Issue #6 THE STAUNCH 8/89'er Jan-Feb-Mar 1988

PORT TO PORTAL --Editorial

The last two months have been extremely interesting. And as you quite rightly expect, preparing this, my first, issue of **Staunch** has been the most absorbing. One of the things I've done is simplify the apparatus Hank originally developed. Assembling and printing the newsletter, though, is still much as he described it in issue #1. I plan to discuss my **Magic Wand/PeachText** "SETUP" section later this year. As an experiment, I'm also increasing the number of lines per page from 66 to 88; let me know if you have trouble reading it.

While you're responding to that, and prompted by William Allen's letter appearing later in this issue, I would also like to poll **all of you** to find out just how long you've had your H/Z computers. **Please send me a postcard with that information.** I would also like you to add whether or not you have a collection of **REMark** back issues and just how far back that goes. Mine starts in late '79, which gives you some idea of my antiquity! Hank and I have **presumed** that these are readily to hand, but maybe that isn't so! Your input will assist Hank and me in planning articles this year and next and I'll report back to you on the results in the next issue.

However, I've also been able to deliver on **one** of the things I mentioned in my column last time. I can announce **Staunch**'s first **paid** author, Dan Jerome; congratulations, Dan! But I haven't found anyone to cover HDOS 3.0, yet. And you'll find no advertising supplement, either; several vendors have expressed interest, but I've received nothing concrete as yet.

One point Hank suggested I mention is the matter of back issues. They can be had from me for the old subscription rate of $5. This includes all five of our first "year's" issues. Just sent a check to me at the address at the end of this number and I'll mail them to you first class. And notice the last issue number of your subscription printed on your envelope this time.

But read on and enjoy! And I welcome your feedback.

--Kirk L Thompson

THE EIGHT-BIT R/W -- Letters

Terminal chip behavior

[From Corky Kirk, Hilo, HI--a merging of his two notes] Thanks for your card of 3 Dec 87 - tried your suggestions regarding [my] Z89A problems:

1) upon boot, screen comes [up] with bottom portion a series of vertical columns
2) after the boot commercial and [CP/M's] "A" prompt, I get 13 lines of A> 00 (as I hit RETURN) then 12 lines of A> (again, as I hit RETURN) and then it repeats the pattern. If I try to list a program or something, the 13 lines that have the A> 00...are all garbled.

[Tried booting the distribution disks, but N]o luck--not a software problem.

Finally took the bull by the horn and tied into the guts of the Z89A. It **was** on the terminal logic card [as you also suggested], U409 to be exact. It was an **unmarked** 2114 [chip] (and the 5th of the six 2114's I tried on the board). Anyway, replacing it with one out of the spare terminal board I bought from Al Davis...brought up the ol' Z89A as it should be....

Again, thx for your inputs. It is really appreciated. [Glad to be of assistance, Corky. And thank you for writing about your difficulty; this is the first I've heard of this specific one. Now we **all** know what to look for! -Ed.]

HDOS overlay SCALL

[From Robert McClure, Jr., Cumby, TX] Here is the answer to the first "Q minus A" [question] in Issue 5: The [HDOS] load overlay call in ASM is SCALL .LOADO which is 377, 010 in octal, or FF, 08 in hex. There should be two calls in quick succession, one for each overlay. The first will be preceded by MVI A,0 (076, 000 octal, or 3E, 000 hex) or some other means of getting 0 into the accumulator, and followed by some conditional jump to check for errors. The second call will be preceded by MVI A,1 (076, 001 octal, or 3E, 01 hex). Hope that helps.

I think you and Hank have the right idea. I think that **Staunch** is going in the right direction. I certainly hope you can get to at least bi-monthly very soon. And I hope I can help provide some material for it. [Thanks, Bob, for the answer to the question. And I, too, would like to see **Staunch** go bi-monthly ASAP, but that depends on increasing circulation. In fact, I've reconsidered that question since my column in #5 and will do that the January **after** we reach 500 subscribers! -Ed.]

Free documentation and miscellaneous stuff for sale or trade

[From Terry Hall, 516 E. Wakeman, Wheaton, IL 60187, (312) 665-4594]...enclosed is a copy of a list of programs for which I have extra copies of the documentation. Got these with [a] couple extra machines I bought. Glad to give them to anyone who can use them or perhaps trade for something...

Microsoft Basic 80 (2 copies)
Microsoft BASIC Compiler User's Manual (vers. 5.3)
Microsoft utilities (HDOS version)
Touch-Typing Tutor (HDOS)
Magnolia CP/M 2.24 operating system user guide
HRUN HDOS Emulator for CP/M (HUG part # 885-1223)
Monitor MTR-88 software ref. manual
HDOS 2.0 software reference manual
DSM-240 installation and operator's guide from Kres
(their 2/4 mhz speed-up module)(2 copies)
Compuserve information service starter kit
Super Ram 89 instruction manual from CDR

I also have 3 green H-19 (or H/Z89-90) CRT's for sale. One is new. All are fine...[Would] Like $20 apiece for tubes plus shipping or $50 for all.

...I also have a good quantity of 8" SSSD disks to give away or trade for something. I've chosen to use...DSDD 8" [consistently]--makes back-up easier, etc. Just ask receiver to pay shipping costs unless they want to trade me some 5 1/4 hard-sector disk for them. [See elsewhere in this issue for a rundown on vendors for **new** hard-sector disks. And Terry adds the following in a postcard: -Ed.]

Couple of things for possible inclusion in next STAUNCH. (1) Have you or anyone in Staunch-land (except Lenny G. [editor of **SEBHC Journal**]) ever received a 19/89 Superset from Lee Hart (old TMSI)? I had one on order for about a year and finally got my money back...But I'd really like to get the hardware. (2) Has anyone gotten Jim Teixeira's UD.DVD running with HDOS 3.0? (3) Has anyone gotten HDOS 3.0 to run with C.D.R.'s Ramdrive? [Terry, for the first, see the "Miscellany" column later in this issue. But can anyone answer the other two? -Ed.]

Graphics on the H-14?

[From T.J. Skollingsberg, Salt Lake City, UT] Concerning the question [in issue #5] "Is there a modification to the H-14 printer so it can be used for graphics?" I'll say yes, if you can get Heath to release the programming data on the custom CPU used in the H-14. It should at least be able to print the entire graphics character set of the H-19 as well as underline (without linefeed) and print the next line without a space. Software support shouldn't be too tough via the H8-4 serial port either. It'll just take some time to do.

Derby releases another utility

[From William S. Derby, Livermore, CA] I have finally finished the programming and documentation of the CLE Command Line Editor program I mentioned in a letter that was printed in Staunch #5...I did some more work to provide a program that makes installation a lot easier. The CLE program gives a natural and powerful facility for the recovery and correction of command lines in a CP/M system on the H89 (and it could be adapted to most computers with a CP/M 2.2 system)...

...Three command line buffers have been identified to provide the editing snd recovery facilities needed most, with a minimum of space. These buffers hold the current command line, the most recent command line, and a command line selected by the user.

The installation program for the CLE command line editor allows it to be installed anywhere in memory that does not cause a conflict with any other elements of the system...When installed the CLE program communicates intimately with the BDOS code standard in any CP/M 2.2 system. All functions of the CP/M system are undisturbed by the presence of the command line editor; and since the CCP is unaffected, the CLE program will also work in systems like ZCPR that replace the CCP. A special version of the CLE program with the editor incorporated into the BIOS also exists for the Heath/Zenith H89 computer...

More descriptive information on the CLE command line editor, and on the earlier Derby CP/M Utilities (SUB/SD/CMP/COPY) is available from the author. The CLE Command Line Editor and its documentation may be ordered for $12.00 (including postage in the U.S.). The Derby CP/M Utilities are also available for $12, but both may be ordered together for $22.00. Orders indicating preference of any standard H89 hard or soft sector 48 TPI CP/M disk format should be sent to W.S. Derby, P.O. Box 2041, Livermore, CA 94550 (415) 443-1741) [Thanks again, Bill, for information on your software and, of course, your continued support of our machines. -Ed.]

Media restrictions and A/D

[From Larry Oneson, 120 Tangmer Crescent, Markham, Ont., L3R 6Y3 Canada] A question to you and other subscribers. If licensed software is no longer available for hard sectored drives, are there any legal restrictions to copying existing software. I'm thinking specifically of things like CBASIC, WordStar, Magic Wand etc., it would be nice to be able to use this stuff now that I am not restricted to storage limitations [on my new drives]?

One more thing. Has anyone out there done anything with the A/D H8 board that Heath sold in the early days? I'm thinking of trying to monitor all my electrical, hydro and heating utility equipment and possibly phone calls. I would appreciate any info at all. [Larry, if you hold a license to the software, you may move it to any media you need to in order to run it on your system. For example, when I converted my system to soft-sector over four years ago, I moved virtually **everything** to the new media. The only exceptions were HUG's "DND" (because it required a modified version of HDOS 1.5, useable only on hard-sector), J.J. Thompson's HDOS to/from CP/M utilities (which recognize only hard-sector), and a public domain version of DUP (also hardware specific). As for your second question, I'm sure someone out there has the answer. -Ed.]

Anapro's 4 MHz mod and C.D.R. RAMdrive conflict

[From Charles G. Bejin, Cheyenne, WY] I found the write-up in Issue #4 about the H89 4MHz speedups very interesting. I've been using Anapro's mod for several months now and am pleased with it. It has worked very well with both Magnolia and CDR double density boards and their CP/M versions. I haven't tried HDOS.

Adapting the 4MHz software modifications was quite straightforward using Anapro's instructions etc. Speeding up the terminal caused some glitches. Some programs using graphics and/or cursor addressing just didn't go. Examples are HFM from Heath User's Group and Magic Wand. But, my speedup consisted of software adjustments and replacing one chip on the TLB. More drastic surgery to the TLB may be required, and right now I'm chicken.

While I'm in a writing mood, have you or any of your readers any experience using a CDR SuperRam 89 with a Magnolia Microsystems 77316 double density disk controller board? Mine works great as long as the SuperRam 89 is designated the last drive in the system, i.e. drive "M". If I designate it drive "A", problems arise accessing the second and successive 5 inch drives. The first 5 inch drive and all 8 inch drives work fine. The problem is identical whether running at 2MHz or 4MHz. [Charles, your problem with C.D.R.'s "SuperRAM 89" will be discussed in this month's "The 8-Bit Iowan." And thanks for the positive feedback on Anapro's product. -Ed.]

Two for the price of one

[From Richard J. Kiessel, Germantown, MD] I greatly enjoyed Hank Lotz's article entitled "Zero-Length CP/M Files: Creating and Using Them" in Issue #5 of the Staunch 8/89'er. It was while following his directions on adding labels to my disk that I came up with a slightly different method of creating the zero-length file...

...I use the PIP command to create the zero-length file. This is done by [creating a] file from the console. Since the console file is empty, a zero-length [file] is created. The syntax for the PIP command line is:

```
PIP fname=con:<RETURN>
```

As soon as the command line is accepted by the computer, the cursor is positioned in the left hand column and advanced one line. The computer is then ready to accept the text of the file I want to create. It knows when I am done when I type a <CONTROL>Z (^Z). Since what I want is an empty file, I immediately type a ^Z and the zero-length file is created.

I believe that the other use for zero-length files that Hank referred to was that of restarting a program without changing the data which it has stored from its previous use--such as being able to reclaim those hundreds of lines of basic I forgot to save before exiting MBASIC. (However, it must be noted that this trick will not work on programs which use overlays such as SUPERCALC and WORDSTAR or programs which initialize upon start up.) Since I use this trick to fool a program into restarting without loading, I find it more logical to call it RESTART.COM rather than Hank's /.COM.

Flying the H-8/H-25

[From William E. Haivala, Grand Island, NE] I will be sticking with my H-8 and THE STAUNCH 8/89'er for some time. I have been forced into the Turbo charged 'Comet' 286 environment at work, but as a hobby, the H-8 does just fine. I also fly a 49 model Piper Cub. The change of pace is great in either, or both machines.

I am looking for a chip set to get 'near letter quality' print from my H-25 Printer. Does anyone out there have info on this? I also would like to get an H-8-compatible copy of XMODEM on H-17 Hard-Sectored format disk. I'm using a Hayes 300 SmartModem.

I'll make contact with the OMAHUG 8-BIT IOWAN'S group to see what they have in XMODEM software.

Sounds like they plan to keep their library up for a while. [Bill, you might try Larry Fina of Fina Software, 16144 Sunset Blvd, #3, Pacific Palisades, CA 90272. He sells replacement PROM and SRAM chip sets for the H-25. -Ed.]

Old timers vs. beginners
[From William H. Allen, Anza, CA] I would appreciate it if some of you old timers would think back to when you first got your H-89 and write some articles for "BEGINNERS" such as me.

I bought my 2 H-89's because they appeared to satisfy all of my needs and were cheap. I am slowly learning how to use them, but, it has been a struggle. For instance how do I use the two external drives with HDOS? I solved using them with CPM, but, not HDOS. If someone would explain it in one or two syllable words I would be most appreciative. [Excellent point, Bill! Both Hank and I **have** inadvertently neglected the total novice. Our sincerest apologies! Indeed, your letter points out users which **Staunch should** be assisting. So I would welcome contributions from those of you who have already "bootstrapped" yourselves to a higher level of expertise. Others like Bill need help and I'm frankly not sure **I** can provide it. My own novice days were altogether too long ago! But to aid Hank and I in planning novice and advanced articles, please participate in the readership poll **Staunch** is conducting as described in my editorial. -Ed.]

=====

THE 8-BIT IOWAN
by Kirk L. Thompson

As I suggested last issue, this time we will explore the productivity benefits of RAMdrive systems. Specifically, I'm running C.D.R.'s **SuperRAM89**, with 1 Mbyte of RAM, under both HDOS and CP/M. I'm quite impressed with it and will use it as an example of this class of enhancement. However, C.D.R.'s product has a number of odd wrinkles and I'll discuss those near the end of this overview. The first question, though, is: What exactly is a RAMdrive?

By Definition. A RAMdrive is a memory device which emulates a disk drive. That's it, pure and simple. Implementing it is another story, however! In the case of C.D.R.'s hardware, the RAMdrive is one or two (piggybacked if two) boards, each containing up to 500K in chips, which go into an expansion slot on the left side of the '89's CPU board. One peculiarity of C.D.R.'s system is that you **must** move the Z80 CPU from the CPU board to the basic RAMdrive board. A custom ribbon cable jumper between the two is provided. If you already have a 4 MHz module installed, room becomes tight. In my case, with Kres Engineering's speed-up (as described in issue #4), I discovered after some use that a mounting bolt for the flyback transformer had punctured the insulation on this jumper. I cut off the bolt's excess length and placed several layers of electrical tape over the hole in the cable.

But where there's hardware, there must also be software to run it. Under both HDOS and CP/M, you lose approximately 1K of working memory. In the case of CP/M, this is all in one block at the top of memory. So you **must** MOVCPMxx 63 to make room for the driver **before** you SYSGEN your disk. For HDOS, C.D.R. provides a utility which patches HDOS.SYS to reserve half a K at the top of memory. There is also a device driver (RD.DVD) which passes commands to the former module.

One other thing to consider when looking at RAMdrives is **which** drives it permits you to set up. These are often limited to drives which are **nonexistent** on your system. Under CP/M, this isn't too much of a hassle since you can PIP the materials you wish to run from floppy to the RAMdrive, log onto that drive, and proceed with abandon. Unfortunately, that isn't the case with HDOS. One advantage which C.D.R.'s product has over others is that you can set up both CP/M's and HDOS's primary drives (A: and B: or SY0: and SY1:) as RAMdrives.

When considering RAMdrives, I also strongly recommend getting as much RAM as you can afford in the hardware. The memory should at least equal **twice** the capacity of your disk drives. That way you can emulate two drives, if necessary for the software you will be running on them.

The RAMdrive Advantage. The principal benefit which RAMdrives provide is raw speed. But where RAMdrives really shine is in disk-intensive activity. Examples include saving long files with **Magic Wand/PeachText** and sorting large databases under **QUERY!3**. A while back, before I installed the 4 MHz speed-up, I timed saving a 5,100-word file when exiting 8-bit **PeachText**. On soft-sector floppy, that took 2 minutes and 31 seconds. On RAMdrive, it only took 13 seconds! The reason for the over 1,000 percent improvement is because **Magic Wand/PeachText** verifies that the file it writes exactly matches the contents of memory. This requires a lot of reading and writing.

RAMdrives also save time when sorting files which are larger than memory. An example of this is the database for **Staunch** which I maintain with Hoyle & Hoyle's **QUERY!3** package under HDOS. This file occupies 258 256-byte sectors and contains 288 subscribers. Sorting this on soft-sector floppy at 4 MHz takes 1:55; on RAMdrive it only takes 24 seconds! Like the preceding example, this process is disk intensive because the file is broken into nine sections which must be saved on disk and merged back together as the sort proceeds.

Drawbacks. Nothing is free. There is a cost to pay for the speed you obtain from RAMdrives and I'm not just referring to out-of-pocket price. One you must contend with is loading files from floppy. This obviously depends on how many you need to do the task at hand. In my case, for most word processing with 8-bit **PeachText**, I move 37 (both program and data) and the SUBMIT-driven operation takes a little over two and a half minutes! However, this is certainly an extreme case. For quick and dirty jobs I work completely from floppy to save time.

I should add, parenthetically, that C.D.R.'s CP/M system preserves its files through a system reset, which makes part of the wait I endure worthwhile. I can hard-reset, boot other CP/M or even HDOS floppies to do other things, then boot back into the RAMdrive to pick up where I left my opus. This is particularly useful when I'm investigating software and writing up what I find on the fly. But I have to be careful that I **don't** boot a disk which re-initializes the RAMdrive. For this reason, I prefer the software C.D.R. includes which prompts you to erase the RAMdrive rather than that which performs that function automatically on cold booting.

A second thing you must be careful to do is **back up** to floppy the data files you've changed. If you forget, you've lost **everything** when you flip the power switch off! In those cases where files are easily defined, as under **QUERY!3**, I have a batch processor file which I automatically execute before I quit. One other thing I've noticed is that if you forget to back up **once**, you usually don't forget again!

C.D.R.'s Kinks. I mentioned that C.D.R.'s system has some peculiar wrinkles. Reader Charles Bejin notes one of them in his letter appearing elsewhere in this issue. But I'll turn, first, to HDOS.

I pointed out above that the CP/M software preserves the contents of the RAMdrive through a hard reset. This is **not** the case with the HDOS version. But if you discover you've forgotten something immediately after leaving, you can "warm" boot back into the RAMdrive (when SY0:). However, as I discovered recently, one of the things you would expect to work in that situation doesn't. In this particular case, I had to rename a pair of device drivers to use my daisywheel printer with **QUERY!3**. So I shuffled the names, exited with BYE, and rebooted.

But when I went to print, **QUERY!3**'s CALC program informed me that it was going to dump the data I wanted as hardcopy to a "NULL" device. In other words, it would be thrown away! Apparently, when I rebooted, the modified HDOS did **not** reconstruct the table of valid device drivers as it does booting from floppy. I eventually renamed the drivers directly on floppy, then cold-booted and was able to get the printout I wanted.

You also have to be careful not to copy files directly between floppy drives under the HDOS version. I haven't tried it myself (I'm chicken!), but reputedly the destination disk is corrupted! If you need to transfer files between floppies, copy them first to RAMdrive, then out to the destination.

Lastly, you are also limited to using C.D.R.'s system only with HDOS 2.0. The patching program mentioned above is specific to Heath's last release and isn't useable with Parrott's HDOS 3.0. Personally, I wish that weren't the case!

The CP/M version of the software has its own set of wrinkles. The biggest is limited compatibility with Magnolia Microsystems' CP/M as Bejin observes. When using programs **other** than those provided with the operating system, I usually get protracted clicking from floppy drives and sometimes read errors. The "fix" is to **only** use built-in commands (DIR, REN, etc.) and system utilities (PIP, STAT) when accessing floppies. This problem does **not** occur under Zenith's 2.2.03.

Another wrinkle is how different BIOS's handle SUBMIT files when run from RAMdrive. C.D.R. includes specific instructions to use one SUBMIT file to both initialize the drive on cold-boot and automatically copy your files to it. This works fine for Zenith's 2.2.03. But it **doesn't** under Magnolia's or, according to James Pilarski (in a letter in **Sextant** #28), C.D.R.'s own BIOS 2.91-M. The "fix" to both is to **only** use the name of the RAMdrive initialization program (ARAM or SRAM) as the cold-boot command. Then you must call SUBMIT and its command file from B: or C: with the keyboard. This may be mildly aggravating, but a minor hassle given the benefits.

Wrap-up. RAMdrives are certainly one way of improving the productivity of your 8-bit system. The raw speed they provide is, in my opinion, as significant as that supplied by 4 MHz modules. And positive side effects I haven't mentioned thus far are the almost total elimination of floppy drive noise and decreased wear and tear on your floppies and disk drives. You only use them when initially reading files to the RAMdrive and when you back up your data.

In the Queue. Next issue I turn to a method of enhancing the communication your system has with its printer. Not counting humans, printers are undoubtedly a computer system's slowest peripheral. A printer buffer is intended to relieve that situation and I'll review Heathkit's SK-203 for you, including how easily it goes together. So see you then!

=====

Q minus A

(An open-forum "Q-only" column. Answers are solicited from readers!)

Q — What changes need to be made when converting an HDOS assembly language program to CP/M -- and vice versa? Perhaps with explicit enough instructions, even non-assembly-language programmers could apply it to a given listing.

Q — How do you use the Z-89-11 card with CP/M 2.2.03 -- no BIOS mods in Heath's package, and the port disappears from the CONFIGUR program?

=====

H89 SCREEN BLACKOUT PROBLEM

by Daniel N. Jerome
801 E. 132nd St.
Burnsville, MN 55337

I have been a proud owner of an H89A since 1981, when I bought an entire system for an atrocious sum from our Minneapolis Heathkit dealer. The machine has since performed flawlessly for me, and has satisfied my needs in many ways. As a member of our Twin Cities (Minneapolis and St. Paul) HUG group, me and my six disk drives have been able to satisfy the requirements of many other members who own an H89 or a "gussied-up" H8, since we can match any system except the 8-inch. I now serve as the Club Secretary and one of two CP/M Librarians.

A few days ago, my computer screen went completely blank right in the middle of an important task. The right and left edges of the screen coalesced into a vertical line in the center. Then the screen became completely dark. Somehow, I was able to keep my head and sent the file I was working on to disk successfully. During a time like this there is a strong tendency to panic. One thinks of down time, technician charges, etc, and it could be a nightmare.

When I contacted our HUG Club hardware specialist, the friendly old "computer doctor," I learned some valuable information which would be applicable to anyone owning an H89. And so, with his permission, I pass the information on to you in the hope that it will also prevent you from suffering unnecessary grief.

You may be happy to know that just because your screen turns black does not signify dire consequences, such as a burned-out CRT. With a little tender, loving care, the CRT will operate without problems for more than 10 years. Fortunately, most blackout problems can be traced to something fairly simple such as corroded terminals.

Herein, I present a "triple whammy" remedy for curing the problem. Even if you have experienced no such problem and you want to perform some preventive maintenance, proceed according to the following steps:

WHAMMY ONE:

(A) Turn off AC power to the computer.

(B) Open the cover and note the 3-wire cable going from the power supply across the tops of the serial interface and disk drive controller cards. This cable plugs into the CPU board on the top on the right side of center. Sometimes the terminals get a little corroded and need to be cleaned.

NOTE: The best product I have found for cleaning terminals is called "TWEEK." It is available from Crutchfield, 1 Crutchfield Park, Charlottesville, Virginia 22906 for a mere pittance of $16.00 for a small bottle. The top cap comes with a brush. If you call toll-free 1-800-446-1640, they will send you a free catalog. The first step in troubleshooting the problem is to clean the terminals of this cable, and the pins attached to the CPU board.

The best way to completely clean the terminals is with TWEEK. This clear liquid comes with a handy brush in the cap. This brush will enable you to apply the chemicals to the exact place needed the easiest way possible. Radio Shack also has some satisfactory chemicals, but they come in spray cans and one should avoid spraying the circuit cards since the chemicals could cause damage.

I emphasize that there are two sets of terminals to clean: the male pins on the CPU board and the female spring contact pins on connector P516. One easy method of cleaning the connector pins is to take an ordinary pencil eraser and carefully rub them clean. Do not use an emery board or sandpaper, as this will scratch the pins and cause more corrosion to occur. In order to clean the spring pins, one must remove them from connector block P516 according to procedures described below.

WHAMMY TWO:

(C) If the first procedure does not solve the problem, there is a second part to our "triple whammy" that will most likely do the trick. Look at the left side of the CPU board. On the top left of the CPU board, you will note two cable connectors, mounted one above the other. Heath does not call out the cable assemblies, but the connectors are called P515 on top and P514 on the bottom. I call the top cable W10 and the bottom cable W3. These connectors are plugged in vertically. Sometimes these cables cause problems. The cables were furnished pre-assembled in the H89 kit and also in the Z90 version, but the terminals were **crimped**, not soldered. The thing to do is this:

(1) Remove the top cable, W10, from CPU board connector P515, and unplug it from the TLB board. Be careful not to damage the ground wire, which may be screwed to the TLB board.

(2) Put the cable on your workbench and remove one wire at a time, so that you will not make a mistake when the wires have been treated and it is time to replace them. To make things even easier, you may want to make a sketch of the cable assembly, showing which color wire goes where.

(3) To remove a wire, put the cable flat against your work table and press a small, flat bladed screwdriver against the metal spring that shows on the side of the connector block. While pressing in the spring, slip the wire out of its slot. I used my steel portable vise, which helped me to hold wires, etc, while both hands were busy.

(4) One wire at a time, carefully solder the cable wire to the spring connector, being careful not to get too much solder on the terminal, or to burn the lead wire. In case too much solder comes, use a piece of copper braid to suck up the excess. The terminal will not go back into the connector block if there is too much solder in the way.

(5) The next step is to carefully slightly open the spring terminal (about 1/32nds of an inch) using a tiny needle nose pliers and/or a tiny flat-tipped screwdriver. The purpose of this is to slightly increase the connector block pins brings to bear on the male pins of the CPU board.

(6) Brush TWEEK on the spring terminals and the male terminals attached to the CPU board. It will dry immediately.

(7) Replace the top cable assembly (W10).

(8) Remove cable assembly (W3) from the bottom connector P514 and perform the same soldering procedure on both ends. Be careful when removing this cable, since it may be tangled with other wires below the board. If necessary, you should be able to perform the operation without removing the cable from the computer, by just letting the connectors dangle.

WHAMMY THREE:

(D) Just because you have thoroughly cleaned several connector pins in your computer, does not mean that this is the end of things. Those connector pins are "just itching" to begin corroding again as soon as you close the H89's cover. I present a third whammy procedure in order to provide even more effective and complete preventive maintenance.

(E) Unplug connector P515 from the CPU board and examine the male pins in the CPU board. Notice that they are **round** pins. Round pins such as these have caused problems on the power supply, as indicated in my earlier article (**Staunch 8/89'er**, Jan-Feb-Mar, 1987 (#2), "Beefing up the H89 in Three Steps.") since they do not make a very firm connection with the spring pins in the cable connector block. A round pin provides a slim contact surface, in comparison with a much more complete surface provided by **square** connector pins.

NOTE: Most of the problems come from the top cable assembly. Reasons for this will become clear by checking out the area using the schematic diagram.

CAUTION: Performing the following steps may be hazardous to the health of your computer unless you know what you are doing. If you are a novice kit builder, or have never built a kit before, I highly recommend that you seek assistance from people with electronics or soldering experience.

(F) To perform the modification which replaces either one or two connectors built into the CPU circuit board, first purchase the necessary parts. Heathkit will not have them. Henry Fale at QUIKDATA, 2618 Penn Circle, Sheboygan, Wisconsin 53081-4250 (414)452-4172 may be able to help. You also may be able to purchase parts from an electronic parts jobber if you live in a large city, or even a radio/television repair shop if you live in a small town. Sometimes the pins come separately. Again, the connectors that we want to replace are those **soldered to the CPU board**. These connectors support cables W3 and W10, i.e., plugs P515 and P514. Again, only **square pins** will do the job.

(G) After buying the parts, it will be necessary for you to do surgery. Remove the CPU board from the computer, and remove all the plug-in boards from the CPU. Lay the CPU board down carefully on your work table over a soft towel or cloth. If you have an H89A, the rear of the board will be covered with a metal shield. Remove the shield. Using a roll of **copper** de-soldering braid or a small, mechanical solder sucker, de-solder the connector(s) and remove the old ones. Do not try to economize on the solder sucker braid: the **copper** style works best. Do not discard the old connector(s).

(H) Compare the new connector with the old connector. Note the missing pin if you have an '89A. This is not a mistake. It is called a keying pin, which insures that one can plug in the associated cable only in one position. If your pins were purchased separately, take a heavy-duty wire cutters and cut them to about the same length as the original. If you have to cut the pins after soldering, the static shock could damage nearby chips. Then take a large, long nosed pliers and force the pins into the holes provided in the connector. [Keep the pins as square as possible during this process. -Ed.] Then simply solder the new square connector(s) to the appropriate place(s) on the CPU board. Use a magnifying glass and visually check out each solder joint. Then reassemble the CPU board and attach all accessories and restore the computer.

(I) Now you are ready for your computer to take you another 100,000 miles.

If for some reason you do not have your H89 computer manuals and still desire to perform this mod, just send me a stamped, self-addressed, legal-sized envelope at the address above and I will do my best to provide you with enough data to do the trick, if I have to sketch it myself. If you have any questions, please do not hesitate to write me.

Until we meet again, dear reader, keep the faith! Heathkit built a superb computer when it designed the H89. Time is a thief. Corrosion is a problem common to every man-made machine.

=====

QUESTIONS and ANSWERS

Q — I have heard of "logical End Of File" and, at other times, "physical End Of File". I thought an EOF was an EOF. Could you please set me straight?

A — A file has a physical length, determined by the number of sectors allotted to it on the disk. The last of these sectors is the **physical** end of the file. But the end of meaningful information in the file may occur before this because many times a file doesn't need all the space in its last allotted sector(s). When this occurs in an ASCII text file, the file's **logical** end under CP/M is designated by a CTRL-Z (ASCII 1A Hex), which may fall before the sector's last byte. [On the other hand, HDOS simply fills out the final sector (between the logical and physical ends of the file) with ASCII 00 Hex. So CTRL-Z does **not** mark the EOF under HDOS. -Ed.]

Q — How necessary is a flowchart in writing programs?

A — Although in some circles flowcharts apparently aren't used as much these days, to me (Hank Lotz) they are frequently a must. When I write a program with a lot of logical branching, it's easier to work from a sheet of paper where I can see blocks connected with lines and arrows than it is to try to keep it all in my head. A flowchart is a tool for organizing your program so it doesn't turn into a kluge. It is something you draw up first, and **then** write the program. Some people find it harder to draw a flowchart than to write a program directly from scratch, and some think the flowchart is supposed to be drawn **after** the program is completed. It can be difficult to flowchart a program once it's written, but three motivations for that can be 1) to serve as a guide to users, 2) to aid in future modification, and 3) to facilitate debugging. (Avoidance of the latter predicament is an excellent argument for doing the chart first.) If flowchart writing is distasteful to you, and you're good at logical or spatial visualization, and your programs, when completed, are not tangled messes, forget the chart.

Q — How different are the mechanics of disk file directories under CP/M and HDOS?

A — References to existing articles may be the best answer here. This subject has been nicely covered by Dave Pelowitz (a former OMAHUG member) in **REMark** Issue 45, Oct. 1983, p.74 (HDOS), and Issue 46, Nov. 1983, p.31 (CP/M). Pat Swayne also shows HDOS and CP/M directory examples in **REMark** Issue 33, Oct. 1982, p.15, where he talks about recovering deleted files. Take a look also at the reference given in the next answer, below. Most readers probably have **REMark** back issues -- a proven asset.

Q — Is there a good use for the 3 unused tracks in nonbootable CP/M disks?

A — A good point. You're talking here about at least 7K bytes going to waste on unused system tracks -- tracks 0, 1, and 2 on a 5-1/4-inch diskette. An obvious use for the space is to store data, but the problems with this may preclude our wanting to even bother finding an access method. The space is not addressable via the directory, so an assembly program would be needed to get at it, and that program itself would occupy space elsewhere. While it's true that "elsewhere" could be a central boot disk in Drive A servicing multiple peripheral data disks with but one copy of the program, there are still other drawbacks. Those tracks cannot be copied with PIP, and they are vulnerable if the disk is ever SYSGEN'ed, or DUP'ed to, absentmindedly. If you remain determined, an interesting **REMark** article mentions other unused sectors (and touches upon an access method); see the Feb 1984 issue, page 48. Maybe our readers can send in additional thoughts.

=====

A CLOSER LOOK AT SOME MBASIC RESOURCES

by Hank Lotz

Instruction manuals are meant to be read, but such an activity can stir up mysteries. Does it really mean what you think it means? Sometimes you have to read between the lines to grasp the entire idea. At those times, "field" experience is the best way to confirm your understanding of concepts you've only glimpsed. From my own "field" days I'll offer here a constructive finding or two about MBASIC's resources, including a useful subroutine. For the record, my MBASIC is version 5.21 for CP/M. Let me begin with a hypothetical exercise.

Suppose you are writing a program and, among other things, you want it to report the size of a disk file to your screen display. First your program has to determine the file's size. For that, you might have it read and count the records one by one, and upon reaching the end of file, compute the file's length in kilobytes. For a long file this could take forever and three days. Obviously no good, so you go to the manual in search of a more effective way. There you notice an "LOF" (length of file) function, and for a minute you think you hit the jackpot! The book says LOF returns the true length of file **if** the file has no more than "one extent". Your hopes are dashed -- you wanted it to work for **any** size file. (Actually, it will; they just don't explicitly tell you you need extraterrestrial programming to do that.) And, horrors, if you're not sure what an "extent" is, you're referred to a CP/M manual! The MBASIC manual also talks of **accessing extents**, but you only know how to access **records**. That's the last straw, so unfortunately, you shy away altogether from using LOF. But take heart.

The MBASIC routine below shows a way of employing the LOF function to find the size of any disk file. In order to use this canned algorithm you don't really need much background information, but here's enough to complete this one little paragraph. Every disk file has **at least** one "extent". Each extent is a 32-byte entry **in the directory** of the disk, and contains information about the file, including its name. The extents tell the system where on the disk the contents of a given file are located. Longer and longer files require more and more directory extents, because one extent can map only up to 128 of the 128-byte sectors (or 16K bytes). Extents do not address individual sectors; instead they address groups (or blocks) of 8 sectors (1K) on hard-sectored disks, and groups of 16 (2K) on soft. If your MBASIC manual speaks of "accessing" an extent, merely realize this: whenever your program accesses any **record** in a user file, the appropriate directory **extent** automatically gets accessed also -- the one containing the location of that record. Once an extent is so accessed, a subsequent LOF call (by your program) returns a value it picks up from that extent. That value is the **quantity of 128-byte sectors** pointed to by that directory extent. I use the term "sectors" to differentiate from user-defined "records". For on this next point, too, the book could have been more explicit: the value LOF returns is **not** the number of **logical records** whose length a user originally defined (if other than 128 bytes) in a random file when he created it. In other words, the LOF function **always** deals in 128-byte sectors. (This is **not** so with the LOC function, as we shall see later.)

In order, then, to be able to get the length of **any** file with the LOF function, we really have to access **all** the extents pertinent to that file, and then add up all the file sectors referenced by those extents. That's what this MBASIC listing does. If you want the file size in K, multiply by 128 (bytes per sector) and divide by 1024 (bytes per kilobyte). Or in other words just divide by 8. As presented here, my routine just sums 128-byte sectors.

```
100 ON ERROR GOTO 280
110 LINE INPUT "Enter the file name -- ? ",FILEN$
120 REM -------- FILE EXISTENCE CHECK -----------
130 OPEN "I",1,FILEN$: CLOSE #1
140 REM ----- OPEN THE FILE (RANDOM ACCESS) -----
150 OPEN "R",2,FILEN$,128
160 REM ----- CP/M-80 FILE SIZE CALCULATION -----
170 KT%=0 : T%=0
180 GET #2,1
190 IF LOF(2)=128 GOTO 200 ELSE 210
200 KT%=KT%+1: L%=LOC(2)+128: GET #2,L%: IF EOF(2)
     GOTO 220 ELSE 190
210 T%=LOF(2)+128*KT%: GOTO 240
220 T%=128*KT%
230 REM ----------- OUTPUT SECTION ---------------
240 PRINT: PRINT "File Name=  ";FILEN$
250 PRINT "File Size= ";T%;" sectors.":PRINT
260 CLOSE #2: END    'Exit the program.
270 REM -------- ERROR-TRAPPING SECTION ----------
280 IF ERR<>53 OR ERL<>130 GOTO 310
290 PRINT: PRINT"Sorry, file  ";FILEN$;"  does not
     exist."
300 RESUME 100
310 ON ERROR GOTO 0
320 END
```

The hard core of this lives in LINES 170 thru 220. The routine (or subroutine, if you will) works for zero-length files, maximum-sized files, and everything in between. And although this same subroutine will report the size of a random **or a sequential** file, it always opens the file as "random" to jump through the sectors with ease. (If you are working with a sequential file, you still must use the routine **as is**. But you can always, for example, CLOSE your file after the routine, and reOPEN it in a sequential mode -- "I" or "O".) We need the ease of the "R" mode because we want to look only at the first sector of each extent until we get an end-of-file indication. This means only sectors #1, 129, 257, 385, etc. LINE 200 locates these cumulatively, using LOC to give the previous record number accessed (which is what LOC really does; my manual was wrong before their errata-sheet correction). However, if an extent of fewer than 128 sectors is ever encountered, we need look no further nor even seek an EOF. So, LINE 190 tests for a less-than-full extent. The variable KT% keeps a running count of **full** extents, and T% is the total number of 128-byte sectors in the disk file. Now let's tie up some loose ends.

Remember I said LOC (not to be confused with LOF) returns a **record** number. I'd like to emphasize that it gives, not a "sector" number, but the number of the last **user-defined** logical record accessed, when used with our random file. Of course, records may be defined as 128 bytes, and so be "equivalent" to sectors.

Also it is important to understand that L% (as an argument of GET) causes GET to read the L%th record, but this will only be a record having whatever length was defined in the OPEN statement. If the record size was set smaller than 128 (in the OPEN statement), we would GET smaller records. Thus the count of GET's (KT%) could go too high, resulting in failure of the routine for files 16K or larger. That is why in LINE 150 I specify 128 -- even though 128 is the default -- so that there is no temptation for a programmer to specify some other record size previously associated with his file. It must be 128 for the routine to work with all files. I also want it to be clear that the aforementioned practice of accessing only the **very first** sector in each extent is not merely whimsical, it is a **requirement** for this method to work consistently.

Notice that, although GET statements are employed, no FIELD statement is necessary for the purpose at hand, as we are not yet retrieving any data **from the file itself**; nor are we writing any. Of course, if a FIELD statement were present it would not hurt anything.

A word about the file existence check. When a file is OPENed in the "R" mode (a random file) and no file currently exists by whatever name was entered, it will be created on the disk. For example, if a user wrongly types BOOK.TZT instead of BOOK.TXT, a file called BOOK.TZT is created. To avoid this we do the existence check (LINE 130) by momentarily OPENing the file as "I". This is read-only mode and cannot create the file. If the file is not found, the error-trapping routine will tell us. (LINE 100 enables that routine.) If the file is found we can CLOSE it, then confidently OPEN it as an "R" file. Other advantages of an existence check are obvious.

By extolling the merits of fruitful experimentation, as I seem to have done at the outset today, I don't mean to discourage in the least the reading of the manual. Read the instructions first, and reread them every few months or so. If you're like me, you'll understand them better the second time. But in the meantime, have a field day!

=====

MISCELLANY

TMSI's Demise: If you've had your 8-bit system for any length of time, you've no doubt heard of Technical Micro Systems, Inc., of Ann Arbor, MI. It has been a longstanding vendor to the '8 and '89, providing hardware and software to those of us still using H/Z's "obsolete" equipment. Sadly, I received a letter the middle of January from Lee Hart announcing its bankruptcy. The same story also occupied a good page and a half in **both H-SCOOP** #95 and **BUSS** #149. A number of us, myself included, are out because of undelivered goods or unreturned, in-warranty, parts.

If this is your situation, I recommend assembling documentation on your loss (cashed checks, letters, invoices) and sending **photocopies** (not the originals!) with cover letters to:

Terry Donnelly, Controller / Box 7227 / Ann Arbor, MI 48107

Regional Chief Postal Inspector / Central Region / Chicago, IL 60607-5401

Consumer Protection Division / Attorney General / your state capitol

Be sure to mention in each letter that you are also sending the same material to the other addresses. While there is **no** guarantee you will recover your money or property, these mailings will at least register your claim. And **do not** send any more orders to the Ann Arbor addresses, either that above or 366 Cloverdale; they are now black holes swallowing everything they encounter! But Lee Hart is also not down and out.

Lee wrote that one of the reasons for the bankruptcy was a falling out among the partners who formed Technical Micro Systems. This left him basically managing the whole thing on his own. But he has now started his **own** business, selling some of the products which Technical Micro Systems developed. These include the Z89-37 soft-sector controller, Flicker-Free kits, Low-Power kits, Write-Hand-Man, (maybe) the H-1000 dual-processor board, and thousands of loose parts. He is also attempting to make good on the back-orders for the SuperSet terminal chip set. If Technical Micro Systems owes you for any of these, send a set of your documentation to:

TMSI, c/o Lee Hart / Box 134 / 530 W. Walnut St. / Kalamazoo, MI 49005 / (616) 345-2960 (Mon-Fri 3-5 pm EST)

He will try to deliver what Technical Micro wouldn't, but makes no guarantees. He also mentioned that he will only ship from stock when your payment clears, whether cash, check, money order, or bank plastic. And the name of the new company has no "connection" with the bankrupt one!

Hard-Sector Disk Suppliers: A lot of you are still running your '8's and '89's with the original Siemens disk drives and finding disks harder and harder to get. But some of the feedback Hank and I have gotten suggest a number of sources for these vanishing treasures. Write or call about availability and price **before** you order and let me know if they provide satisfactory service. And tell them you got the information about them here!

Anapro Corp. / Peter Shkabara / 6905 El Camino Real #4 / Atascadero, CA 93422 / (805) 466-1589
Archives / 50 Howe Ave. / Millburg, MA 01527 / (617) 865-4451
Lindley Systems / William Lindley / 21 Hancock St. / Bedford, MA 01720-1613 / (617) 275-6821
Lyben Computer Systems / Box 1237 / Troy, MI 48099 / (313) 589-3440
Sofwood Enterprises / Michael Lewis / 543 N. Washington St. / Papillion, NE 68046 / (402) 339-6281

My thanks to Eugene Ducheauneau and Parks Watson for all but the last. Sofwood Enterprises supplies supplies to OMAHUG, so I've mail-ordered CDC disks from Mike.

Surplus Dealers: John Dirks writes that there are two other dealers besides Al Davis (mentioned in issue #1) handling surplus parts from Heath's operations. He also adds that Davis's phone number is (616) 471-1792. Unlike Davis, these are located right in Benton Harbor and are:

Surplus Trading Co. / Les Turk / 609 Paw Paw Ave. / Benton Harbor, MI 49022 / (616) 926-6391
Walch Surplus Electronics / Pat Walch / 219 Chippewa / Benton Harbor, MI 49022 / (616) 926-2901

The last has the largest amount of stuff for the '89 and most reasonable prices. Call to see what they have. All three will also ship U.P.S.

CP/M Vendors: Over the last six months or so, I've stumbled across a pair of vendors still selling commercial CP/M software. I suggest writing for their catalogs. They are:

Central Computer Products / 330 Central Ave. / Fillmore, CA 93015 (specify the CP/M catalog since this company also sells IBM software)
CP/M CONNECTION / Box 236 / McPherson, KS 67460

The first is the larger of the two operations and hence handles more stuff. It **also** offers a free utility program if two or more products are ordered; the **caviat** is that this freebie **only** comes on a Kaypro-formatted disk! I've ordered from both and am surprised there isn't more product crossover than there is. So CP/M **isn't** dead after all. But mention **Staunch** when you write!

Corrupt Disk Recovery: If you're having trouble with disks that go bad for other than actual physical damage, you might try this trick. Expose them to an AC-powered magnetic bulk tape eraser, such as those used for audio tape, for 15 to 20 seconds. Apparently the AC field from the eraser shakes up the magnetic domains, especially on old disks, and makes them more amenable to digital data recording. I've used this technique on both hard- and soft-sector disks since last fall with great success. To protect the disk, leave it in its sleeve and erase against the sleeve's back. I'm using an old Radio Shack catalog #44-210 eraser, but any model suitable for reel-to-reel audio tape will do the trick.

OMDOS/SMALLDOS HDOS Clone: You and I have already heard of HDOS 3.0, either on these pages or elsewhere. Imagine my surprise, then, when I read about **another** HDOS-compatible operating system in an article by Don Keller in the December **REMark.** This one was written by W. Craig (Skip) Chambers and requires less memory and disk space than HDOS 2.0. OMDOS eliminates HDOS's overlay structure entirely to save **half** of the disk space required by HDOS 2.0 and uses about 6K less RAM. SMALLDOS includes many of the features of OMDOS, but retains the system overlays and will **not** run in STAND-ALONE mode. Both systems reduce memory requirements by using RAM normally unused in the H-8's or H/Z-89's monitor. Skip includes a memory test program to check these areas.

But you **do** need one program from HDOS 2.0 to run this package: INIT.ABS. Skip doesn't supply this, but furnishes a program to patch it to make it compatible with OMDOS/SMALLDOS. He also includes three different versions of these operating systems, depending on the version of the monitor (MTR-89 or MTR-90) and number of disk drives (three or four) you have. You **can** run soft-sector drives from these systems, but you **can't** SYSGEN a soft-sector disk. SYSCMD.SYS is also enhanced to permit some command abbreviations, three new switches (today, datesince, and hold), volume label in the directory listing, and provision for three stored (or predefined) commands to reduce typing long, repetitive commands.

If for some reason you **can't** use either, Skip includes programs to patch HDOS 2.0's standard system to use 4K less RAM. And one limitation is that both systems make extensive use of Z80 instructions. So you **must** have a Z80 card in your H-8 to run these. The H-8's U8251 is also not supported, nor is MTR-88 on the '89.

But if you're interested, simply send me a check for $12 (cheaper than HDOS 3.0!) and I'll send you two standard hard-sector disks containing the systems. Both disks can be read from standard HDOS 2.0, so you can make hardcopies of the on-disk documentation before you boot and configure a system for your machine. I should further note that Skip and I will divide the proceeds since he is giving me more-or-less exclusive distribution rights to the package. So your check will also reward the developer! I've also asked Skip to assemble the source code. That will probably cover 6 disks, cost $30, and be available separately, unlike HDOS 3.0.

Parenthetically, I'm amazed at how much activity Heath's publication of the HDOS source code spawned. Too bad Digital Research won't do the same for CP/M!

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. Neither this newsletter nor its editor is responsible for damages or losses resulting from use of any information presented herein. Info from **THE STAUNCH 8/89'er** *may be reprinted only if this publication's name and* address is included. Credit should also be given to authors and other sources of said material, if known. CP/M is a registered trademark of Digital Research, Inc. REMark is a registered trademark of Heath/Zenith Users' Group. EOF