order

Your cost depends on what you supply. If you send:

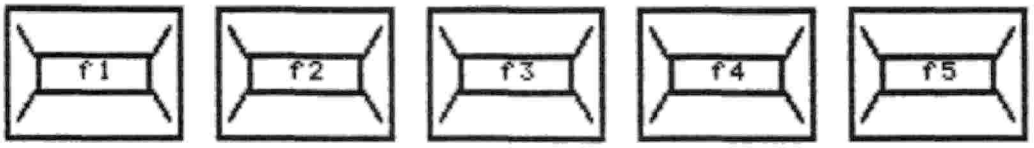
Formatted disk and self-addressed, stamped return mailer -- $2.00
Formatted disk without mailer -- $4.00
No disk or mailer -- $6.00

Formats available are standard (SS/SD) hard-sector and 40-track (48 tpi) soft-sector, single- or double-sided, for both HDOS and CP/M. Please clearly indicate the format you are supplying or require. If you desire soft-sector, I will pack multiple items as described above onto one disk for the **single-disk** charge. But I will not subdivide a disk.

Send your order to:

Kirk L Thompson / **The Staunch 8/89'er** / #6 West Branch Mob Hom Vil / Rte 1 / West Branch, IA 52358

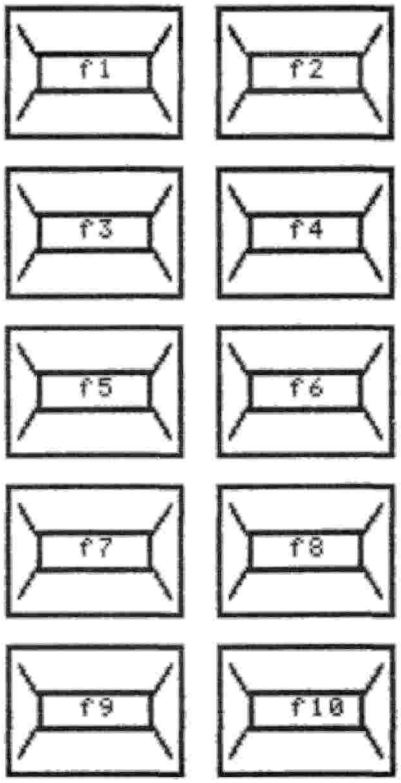
NOTE: As of 1 June, 1988, the mailorder price of the keyboard overlays described on the reverse side rises to $6.00 per half dozen postpaid.



===> ARE YOU TIRED? <===

Are you tired of trying to remember what the special function keys on your keyboard mean? Or referring back to the program manual every time you bring up a little-used software package? Or remembering all those new shift, alt, and control functions when you installed that IBM PC hardware emulator? Wouldn't it be nice to have something you could just drop over the keyboard when you load the odd inexpensive or public-domain program with the key definitions already on it? Especially something inexpensive?

Well, that something is a "generic" keyboard overlay and it's here! It wraps around the keypad (numeric on the -19/88/89/90 and -100; function on IBM-compatibles) and stretches across the top of the keyboard. Plenty of space is available on it for you to mark key definitions, program name and version, and any other information you might deem necessary during use. All overlays are printed on heavy card stock, die-cut, and include double-faced tape for more permanent attachment to the keyboard.



PRODUCT INFORMATION
==========

KO-19: overlay for H/Z-19/88/89/90.

KO-100: overlay for the H/Z-100 series (not IBM).

KO-150: full-sized overlay for IBM and compatibles (-148,-150, etc).

KO-151: miniature overlay for only the function keypad of IBM and compatibles.

PRICING
(per set of six)
==========

KO-19, -100, and -150: $5.00, postpaid!

KO-151: $3.00, postpaid.

===> AVAILABLE NOW! <===

Order from:

Kirk L Thompson
#6 West Branch Mob Hom Vil
West Branch, IA 52358

---> Iowa residents please include 4% sales tax! <---

If you have a custom requirement, write to me with details.

 For: HUGCON 87

already discovered in my editorial. But for further information about ANAPRO, contact Peter at: ANAPRO Corp., 6905 El Camino Real #4, Atascadero, CA 93422 (805) 466-1589. Hours: Mon-Sat 9:30 to 6:30, Sun 11 to 5 (Pacific). -Ed.]

More about TMSI. [From Lee Hart] Just got my first issue of "Staunch", and was very impressed. You pack a lot of high-tech questions and answers in a very small space...

...I have moved from my temporary residence with friends in Kalamazoo to a more permanent address. All correspondence to me or (the new) TMSI should be addressed to: Lee Hart, 28612 Middle Crossing Road, Dowagiac, MI 49047, phone 616-782-3980.

Thanks of the brief write-up on TMSI [in the last issue]. The publicity in H-Scoop, Buss, Sextant and Staunch have all helped put pressure on the other members of [old] TMSI ... Terry Donnelly (the Controller) [received] the rest of the business records, and he passed copies on to me. This, plus letters from TMSI customers has let me compile what I hope is an accurate list of outstanding orders. They are being filled from stock as fast as I can, with profits from new orders being used to purchase any missing parts not left in the old TMSI's inventory.

I have received many calls and letters of encouragement from loyal H89'ers who don't want to see their machine abandoned. Several made more substantial contributions. In particular, I would like to thank Ross Towbin for loaning me his H89; Lee Foltz for donating a tube of 27C64 EPROM's needed for the Supersets; and Jim Quillin for printing up a dozen Superset manuals for me.

...I'm not yet filling Superset orders from stock; I'm filling backorders at about 4-5 a week, with about a dozen to go. I do accept COD's, but not for phone orders...

I am no longer associated with Technical Micro Systems and am NOT responsible for its debts. But I have the remaining parts inventory. If customers also send copies of their claims to me, I am voluntarily trying to fill old orders where I can. You got it right in [the last issue of] "Staunch"... [Thanks for the update, Lee. And thank you **very** much for the **Superset/Superfont** chip set you shipped; very impressive. All I need now is the time to install it! I suggest that one way readers can reward you for picking up the ball old TMSI fumbled is to pay you upon delivery of the latter's orders. While there is no legal obligation on either you (to ship) or the receiver (to pay), I think there is a moral burden on the latter. Besides, paying would further ensure your continued tenure as a valued vendor to the 8-bit community. So hang in there! -Ed.]

SMUGH P.D. Library. [From Daniel Jerome, club CP/M librarian] The St. Paul/Minneapolis Heath Users' Group (S-M-U-G-H) has decided to distribute their library of 8-bit public domain CP/M and HDOS software in order to encourage nonmember owners of H89/Z90 and H8/H19 computer systems to profit from our store of software treasures.

As of 31-March-1988, our CP/M Library will have 46 90k disks available, with a new disk being offered each month. Our HDOS Library will have 34 400 sector disks available, with a new disk coming out from time to time. Many of the programs have been written by local talent, and are not available from any other source. Some programs come from national public domain generating groups.

We are not selling the public domain software per se, we are merely distributing it. The distribution fee is $5.00 per disk, which includes disk, handling, and postage. We feel this is a reasonable charge.

If you are interested, it is suggested that you first purchase the CAT disk. For CP/M-80 we have 86k in 5 CPMLIB ASCII files, plus one 28k MASTER.CAT file. For HDOS we have 58k in 4 HDOSLIB ASCII files, plus a 28k MASTER.CAT. Make checks out to SMUGH. For those people who only have hard sector drives, special arrangements will be made. The CAT disk also contains an order form that may be used to place orders...

SUMMARY: To order a public domain disk, please let us know exactly how you want your disk to be formatted. The order form ... furnished on the [CAT] disk will aid in securing needed information about your computer system. After you have the CAT files, you can check what is available and order only the disks you want ... Only ... entire disk[s] may be ordered. These disks may be ordered by number. Each disk has its own integral disk .DOC file, which will help you to determine what you want...

To order CP/M disks write:

SMUGH CP/M Librarian / 801 E. 132nd St. / Burnsville, MN 55337

To order HDOS disks write:

SMUGH HDOS Librarian / 246 E. Bernard St. / West St. Paul, MN 55118

[In later correspondence, Dan mentioned that many files provided on standard hard-sector (90K) disks are supplied in a "crunched" format to take up less space. An executable "uncruncher" is included so you can recover them before use. -Ed.]

CHUG P.D. Library. The largest local HUG group in the nation also provides nonmember access to its library. Jim L. Nielsen recently provided me with squeezed master catalogs on both CP/M hard- and soft-sector, along with NSWEEP.COM to convert them to a readable format.

The CP/M library numbers 112 disks as of November of last year. Some of the material is duplicated in the SIG/M library (more on this group next time), but it includes languages (STOIC, ALGOL-M, SAM76, JRT Pascal [without documentation], MUMPS, and COBOL), utilities, games, assembler libraries, a spelling checker, and on-disk versions of **>CHUG** (the club newsletter) and index disks to same.

The HDOS library numbers 122 disks. The software here includes the ubiquitous games; some modifications of SYSCMD.SYS for both versions 1.6 and 2.0; utilities; an assembler preprocessor; database packages; FORTRAN and MAC-80 utilities; MX-80, C. Itoh, and Okidata printer device drivers; LISP and Tiny Pascal; Steve Robbins's EDIT19 editor; and, once again, on-disk versions of **>CHUG**. But my space here is simply too limited to even scratch the surface of CHUG's immense holdings. They're definitely worth your look.

The March issue of **>CHUG** noted that "[e]ach disk is priced at $3.50 for CHUG members, $4.50 for nonmembers plus $1 shipping for [the] first disk and 50 cents for each additional disk in the same order. Since some disks are restricted to members only, you must specify your CHUG membership number with your order." The catalogs are normally supplied on MSDOS-format disk, so specify the format you require. CHUG is none too speedy on delivery, either, so some patience on your part will be necessary.

For orders or further information, write to:

Capital Heath Users' Group / Box 16406 / Arlington, VA 22215-1406

HDOS 3.0 enhancements. [From Rick Streeter, Route 1, Box 424, Bigfork, MN 56628] I have modified HDOS 3.0's system command processor (SYSCMD.SYS) and peripheral interface program (PIP.ABS) to what I feel give a user a much better interface to HDOS 3.0. While HDOS as distributed by Bill Parrott

supports generic 8080-type processors (8080, Z80, etc.) and terminals, I have decided to support the H-19 ... exclusively due to its excellent characteristics. The enhancements which I have made are as follows:

1) "USER" areas and numbers are fully implemented...
2) New commands for SYSCMD.SYS...
3) New PIP switches...
4) Other commands and switches...
5) The ctrl-A command line editor now is What-You-See-Is-What-You-Get (WYSIWIG) using the H-19 terminal.
6) There is an escape from an error on COPY (Ignore, Retry, Abort choice). This was formerly an abort error only.
7) A few innocent but messy bugs in PIP and SYSCMD have been fixed.
8) The special function keys have been enabled in SYSCMD.SYS. This was done so they won't usually interfere with an overlay program ... As long as the [latter] eats the trigger keys the sequence is never seen by SYSCMD.SYS and is not acted upon. The RED key TYPEs SYSHELP.H19 to the terminal, the ERASE key clears the screen, and the other keys either run a *.ABS or a *.BAT program as determined by the user...

There are many other code refinements which can only be appreciated through use of this enhancement package. I feel that I have helped make a gem more of a jewel.

I originally wrote the programs using Z80 code throughout but by popular request have the package available in both Z80 ... and 8080 versions. The Z80 version is not much bigger than the original HDOS 3.0 version and uses the same memory locations for loading and ... buffers. The 8080 version is MUCH larger and PIP loads 1000A bytes higher than the Z80 version.

Send an initialized HDOS 3.0 diskette in any H-17 or H-37 format (40 track, single- or double-sided is less prone to read errors) and $15.00 and I will return it with my package on it. The files included will be: SYSCMD.SYS, PIP.ABS, EDIT.ABS (modified for USER), INIT.ABS (no cute messages), ERRORMSG.SYS, HELP., HELP.DOC, SYSHELP.H19, SYSHELP.DOC, LABEL.ABS (enhanced), and INSTALL.BAT (an installation program). Please specify if you want the Z80 or the 8080 version, or add $3.00 for both.

=====

A Turbo Pascal Peg Game

by Norman L. Riger

The puzzle described in this article places thirty-two pegs and one blank space on the screen of a Heath H89 computer system. The object of the game is to remove as many pegs as possible by making one peg jump over another peg and landing on a blank space. Only vertical and horizontal jumps are allowed. Each peg is represented on the screen by an "O" and each blank space is represented by a period. A peg can only be moved by jumping over another peg. Any peg may be selected by using the arrow keys on the keypad. The selected peg is moved by pressing the "5" key and one of the arrow keys (2, 4, 6 or 8) to indicate the direction of the jump. The "O" representing the peg to be jumped is replaced by a period after the jump is performed. Illegal moves are prevented by the program's code. Termination occurs when a lower case "q" is typed by the operator.

The program is written in Borland's Turbo Pascal, version 3 and the source code [Listing 1] is included to encourage beginning and student programmers. The code is meant to be modified, translated to other programming languages and adapted to other computer systems. Although the peg game may be used as is, my philosophy is that the more changes you make, the more you will learn.

Turbo Pascal is powerful and easy to use. Programs are compiled without exiting the integral full screen editor. The compiler, editor and source code fit on a single 90k disk. Errors are detected by the compiler and the editor's cursor is placed on the faulty code. The source code is 4k and executable code is 12k in length.

When typing in the code, omit the section numbers at the left, place each statement on a separate line and arrange the format so that each "begin" starts in the same column as the corresponding "end". Test each change as soon as it is written to simplify debugging.

Section 1 names the program and author and provides the date of completion.

Section 2 defines the type gg, which is needed to define the two dimensional array gs.

Section 3 defines six variables (byte instead of integer in order to save memory space) used for loop counters (a and b), screen locations (h and v) and array locations (ha and va).

Section 4 defines two char variables used to store commands from the operator.

Section 5 defines a two dimensional array used to store the arrangement of 33 peg positions. The array contains 49 cells, but sixteen (four in each corner) are not used.

Section 6 is the actual start of the program and begins by clearing the screen (writing an Escape E) and unshifting the keypad to the numeric mode. I've configured the editor to use the keypad for commands and I need to shift it into the numeric mode.

Section 7 contains two loops that fill all 49 cells of the array with 79, upper case O.

Section 8 sets the screen location variables to the upper left hand corner of the screen design and sets the middle location of the array to a period.

Sections 9 through 12 set four cells in each corner of the array to 32, a space, as they are not used.

Sections 13 through 15 contain two loops that draw thirty-two O's and one period on the screen and position the cursor on the period in the center.

Section 16 gets the first command from the operator and matches the current array variables with the current screen variables.

Sections 17 and 18 check for an input of 8 and if it is found move the cursor position upward if it not already as high as it can go and if it won't move into one of the unused positions in the corners. The next three pairs of sections work similarly for inputs of 2, 4 and 6. The cursor is only allowed to move to legal positions.

Section 25 checks for an input of 5 and if it is found waits for another command from the operator to indicate the direction of the intended jump.

Sections 26 through 31 check for an input of 4 and if it is found perform a jump to the left if it is legal. A jump to the left cannot be made from the two leftmost columns of the screen arrangement. The jump can only be performed if the selected position contains a peg, the next position to the left also contains a peg, and the second position to the left contains a period. A jump results in changing the contents of three cells in the array, changing two O's on the screen to periods, and changing one period on the screen to an O. Sections 32 through 49 similarly control jumps in the other three directions.

Section 50 ends the program when an input of "q" is detected. Notice that the keypad is shifted back to its original configuration.

The peg game may be too difficult for beginners because of the many levels of nested IF..THEN statements. Omit sections 25 through 49 and practice making changes until a better understanding of the operation is achieved.

There are other ways to write this program and valuable experience may be gained by experimen-

ting. Consider initiating a jump by pressing the shift key and an arrow key. Try using a nine-by-nine array with all four borders filled with spaces. Change the size and shape of the screen arrangement.

Such as a triangular version of this game with one peg on the top row, two on the second row, three on the third row and so forth. This version requires jumps in six directions instead of four because the pegs in each lower row are not directly underneath the pegs above.

Record all the moves in memory to allow later analysis of the game or save them in a disk file. Allow the correction of mistakes by allowing the operator to undo the last move or sequence of moves.

This program is derived from a character font editor that designs foreign language alphabets and calculates the matrix codes of each custom character (Russian, Arabic etc.). Once the basic principles are understood, the program may be converted to a number game where numbers are shifted up and down and to the left and right until they are all in order. A more complex game, similar to Pacman, results by enlarging the array, using graphic characters to form a maze and adding monsters that move independently at their own speed on the screen.

A disk containing the peg game (both source code and executable version) and other programs is available by sending $5.00 to Norman Riger, 3148 Holmes Ave. So., Minneapolis, MN 55408-2629, but be sure to indicate whether you want the H89 CP/M (40 track hard sector or 80 track soft sector) or IBM MSDOS version.

Listing 1.

```
{1} program peg_game; {by Norman L. Riger 2/7/88}
{2}  type gs=1..7; {adapted to Heath H89 3/28/88}
{3}  var a, b, h, ha, v, va: byte;
{4}   response, peck: char;
{5}   gg: array[gs,gs] of byte;
{6} begin write(#27#69); write(#27,'u');
{7}  for a:=1 to 7 do begin
      for b:=1 to 7 do begin
       gg[a,b]:=79;
     end; end;
{8}  v:=8; h:=24; gg[4,4]:=46;
{9}  gg[1,1]:=32; gg[1,2]:=32; gg[1,6]:=32;
     gg[1,7]:=32;
{10} gg[2,1]:=32; gg[2,2]:=32; gg[2,6]:=32;
     gg[2,7]:=32;
{11} gg[6,1]:=32; gg[6,2]:=32; gg[6,6]:=32;
     gg[6,7]:=32;
{12} gg[7,1]:=32; gg[7,2]:=32; gg[7,6]:=32;
     gg[7,7]:=32;
{13} for a:=1 to 7 do begin
      for b:=1 to 7 do begin
       gotoXY(h,v);
{14}   write(chr(gg[a,b])); h:=h+4; end;
     h:=24; v:=v+2; end;
{15} v:=14; h:=36; gotoXY(h,v);
{16} repeat read(kbd, response);
      ha:=(h-20) div 4; va:=(v-6) div 2;
{17}  if response='8' then begin
       if v>8 then begin
        if gg[ha,va-1]<>32
{18}     then begin
          v:=v-2; gotoXY(h,v);
       end; end; end;
{19}   if response='4' then begin
        if h>24 then begin
         if gg[ha-1,va]<>32
{20}      then begin
           h:=h-4; gotoXY(h,v);
       end; end; end;
{21}   if response='6' then begin
        if h<48 then begin
         if gg[ha+1,va]<>32
{22}      then begin
           h:=h+4; gotoXY(h,v);
       end; end; end;
{23}   if response='2' then begin
        if v<20 then begin
         if gg[ha,va+1]<>32
{24}      then begin
           v:=v+2; gotoXY(h,v);
      end; end; end;
{25}   if response='5' then begin
        read(kbd,peck);
{26}    if peck='4' then begin
         if h>28 then begin
          if gg[ha,va]=79 then begin
{27}       if gg[ha-1,va]=79 then begin
            if gg[ha-2,va]=46 then begin
{28}         gotoXY(h,v); write('.');
             h:=h-4; gotoXY(h,v);
{29}         write('.'); h:=h-4; gotoXY(h,v);
             write('O'); gotoXY(h,v);
{30}         gg[ha,va]:=46; gg[ha-1,va]:=46;
             gg[ha-2,va]:=79;
{31}    end; end; end; end; end;
{32}    if peck='6' then begin
         if h<44 then begin
          if gg[ha,va]=79 then begin
{33}       if gg[ha+1,va]=79 then begin
            if gg[ha+2,va]=46 then begin
{34}         gotoXY(h,v); write('.'); h:=h+4;
             gotoXY(h,v); write('.'); h:=h+4;
{35}         gotoXY(h,v); write('O'); gotoXY(h,v);
{36}         gg[ha,va]:=46; gg[ha+1,va]:=46;
             gg[ha+2,va]:=79;
{37}    end; end; end; end; end;
{38}    if peck='8' then begin
         if v>10 then begin
          if gg[ha,va]=79 then begin
{39}       if gg[ha,va-1]=79 then begin
            if gg[ha,va-2]=46 then begin
{40}         gotoXY(h,v); write('.'); v:=v-2;
             gotoXY(h,v); write('.'); v:=v-2;
{41}         gotoXY(h,v); write('O'); gotoXY(h,v);
{42}         gg[ha,va]:=46; gg[ha,va-1]:=46;
             gg[ha,va-2]:=79;
{43}    end; end; end; end; end;
{44}    if peck='2' then begin
         if v<18 then begin
          if gg[ha,va]=79 then begin
{45}       if gg[ha,va+1]=79 then begin
            if gg[ha,va+2]=46 then begin
{46}         gotoXY(h,v); write('.'); v:=v+2;
             gotoXY(h,v); write('.'); v:=v+2;
{47}         gotoXY(h,v); write('O'); gotoXY(h,v);
{48}         gg[ha,va]:=46; gg[ha,va+1]:=46;
             gg[ha,va+2]:=79;
{49}  end; end; end; end; end; end;
{50} until (response='q'); write(#27,'t'); end.
```

MISCELLANY

HFM's Undocumented Features: [From Hank Lotz] To answer a question for this issue I delved into the CP/M HUG File Manager, HUG Part No. 885-1246[-37]. (See today's Q/A column.) In so doing I may have exposed the world to new bugs, but the heading on this piece doesn't always mean **bad** features. I found good ones too. "Undocumented" means this info was not found in the doc, nor in the on-screen help file, nor in any of the separate help files you call up for each menu command by hitting f1. One good discovery has to do with Listhex, the command that shows a file in hex, and in ASCII to the right of it. In the ASCII portion, **some** characters are in reverse video and you aren't told why. The significance is that a highlighted character represents an ASCII byte, but with its high bit set. This can be useful to hackers and it's great to know it's there; I didn't know it before. Another good fact I found is more like a proper-usage tip than a new feature. I had copied a file to a different name on the same disk (and under the same user number) but HFM's directory display didn't show the new file

at the end of the operation, so I doubted the copying actually took place. Later I discovered you have to do a Newdisk command to update the display. It's an extra step, but at least knowing how to work the program increases its value. Also, if you Flag some files and want to clear all the flags at once, they tell you to use Unsort. Well the Newdisk command does this too (whether you want it to or not!), and has the advantage of not needing to Sort afterwards.

HFM's bugs show up in the Copy and Newdisk commands. It seems something is wrong with the user-number handling. Newdisk is supposed to change drives **and/or** user numbers, but I can't get it to work unless I change only 1 spec at a time. For example, when I'm in 0B (user 0, drive B:) and I type "1C" as a Newdisk command, it should go to user 1, drive C:, but it goes to user 1, drive B:. The Copy bug is even worse. Just as one example, logged into user 0 on drive B:, I tried to copy a file AAAABBBB.ZZZ to user 1, drive C:. For my destination spec, I used "1C:" (in accordance with the instructions in the Copy help screen). Instead of getting a file AAAABBBB.ZZZ in user 1, drive C:, the file was created on drive B:, but in user 1, and its name was actually 1C:ABBBB.ZZZ, the first 3 A's being overwritten by the 1C:, and the colon was part of the filename, which is illegal in CP/M! I even tried, letter-for-letter, their Copy-help-screen example using NAME.TYP, and ended up with 2B:E.TYP **as a filename** on drive A:, user 2, instead of NAME.TYP on drive B:, user 2. Finally, if you try Copy with insufficient disk space, you get a "cannot write" message, but HFM **does** write -- a zero-length file, which you must erase to get rid of. I wish HUG would fix these things up; this is otherwise a useful utility.

Two New Enhancement Patches for Derby's SD.COM: [From Hank Lotz] SD.COM is from the Derby CP/M Utilities package that **Staunch** described in Issue #3 (p.5), reviewed in #4 (p.6), and printed a patch for in #5 (p.1). Bill Derby informs me: "I have recently made another patch for SD.COM; this may be of particular interest to those with a hard disk. [Of course it's not **limited** to hard disks --HL]. It allows the listing of files on a user number other than the one currently logged." This patch of Bill's is 48 bytes long. And back in September I myself wrote a 10-byte patch for the SD program which puts more space between its columns on the screen. Rather than publish these patches I'll send both of them (free) to any **Staunch** subscriber who requests them **and** sends me a self-addressed, stamped envelope (SASE). Both patches are made using DDT (like the one in **Staunch** #5, p.1) so some understanding of that patch procedure is assumed. Two 8-1/2 x 11 sheets list the patches (with **very** few sentences of reading matter). My address is: Hank Lotz / 2024 Sampson St. / Pittsburgh, PA 15221.

Hank's Utilities for CP/M. If you're using **Magic Wand** or an old Heathkit H-14 printer, Hank recently sent me some utilities for distribution which you could find of value. One is a program to recover work if you accidently QUIT **Magic Wand**'s editor, do a hard reset, or suffer a BDOS error. After you've SAVE'd the contents of memory (using CP/M's built-in command), Hank's SALVAGE utility eliminates most of the binary garbage from the beginning of the saved file, recovering the ASCII text so you can reedit. He provides MBASIC source and a compiled version. I've included it as standard equipment on my own 8-bit **PeachText** disks!

The second program is a utility for use with an H-14 printer under CP/M. It let's you set the number of characters per line, the number of lines per inch, sends one to 99 line feeds, issues formfeeds, or flushes the buffer, all from the keyboard. Commands are entered on the command line, so it's fast. Hank supplies you with the executable program and its FORTRAN and assembler source.

As a bonus, Hank has thrown in an MBASIC utility which creates two different types of experimental sequential files. The files created by this program could be used during program development and testing. And, again, Hank includes both MBASIC source and a compiled version. For ordering information, see the insert.

HDOS. Saving the best till last, you may recall my suggestion in issue #5 for a writing campaign to save HDOS. I received a letter from Bob Ellerton (former HUG manager), as this issue went into final assembly, which bears very positively on the question.

In his most relevant paragraph, he writes: "[Heath president Bill] Johnson suggested that we release the original HDOS 2.0 operating system and updates to the public domain. Soon, HUG will be placing the data on the HUGPBBS system for download. The HUG board can be reached by any active member of the Heath Users' Group by dialing (616) 982-3956."

Therefore, with a substantial part of my goal achieved, I would like to thank all of you who wrote to Bill Johnson, HUG, and ZDS president John Frank at my prompting. Actually, you were writing on your **own** behalf because, in my opinion, the survival of HDOS as a useable system required that Heath place it in the public domain. I strongly recommend you write to Heath president Bill Johnson (Heath Co., Benton Harbor, MI 49022) and thank him for this most generous action on our behalf.

I also talked with Bob Ellerton on the phone a few days after I received his letter. I thanked him and we discussed several related topics. Among them, Bob mentioned that, like Parrott's HDOS 3.0, the UltiMeth HDOS assembler will probably be required to assemble the source for version 2.0. Quikdata (2618 Penn Circle, Sheboygan, WI 53081-4250) carries this assembler for $75 postpaid.

Finally, I suggest another writing effort. HDOS still lacks more sophisticated program development tools for BASIC, FORTRAN, and assembler. Release into the public domain of the MBASIC interpreter and compiler, FORTRAN compiler, and M80 macro assembler by Microsoft would be most welcome. The address is:

Mr. William Gates / President / Microsoft Inc. /
16011 NE 36th Way / Redmond, WA 98073-9717

Microsoft will be **less** inclined than Heath to release its "obsolete" products. Perhaps (but only perhaps!), if you and I can impose on Microsoft to set a precedent with HDOS, there is some chance a few years from now that it might do the same for its CP/M products. So even if you are only running the latter, I recommend writing. You could well be helping yourself in the long run!

THE STAUNCH 8/89'er, created by Hank Lotz, is a quarterly newsletter on 8-bit H/Z computers. The editor is Kirk L. Thompson; #6 West Branch Mob Hom Vil; Route 1; West Branch, IA 52358. Subscriptions always start and end with the calendar year. Rate: $8.00/year. (Overseas, add $2.) Single copies: $2. Make checks payable to "Kirk L. Thompson". **Staunch** pays authors for their articles; write for an author's guide. It also accepts commercial ads for a modest fee; contact the editor. EOF